

EDITORS: R.S. Amano and B. Sundén

Computational Fluid Dynamics and Heat Transfer

EMERGING TOPICS



WIT PRESS

Computational Fluid Dynamics and Heat Transfer

WITPRESS

WIT Press publishes leading books in Science and Technology.

Visit our website for the current list of titles.

www.witpress.com

WITeLibrary

Home of the Transactions of the Wessex Institute, the WIT electronic-library provides the international scientific community with immediate and permanent access to individual papers presented at WIT conferences. Visit the WIT eLibrary at <http://library.witpress.com>

International Series on Developments in Heat Transfer

Objectives

The Developments in Heat Transfer book Series publishes state-of-the-art books and provides valuable contributions to the literature in the field of heat transfer, thermal and energy engineering. The overall aim of the Series is to bring to the attention of the international community recent advances in thermal sciences by authors in academic research and the engineering industry.

Research and development in heat transfer is of significant importance to many branches of technology, not least in energy technology. Developments include new, efficient heat exchangers, novel heat transfer equipment as well as the introduction of systems of heat exchangers in industrial processes. Application areas include heat recovery in the chemical and process industries, and buildings and dwelling houses where heat transfer plays a major role. Heat exchange combined with heat storage is also a methodology for improving the energy efficiency in industry, while cooling in gas turbine systems and combustion engines is another important area of heat transfer research. Emerging technologies like fuel cells and batteries also involve significant heat transfer issues.

To progress developments within the field both basic and applied research is needed. Advances in numerical solution methods of partial differential equations, turbulence modelling, high-speed, efficient and cheap computers, advanced experimental methods using LDV (laser-doppler-velocimetry), PIV (particle-image-velocimetry) and image processing of thermal pictures of liquid crystals, have all led to dramatic advances during recent years in the solution and investigation of complex problems within the field.

The aims of the Series are achieved by contributions to the volumes from invited authors only. This is backed by an internationally recognised Editorial Board for the Series who represent much of the active research worldwide. Volumes planned for the series include the following topics: Compact Heat Exchangers, Engineering Heat Transfer Phenomena, Fins and Fin Systems, Condensation, Materials Processing, Gas Turbine Cooling, Electronics Cooling, Combustion-Related Heat Transfer, Heat Transfer in Gas-Solid Flows, Thermal Radiation, the Boundary Element Method in Heat Transfer, Phase Change Problems, Heat Transfer in Micro-Devices, Plate-and-Frame Heat Exchangers, Turbulent Convective Heat Transfer in Ducts, Enhancement of Heat Transfer, Transport Phenomena in Fires, Fuel Cells and Batteries as well as Thermal Issues in Future Vehicles and other selected topics.

Series Editor

B. Sundén

Lund University
PO Box 118
SE-22100 Lund
Sweden

Associate Editors

R. Amano

University of Wisconsin, USA

C.A. Brebbia

Wessex Institute of Technology, UK

G. Comini

University of Udine, Italy

R.M. Cotta

COPPE/UFRJ, Brazil

S.K. Das

India Institute of Technology, Madras,
India

L. De Biase

University of Milan, Italy

G. De Mey

University of Ghent, Belgium

S. del Guidice

University of Udine, Italy

M. Faghri

University of Rhode Island, USA

C. Herman

John Hopkins University, USA

Y. Jaluria

Rutgers University, USA

S. Kabelac

Helmut Schmidt University, Hamburg,
Germany

D.B. Murray

Trinity College Dublin, Ireland

P.H. Oosthuizen

Queen's University Kingston, Canada

P. Poskas

Lithuanian Energy Institute, Lithuania

B. Sarler

Nova Gorica Polytechnic, Slovenia

A.C.M. Sousa

University of New Brunswick, Canada

D.B. Spalding

CHAM, UK

J. Szmyd

University of Mining and Metallurgy,
Poland

D. Tafti

Virginia Tech., USA

Q. Wang

Xi'an Jiatong University, China

S. Yanniotis

Agricultural University of Athens, Greece

This page intentionally left blank

Computational Fluid Dynamics and Heat Transfer

EMERGING TOPICS

Editors

R.S. Amano

University of Wisconsin-Milwaukee, USA

&

B. Sundén

Lund University, Sweden

WITPRESS Southampton, Boston 

R.S. Amano

University of Wisconsin-Milwaukee, USA

B. Sundén

Lund University, Sweden

Published by

WIT Press

Ashurst Lodge, Ashurst, Southampton, SO40 7AA, UK

Tel: 44 (0) 238 029 3223; Fax: 44 (0) 238 029 2853

E-Mail: witpress@witpress.com

<http://www.witpress.com>

For USA, Canada and Mexico

WIT Press

25 Bridge Street, Billerica, MA 01821, USA

Tel: 978 667 5841; Fax: 978 667 7582

E-Mail: infousa@witpress.com

<http://www.witpress.com>

British Library Cataloguing-in-Publication Data

A Catalogue record for this book is available
from the British Library

ISBN: 978-1-84564-144-3

ISSN (print): 1369-7331

Library of Congress Catalog Card Number: 2010920129

*The texts of the papers in this volume were set
individually by the authors or under their supervision.*

No responsibility is assumed by the Publisher, the Editors and Authors for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions or ideas contained in the material herein. The Publisher does not necessarily endorse the ideas held, or views expressed by the Editors or Authors of the material contained in its publications.

© WIT Press 2011

Printed in Great Britain by Quay Digital, Bristol.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the Publisher.

Contents

Preface	xiii
I. Finite-Volume Method	1
1 A higher-order bounded discretization scheme	3
<i>Baojun Song and R.S. Amano</i>	
1.1 Introduction	3
1.2 Numerical Formulation	6
1.2.1 Governing equations.....	6
1.2.2 Discretization.....	6
1.2.3 Higher-order schemes	7
1.2.4 Weighted-average coefficient ensuring boundedness	8
1.3 Test Problem and Results	11
1.3.1 Pure convection of a box-shaped step profile	11
1.3.2 Sudden expansion of an oblique velocity field in a cavity.....	12
1.3.3 Two-dimensional laminar flow over a fence	15
1.4 Conclusions	16
2 Higher-order numerical schemes for heat, mass, and momentum transfer in fluid flow	19
<i>Mohsen M.M. Abou-Ellail, Yuan Li and Timothy W. Tong</i>	
2.1 Introduction	20
2.2 Single-Grid Schemes	21
2.3 New Numerical Simulation Strategy	22
2.4 Novel Multigrid Numerical Procedure	23
2.5 The First Test Problem.....	29
2.6 Numerical Results of the First Test Problem.....	30
2.7 The Second Test Problem.....	39
2.8 Numerical Results of the Second Test Problem	41
2.9 Application of NIMO Scheme to Laminar Flow Problems.....	44
2.9.1 Steady laminar flow in pipes	47
2.9.2 Steady laminar flow over a fence.....	48

2.10	Application of the NIMO Higher-Order Scheme to the Turbulent Flow in Pipes	52
2.11	Conclusions	58
3	CFD for industrial turbomachinery designs	61
<i>C. Xu and R.S. Amano</i>		
3.1	Introduction	61
3.1.1	Computational methods in turbomachinery	61
3.1.2	Grid-free vortex method	63
3.2	Numerical Methods for Incompressible Flow	64
3.3	Numerical Methods for Compressible Flow.....	65
3.4	Governing Equations for Two-Dimensional Flow	70
3.5	Decomposition of Flux Vector	72
3.5.1	Governing equations.....	72
3.5.2	Special treatment of the artificial dissipation terms and numerical algorithm.....	74
3.6	Stability Analysis	77
3.7	Applications in Turbine Cascade.....	80
3.7.1	C3X turbine cascade.....	80
3.7.2	VKI turbine cascade	85
3.8	Numerical Method for Three-Dimensional Flows.....	88
3.9	Applications of Three-Dimensional Method	89
3.9.1	Analysis of pitch-width effects on the secondary- flows of turbine blades	89
3.9.2	Flow around centrifuge compressors scroll tongue ...	101
3.10	CFD Applications in Turbomachine Design	113
3.10.1	Flow solver for section analysis	116
3.10.2	Optimization	116

II. Finite Element Method **127**

4	The finite element method: discritization and application to heat convection problems	129
<i>Alessandro Mauro, Perumal Nithiarasu, Nicola Massarotti and Fausto Arpino</i>		
4.1	Governing Equations	129
4.1.1	Non-dimensional form of fluid flow equations	129
4.1.2	Non-dimensional form of turbulent flow equations....	138
4.1.3	Porous media flow: the generalized model equations.	143
4.2	The Finite Element Method.....	146
4.2.1	Strong and weak forms	146
4.2.2	Weighted residual approximation.....	148
4.2.3	The Galerkin, finite element, method	149
4.2.4	Characteristic Galerkin scheme for convection-diffusion equation	150

4.2.5	Stability conditions	155
4.2.6	Characteristic-based split scheme	157
5	Equal-order segregated finite-element method for fluid flow and heat transfer simulation	171
<i>Jianhui Xie</i>		
5.1	Introduction.....	171
5.2	Finite-Element Description.....	175
5.2.1	Two-dimensional elements	175
5.2.2	Three-dimensional elements	177
5.2.3	Degenerated elements	179
5.2.4	Special elements (rod and shell)	179
5.3	Governing Equations for Fluid Flow and Heat Transfer Problems	179
5.3.1	General form of governing equations	180
5.3.2	Discretized equations and solution algorithm	182
5.3.3	Stabilized method.....	187
5.4	Formulation of Stabilized Equal-Order Segregated Scheme .	193
5.4.1	Introduction.....	193
5.4.2	FEM-based segregated formulation	195
5.4.3	Data storage and block I/O process.....	205
5.5	Case Studies.....	208
5.5.1	Two-dimensional air cooling box	208
5.5.2	CPU water cooling analysis	210
III.	Turbulent Flow Computations/Large Eddy Simulation/Direct Numerical Simulation	215
6	Time-accurate techniques for turbulent heat transfer analysis in complex geometries	217
<i>Danesh K. Tafti</i>		
6.1	Introduction.....	217
6.2	General Form of Conservative Equations	218
6.2.1	Incompressible constant property assumption	220
6.2.2	Modeling turbulence	224
6.3	Transformed Equations in Generalized Coordinate Systems	225
6.3.1	Source terms in rotating systems.....	226
6.4	Computational Framework	227
6.5	Time-Integration Algorithm.....	231
6.5.1	Predictor step	231
6.5.2	Pressure formulation and corrector step.....	232
6.5.3	Integral adjustments at nonmatching boundaries	239

6.6	Discretization of Convection Terms.....	240
6.7	Large-Eddy Simulations and Subgrid Modeling	246
6.8	Detached Eddy Simulations or Hybrid RANS-LES	252
6.9	Solution of Linear Systems	254
6.10	Parallelization Strategies	256
6.11	Applications	259
7	On large eddy simulation of turbulent flow and heat transfer in ribbed ducts	265
	<i>Bengt Sundén and Rongguang Jia</i>	
7.1	Introduction	265
7.2	Numerical Method.....	267
7.3	Results And Discussion.....	267
7.3.1	Fully developed pipe flow	267
7.3.2	Transverse ribbed duct flow	269
7.3.3	V-shaped ribbed duct flow	271
7.4	Conclusions	273
8	Recent developments in DNS and modeling of turbulent flows and heat transfer	275
	<i>Yasutaka Nagano and Hirofumi Hattori</i>	
8.1	Introduction	275
8.2	Present State of Direct Numerical Simulations	276
8.3	Instantaneous and Reynolds-Averaged Governing Equations for Flow and Heat Transfer	277
8.4	Numerical Procedures of DNS	279
8.4.1	DNS using high-accuracy finite-difference method ...	280
8.4.2	DNS using spectral method	280
8.5	DNS of Turbulent heat Transfer in Channel Flow with Transverse-Rib Roughness: Finite-Difference Method	281
8.5.1	Heat transfer and skin friction coefficients.....	281
8.5.2	Velocity and thermal fields around the rib	284
8.5.3	Statistical characteristics of velocity field and turbulent structures	287
8.5.4	Statistical characteristics of thermal field and related turbulent structures	291
8.6	DNS of Turbulent heat Transfer in Channel Flow with Arbitrary Rotating Axes: Spectral Method.....	295
8.7	Nonlinear Eddy Diffusivity Model for Wall-Bounded Turbulent Flow	301
8.7.1	Evaluations of existing turbulence models in rotating wall-bounded flows.....	301
8.7.2	Proposal of nonlinear eddy diffusivity model for wall-bounded flow.....	304

8.8	Nonlinear Eddy Diffusivity Model for Wall-Bounded Turbulent Heat Transfer.....	310
8.8.1	Evaluations of turbulent heat transfer models in rotating channel flows	310
8.8.2	Proposal of nonlinear eddy diffusivity model for wall-bounded turbulent heat transfer.....	312
8.9	Model Performances	316
8.9.1	Prediction of rotating channel flow using NLEDMM	316
8.9.2	Prediction of rotating channel-flow heat transfer using NLEDHM.....	322
8.10	Concluding Remarks.....	327
9	Analytical wall-functions of turbulence for complex surface flow phenomena	331
	<i>K. Suga</i>	
9.1	Introduction.....	331
9.2	Numerical Implementation of Wall Functions.....	333
9.3	Standard Log-Law Wall-Function (LWF)	334
9.4	Analytical Wall-Function (AWF)	335
9.4.1	Basic strategy of the AWF	335
9.4.2	AWF in non-orthogonal grid systems	338
9.4.3	AWF for rough wall turbulent flow and heat transfer	339
9.4.4	AWF for permeable walls	352
9.4.5	AWF for high Prandtl number flows.....	359
9.4.6	AWF for high Schmidt number flows.....	365
9.5	Conclusions.....	369
9.6	Nomenclature.....	376
IV.	Advanced Simulation Modeling Technologies	381
10	SPH – a versatile multiphysics modeling tool	383
	<i>Fangming Jiang and Antonio C.M. Sousa</i>	
10.1	Introduction.....	383
10.2	SPH Theory, Formulation, and Benchmarking	384
10.2.1	SPH theory and formulation.....	384
10.2.2	Benchmarking	389
10.3	Control of the Onset of Turbulence in MHD Fluid Flow.....	392
10.3.1	MHD flow control.....	394
10.3.2	MHD modeling	395
10.3.3	SPH analysis of magnetic conditions to restrain the transition to turbulence	396
10.4	SPH Numerical Modeling for Ballistic-Diffusive Heat Conduction	400
10.4.1	Transient heat conduction across thin films	402

10.4.2	SPH modeling	403
10.4.3	Boundary treatment.....	405
10.4.4	Results.....	406
10.5	Mesoscopic Pore-Scale SPH Model for Fluid Flow in Porous Media	409
10.5.1	Modeling strategy.....	412
10.5.2	Results and discussion.....	416
10.6	Concluding Remarks.....	420
11	Evaluation of continuous and discrete phase models for simulating submicrometer aerosol transport and deposition	425
	<i>Philip Worth Longest and Jinxiang Xi</i>	
11.1	Introduction.....	426
11.2	Models of Airflow and Submicrometer Particle Transport....	428
11.2.1	Chemical species model for particle transport	428
11.2.2	Discrete phase model	429
11.2.3	Deposition factors	430
11.2.4	Numerical methods	430
11.3	Evaluation of Inertial Effects on Submicrometer Aerosols....	431
11.4	An Effective Eulerian-Based Model for Simulating Submicrometer Aerosols.....	434
11.5	Evaluation of the DF-VC Model in an Idealized Airway Geometry.....	437
11.6	Evaluation of the DF-VC Model in Realistic Airways.....	441
11.6.1	Tracheobronchial region	441
11.6.2	Nasal cavity	446
11.7	Discussion	451
12	Algorithm stabilization and acceleration in computational fluid dynamics: exploiting recursive properties of fixed point algorithms	459
	<i>Aleksandar Jemcov and Joseph P. Maruszewski</i>	
12.1	Introduction.....	460
12.2	Iterative Methods for Flow Equations.....	462
12.2.1	Discrete form of governing equations	463
12.2.2	Recursive property of iterative methods.....	465
12.3	Reduced Rank Extrapolation.....	467
12.4	Numerical Experiments.....	470
12.4.1	RRE acceleration of implicit density-based solver... 470	
12.4.2	RRE acceleration of explicit density-based solver ... 475	
12.4.3	RRE acceleration of segregated pressure-based solver	477
12.4.4	RRE acceleration of coupled pressure-based solver. 480	
12.5	Conclusion	482

Preface

The main focus of this book is to introduce computational methods for fluid flow and heat transfer to scientists, engineers, educators, and graduate students who are engaged in developing and/or using computer codes. The topic ranges from basic methods such as a finite difference, finite volume, finite element, large-eddy simulation (LES), and direct numerical simulation (DNS) to advanced, and smoothed particle hydrodynamics (SPH). The objective is to present the current state-of-the-art for simulating fluid flow and heat transfer phenomena in engineering applications.

The first and second chapters present higher-order numerical schemes. These schemes include second-order UPWIND, QUICK, weighted-average coefficient ensuring boundedness (WACEB), and non-upwind interconnected multigrid overlapping (NIMO) finite-differencing and finite-volume methods. Chapter 3 gives overview of the finite-difference and finite-volume methods covering subsonic to supersonic flow computations, numerical stability analysis, eigenvalue-stiffness problem, features of two- and three-dimensional computational schemes, and flux-vector splitting technique. The chapter shows a few case studies for gas turbine blade design and centrifugal compressor flow computations.

The fourth and fifth chapters give overview of the finite-element method and its applications to heat and fluid flow problems. An introduction to weighted residual approximation and finite-element method for heat and fluid flow equations are presented along with the characteristic-based split algorithm in Chapter 4. Chapter 5 discusses two important concepts. One is the equal-order mixed-GLS (Galerkin Least Squares) stabilized formulation, which is a generalization of SUPG (streamline-upwind/Petrov–Galerkin) and PSPG (pressure stabilizing/Petrov–Galerkin) method. The second is the numerical strategies for the solution of large systems of equations arising from the finite-element discretization of the above formulations. To solve the nonlinear fluid flow/heat transfer problem, particular emphasis is placed on segregated scheme (SIMPLE like in finite-volume method) in nonlinear level and iterative methods in linear level.

Chapters 6 through eight give numerical methods to solve turbulent flows. Chapter 6 overviews most important methods including RANS approach, LES, and DNS and describes the numerical and theoretical background comprehensively to enable the use of these methods in complex geometries. Chapter 7 demonstrates the advantages of large-eddy simulation (LES) for computations of the flow and heat transfer in ribbed ducts through a gas turbine blade. Direct numerical simulation (DNS)

is introduced in Chapter 8. In this chapter recent studies on DNS and turbulence models from the standpoint of computational fluid dynamics (CFD) and computational heat transfer (CHT) are reviewed and the trends in recent DNS research and its role in turbulence modeling are discussed in detail. Chapter 9 discusses the analytical wall-function of turbulence for complex surface flows. This chapter introduces the recently emerged analytical wall-function (AWF) methods for surface boundary conditions of turbulent flows.

Some advanced simulation modeling technologies are given in chapters 10 and 11. In Chapter 10 the current state-of-the-art and recent advances of a novel numerical method – the smoothed particle hydrodynamics (SPH) is reviewed through case studies with particular emphasis on fluid flow and heat transport. To provide sufficient background and to assess its engineering/scientific relevance, three particular case studies are used to exemplify macro- and nanoscale applications of this methodology. The first application in this chapter deals with magnetohydrodynamic (MHD) turbulence control. Chapter 11 provides the continuous and discrete phase models for simulating submicrometer aerosol transport and deposition. Lastly Chapter 12 discusses convergence acceleration of nonlinear flow solvers through use of techniques that exploit recursive properties fixed-point methods of CFD algorithms.

The authors of the chapters were all invited to contribute to this book in accordance with their expert knowledge and background. All of the chapters follow a unified outline and presentation to aid accessibility and the book provides invaluable information to researchers in computational studies.

Finally, we are grateful to the authors and reviewers for their excellent contributions to complete this book. We are thankful for the ceaseless help that was provided by the staff members of WIT Press, in particular Mr. Brian Privett and Mrs. Elizabeth Cherry, and for their encouragement in the production of this book. Finally, our appreciation goes to Dr. Carlos Brebbia who gave us strong support and encouragement to complete this project.

Ryoichi S. Amano and Bengt Sundén

I. Finite-Volume Method

This page intentionally left blank

1 A higher-order bounded discretization scheme

Baojun Song and Ryo S. Amano

University of Wisconsin–Milwaukee, Milwaukee, WI, USA

Abstract

This chapter presents an overview of the higher-order scheme and introduces a new higher-order bounded scheme, weighted-average coefficient ensuring boundedness (WACEB), for approximating the convective fluxes in solving transport equations with the finite-volume difference method. The weighted-average formulation is used for interpolating the variables at cell faces, and the weighted-average coefficient is determined from normalized variable formulation and total variation diminishing (TVD) constraints to ensure the boundedness of solutions. The new scheme is tested by solving three problems: (1) a pure convection of a box-shaped step profile in an oblique velocity field, (2) a sudden expansion of an oblique velocity field in a cavity, and (3) a laminar flow over a fence. The results obtained by the present WACEB are compared with the upwind and QUICK schemes and show that this scheme has at least the second-order accuracy while ensuring boundedness of solutions. Moreover, it is demonstrated that this scheme produces results that better agree with the experimental data in comparison with other schemes.

Keywords: Finite-volume method, Higher-order scheme

1.1 Introduction

The approximation of the convection fluxes in the transport equations has a decisive influence on the overall accuracy of any numerical solution for fluid flow and heat transfer. Although convection is represented by a simple first-order derivative, its numerical representation remains one of the central issues in CFD. The classic first-order schemes such as upwind, hybrid, and power-law are unconditionally bounded, but tend to misrepresent the diffusion transport process through the addition of numerical or “false” diffusion arising from flow-to-grid skewness. Higher-order schemes, such as the second-order upwind [1] and the third-order upwind (QUICK) [2], offer a route to improve accuracy of the computations. However, they all suffer from the boundedness problem; that is, the solutions may display unphysical

4 COMPUTATIONAL FLUID DYNAMICS AND HEAT TRANSFER

oscillations in regions of steep gradients, which can be sufficiently serious to cause numerical instability.

During the past two decades, efforts have been made to derive higher resolution and bounded schemes. In 1988, Zhu and Leschziner proposed a local oscillation-damping algorithm (LODA) [3]. Since the LODA scheme introduces the contribution of the upwind scheme, the second-order diffusion is introduced into those regions where QUICK displays unbounded behavior. In 1988, Leonard [4] developed a normalized variable formulation and presented a high-resolution bounded scheme named SHARP (simple high-accuracy resolution program). Gaskell and Lau [5] developed a scheme called SMART (sharp and monotonic algorithm for realistic transport), which employs a curvature-compensated convective transport approximation and a piecewise linear normalized variable formulation. However, numerical testing [6] shows that both SMART and SHARP need an under-relaxation treatment at each of the control volume cell faces in order to suppress the oscillatory convergence behavior. This drawback leads to an increase in the computer storage requirement, especially for three-dimensional flow calculation. In 1991, Zhu [7] proposed a hybrid linear/parabolic approximation (HLPA) scheme. However, this method has only the second-order accuracy.

In the present study, a weighted-averaged formulation is employed to interpolate variables at cell faces and the weighted-average coefficient is determined based on the normalized variable formulation and total variation diminishing (TVD) constraints. Three test cases are examined: a pure convection of a box-shaped step profile in an oblique velocity field, a sudden expansion of an oblique velocity field in a cavity, and laminar flow over a fence. Computations are performed on a generalized curvilinear coordinate system. The schemes are implemented in a deferred correction approach. The computed results are compared with those obtained using QUICK and upwind schemes and available experimental data.

In CFD research, there are three major categories to be considered for flow studies in turbines:

1. *Mathematical models* – the physical behaviors that are to be predicted totally depend on mathematical models. The choice of mathematical models should be carefully made, such as inviscid or viscous analysis, turbulence models, inclusion of buoyancy, rotation, Coriolis effects, density variation, etc.
2. *Numerical models* – selection of a numerical technique is very important to judge whether or not the models can be effectively and accurately solved. Factors that need to be reviewed for computations include the order of accuracy, treatment of artificial viscosity, consideration of boundedness of the scheme, etc.
3. *Coordinate systems* – the type and structure of the grid (structured or unstructured grids) directly affect the robustness of the solution and accuracy.

Numerical studies demand, besides mathematical representations of the flow motion, a general, flexible, efficient, accurate, and – perhaps most importantly – stable and bounded (free from numerical instability) numerical algorithm for solving a complete set of average equations and turbulence equations. The formulation

Table 1.1. Schemes used in CFD

Scheme	Developers	Order	False diffusion	Boundedness
Upwind	–	1st	High	Unconditionally bounded
Hybrid	Gosman (1977)	1st	High	Unconditionally bounded
Power-law	Patankar (1980)	1st	Medium	Unconditionally bounded
Second-order upwind	Price et al. (1966)	2nd	Low	Unbounded
QUICK	Leonard (1979)	3rd	Low	Unbounded
LODA	Zhu–Leschziner (1988)	2nd	Low	Conditionally bounded
SHARP	Leonard (1988)	2nd	Low	Conditionally bounded
SMART	Gaskell–Lau (1988)	2nd	Low	Conditionally bounded
WACEB	Song et al. (1999)	2nd	Low	Unconditionally bounded

of the discretization scheme of convection fluxes may be one of the major tasks to meet such demands.

As for the numerical method, the classic first-order schemes such as upwind, hybrid, and power-law [8] are unconditionally bounded (solutions do not suffer from over/undershoot), but tend to misrepresent the transport process through addition of numerical diffusion arising from flow-to-grid skewness. These are the schemes that most of the commercial codes employ. In some applications, small overshoots and undershoots may be tolerable. However, under other circumstances, the nonlinear processes of turbulence diffusion will feed back and amplify these over/undershoots, and may lead to divergence of a solution. During the past decade, efforts have been made to derive high-resolution and bounded schemes. LODA, SHARP, and SMART all display unbounded behavior, which leads to an increase in the computer storage requirement, especially for three-dimensional flow calculations. Therefore, the traditional method for simulating turbulent flows is the hybrid (upwind/central differencing) scheme, and the upwind is used for turbulence equations such as kinetic energy equation, dissipation rate equation, and Reynolds stress equations. Since it has a poor track record, one should always be suspicious of the first-order upwind scheme.

1.2 Numerical Formulation

1.2.1 Governing equations

The conservation equations governing incompressible steady flow problems are expressed in the following general form:

$$\operatorname{div}[\rho \vec{V} \Phi - \Gamma_\Phi \operatorname{grad}(\Phi)] = S_\Phi \quad (1)$$

where Φ is any transport variable, \vec{V} the velocity vector, ρ the density of the fluid, Γ^Φ the diffusive coefficient, and S_Φ is the source term of variable Φ .

With ξ , η , and ζ representing the general curvilinear coordinates in three-dimensional framework, the transport equation (1) can be expressed as:

$$\begin{aligned} \frac{1}{J} \left[\frac{\partial \rho U \Phi}{\partial \xi} + \frac{\partial \rho V \Phi}{\partial \eta} + \frac{\partial \rho W \Phi}{\partial \zeta} \right] &= \frac{1}{J} \frac{\partial}{\partial \xi} \left[\frac{\Gamma_\Phi}{J} (q_{11} \Phi_\xi) \right] + \frac{1}{J} \frac{\partial}{\partial \eta} \left[\frac{\Gamma_\Phi}{J} (q_{22} \Phi_\eta) \right] \\ &\quad + \frac{1}{J} \frac{\partial}{\partial \zeta} \left[\frac{\Gamma_\Phi}{J} (q_{33} \Phi_\zeta) \right] + S^{\text{CD}} + S_\Phi(\xi, \eta, \zeta) \end{aligned} \quad (2)$$

where U , V , and W are contravariant velocities defined as follows:

$$U = j_{11} u + j_{21} v + j_{31} w \quad (3a)$$

$$V = j_{12} u + j_{22} v + j_{32} w \quad (3b)$$

$$W = j_{13} u + j_{23} v + j_{33} w \quad (3c)$$

J is the Jacobian coefficient, q_{ij} and j_{ij} ($i = 1 - 3$ and $j = 1 - 3$) are the transformation coefficients (refer to the appendix), and S^{CD} is the cross-diffusion term (refer to the appendix).

1.2.2 Discretization

The computational domain is uniformly divided into hexahedral control volumes, and the discretization of transport equation (2) is performed in the computational domain following the finite-volume method.

Integrating equation (2) over a control volume as shown in Figure 1.1 and applying the *Gauss Divergence Theorem* in conjunction with central difference for diffusion, we have:

$$F_e - F_w + F_n - F_s + F_t - F_b = S_\Phi \Delta V + S^{\text{CD}} \Delta V \quad (4)$$

where F_f represents the total fluxes of Φ across the cell face f ($f = e, w, b, s, t$).

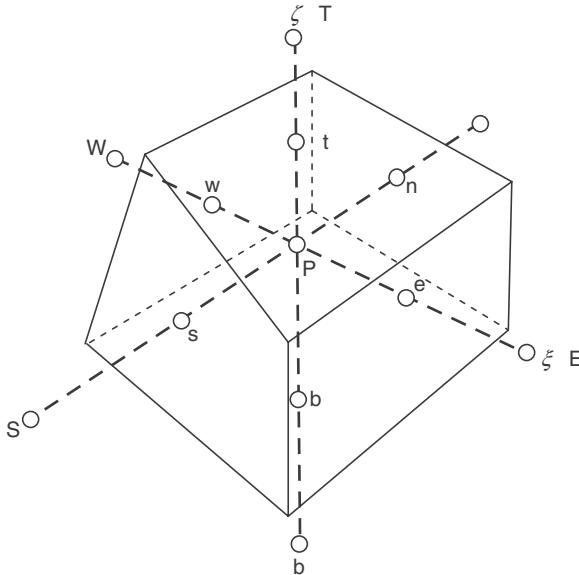


Figure 1.1. A typical control volume.

Taking the east face as an example, the total fluxes across it can be written as:

$$F_e = (\rho U \Phi)_e - \left(\frac{\Gamma^\Phi}{J} J_{11} \right)_e (\Phi_E - \Phi_P) \quad (5)$$

In the above equation, the cell face values of Φ can be approximated with different schemes.

For the first-order upwind scheme, the cell face value is expressed as:

$$\begin{aligned} \Phi_e &= \Phi_P && \text{if } U_e > 0 \\ \Phi_e &= \Phi_E && \text{if } U_e < 0 \end{aligned} \quad (6)$$

Substituting equations (5) and (6) into equation (4), we have:

$$A_P \Phi_P = \sum_{i=E,W,N,S,T,B} A_i \Phi_i + S_c \quad (7)$$

where subscript i denotes neighboring grid points, A_P and A_i the coefficients relating to the convection and diffusion, and S_c is the source term.

1.2.3 Higher-order schemes

The approximation of convection has a decisive influence on the overall accuracy of the numerical simulations for a fluid flow. The first-order schemes such as upwind, hybrid, and power-law all introduce the second-order derivatives that then lead to falsely diffusive simulated results. Therefore, the higher-order schemes have

8 COMPUTATIONAL FLUID DYNAMICS AND HEAT TRANSFER

to be used to increase the accuracy of the solution. Generally, with uniform grid spacing, the higher-order interpolation schemes can be written in the following weighted-average form:

$$\begin{aligned}\underline{\Phi_e} &= \underline{\Phi_P} + \frac{1}{4}[(1 - \kappa)\Delta_e^- + (1 + \kappa)\Delta_e] \quad \text{if } U_e > 0 \\ \underline{\Phi_e} &= \underline{\Phi_E} - \frac{1}{4}[(1 - \kappa)\Delta_e^+ + (1 + \kappa)\Delta_e] \quad \text{if } U_e < 0\end{aligned}\quad (8)$$

where $\Delta_e^- = \Phi_P - \Phi_W$, $\Delta_e = \Phi_E - \Phi_P$, $\Delta_e^+ = \Phi_{EE} - \Phi_E$ and κ is the weighted-average coefficient. In equation (8), the underlined terms represent the fragments of the first-order upwind scheme. Therefore, the higher-order schemes can be implemented in a deferred correction approach proposed by Khosla and Rubin [9]; that is,

$$\Phi_f^{n+1} = \Phi_f^{\text{UP},n+1} + (\Phi_f^{\text{HO},n} - \Phi_f^{\text{UP},n}) \quad (9)$$

where n indicates the iteration level, and UP and HO refer to the upwind and higher-order schemes, respectively. The convective fluxes calculated by the upwind schemes are combined with the diffusion term to form the main coefficients of the difference equation, while those resulting from the deferred correction terms are collected into the source term, say, S^{DC} . Such a treatment leads to a diagonally dominant coefficient matrix and enables a higher-order accuracy to be achieved at a converged stage.

With this method, the deferred correction source term, taking east–west direction as an example, is calculated by:

$$\begin{aligned}S^{\text{DC}} &= \frac{1}{4} \left\{ U_e^+ U_e [(1 + \kappa)\Delta_e + (1 - \kappa)\Delta_e^-] - U_e^- U_e [(1 + \kappa)\Delta_e + (1 - \kappa)\Delta_e^+] \right. \\ &\quad \left. - U_w^+ U_w [(1 + \kappa)\Delta_w + (1 - \kappa)\Delta_w^-] + U_w^- U_w [(1 + \kappa)\Delta_w + (1 - \kappa)\Delta_w^+] \right\}\end{aligned}\quad (10)$$

where U_e^\pm is defined as:

$$U_f^\pm = \frac{1 \pm \text{sgn}(U_f)}{2}$$

If κ is fixed at a suitable constant value everywhere, several well-known schemes can be formed.

However, the schemes listed in Table 1 all suffer from boundedness problem; that is, the solutions may display unphysical oscillations in regions of steep gradients, which can be sufficiently serious to lead to numerical instability.

1.2.4 Weighted-average coefficient ensuring boundedness

Based on the variable normalization proposed by Leonard [4], with a three-node stencil as shown in Figure 1.2, we introduce a normalized variable defined as:

$$\tilde{\Phi} = \frac{\Phi - \Phi_U}{\Phi_D - \Phi_U} \quad (11)$$

Table 1.2. Typical interpolation schemes

Expression for Φ_e when $u > 0$	Leading truncation error term
$1/2(3\Phi_P - \Phi_W)$	$3/8\Delta x^3\Phi''$
$1/2(\Phi_E + \Phi_P)$	$1/8\Delta x^2\Phi''$
$1/8(3\Phi_E + 6\Phi_P - \Phi_W)$	$1/16\Delta x^3\Phi'''$
$1/6(2\Phi_E + 5\Phi_P - \Phi_W)$	$-1/24\Delta x^2\Phi''$

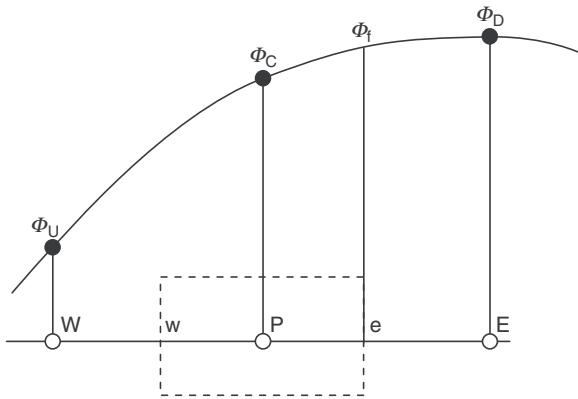


Figure 1.2. Three-node stencil.

where the subscripts U and D represent the upstream and downstream locations, respectively. In the normalized form, the higher-order schemes can be rewritten as:

$$\tilde{\Phi}_f = \tilde{\Phi}_C + \frac{1}{4}[(1 + \kappa)(1 - \tilde{\Phi}_C) + (1 - \kappa)\tilde{\Phi}_C] \quad (12)$$

See Figure 1.2 for notations of the terms. Solving for κ ,

$$\kappa = \frac{4\tilde{\Phi}_f - 4\tilde{\Phi}_C - 1}{1 - 2\tilde{\Phi}_C} \quad (13)$$

In order to ensure boundedness, the TVD constraints can be used; that is,

$$\begin{aligned} \tilde{\Phi}_f &\leq 1, \tilde{\Phi}_f \leq 2\tilde{\Phi}_C, \tilde{\Phi}_f \geq \tilde{\Phi}_C & \text{for } 0 < \tilde{\Phi}_C < 1 \\ \tilde{\Phi}_f &= \tilde{\Phi}_C & \text{for } \tilde{\Phi}_C \leq 0 \text{ or } \tilde{\Phi}_C \geq 1 \end{aligned} \quad (14)$$

which correspond to the triangle region shown in Figure 1.3.

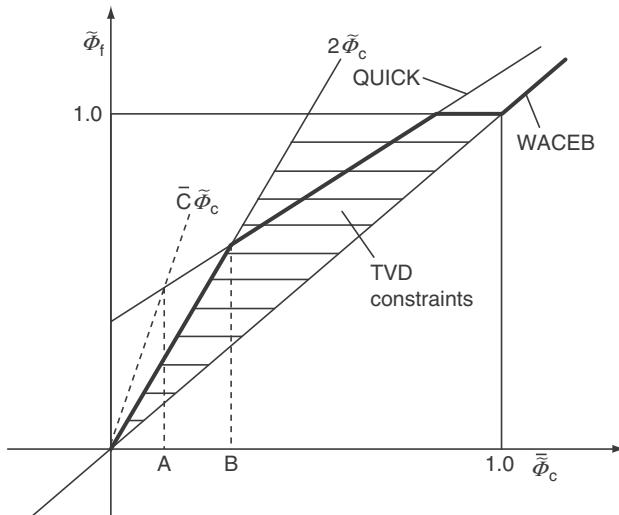


Figure 1.3. Diagrammatic representation of the TVD constraint and WACEB scheme.

The Taylor series expansion shows that the first two leading truncation error terms of the interpolation scheme (7) are $1/4(\kappa - 1/2)\Delta x^2\Phi''$ and $1/8(1 - \kappa)\Delta x^3\Phi'''$. Therefore, the scheme has at least the second-order accuracy. The maximum accuracy (third order) can be achieved if κ is set equal to 1/2. Thus, the scheme can be formed in such a way that κ lies as close as possible to 1/2, while satisfying the TVD constraints. Based on this idea, the normalized cell face value can be computed by the following expressions:

$$\tilde{\Phi}_f = \begin{cases} \tilde{\Phi}_C & \tilde{\Phi}_C \notin [0, 1] \\ 2\tilde{\Phi}_C & \tilde{\Phi}_C \in [0, 0.3) \\ 3/8(2\tilde{\Phi}_C + 1) & \tilde{\Phi}_C \in [0.3, 5/6] \\ 1 & \tilde{\Phi}_C \in (5/6, 1] \end{cases} \quad (15)$$

As shown in Figure 1.3, TVD constraints are overly restrictive according to convection boundedness criterion (CBC). However, the use of a larger multiplying constant will not noticeably increase the accuracy. The reasons are that, first, the constant affects the accuracy only in the range from A to B (see Figure 1.3), and this range varies at most from 0 to 0.3 (if we use constant 3, $A = 0.1666$ and $B = 0.3$). Secondly, even with the smaller constant, the accuracy of the scheme is still second order. Therefore, the present WACEB (weighted-average coefficient ensuring boundedness) scheme employs normalized variable formulation (15) to calculate the weighted-average coefficient to preserve boundedness.

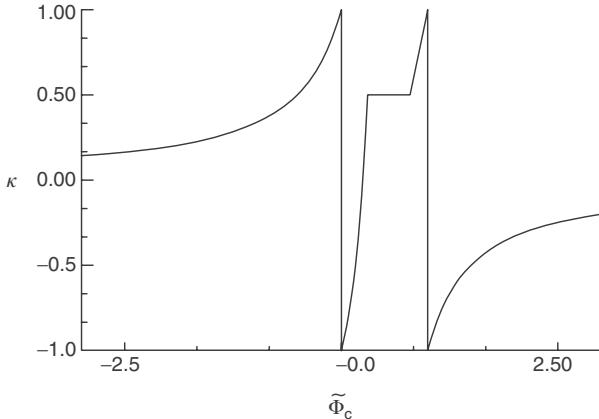


Figure 1.4. The variation in weighted-average coefficient with normalized variable.

From equations (7) and (8), the weighted-average coefficient can be given by:

$$\kappa = \begin{cases} 1/(1 - 2\tilde{\Phi}_C) & \tilde{\Phi}_C \notin [0, 1] \\ (4\tilde{\Phi}_C - 1)/1 - 2\tilde{\Phi}_C & \tilde{\Phi}_C \in [0, 0.3) \\ (3 - 4\tilde{\Phi}_C)/(1 - 2\tilde{\Phi}_C) & \tilde{\Phi}_C \in [0.3, 5/6] \\ 1/2 & \tilde{\Phi}_C \in (5/6, 1] \end{cases} \quad (16)$$

The variation in κ with $\tilde{\Phi}_C$ is shown in Figure 1.4. It is easy to see that the present WACEB scheme satisfies convective stability condition [2]. It is necessary to mention that the above algorithm is formulated on the assumption of the constant grid spacing. For nonuniform grids, the weighted-average coefficient will also be the function of the grid spacing aspect ratio.

1.3 Test Problem and Results

The governing transport equations are solved by using the nonstaggered finite-volume method. A special interpolation procedure developed by Rhie and Chow [10] is used to prevent pressure oscillations due to nonstaggered grid arrangement. Pressure and velocity coupling is achieved through the SIMPLE algorithm [8].

It is necessary to mention that QUICK and WACEB schemes all need to employ two upstream nodes for each cell face, which mandates one to involve a value outside the solution domain for a near-boundary control volume. Therefore, the upwind scheme is used for all the control volume adjacent to boundaries.

1.3.1 Pure convection of a box-shaped step profile

The flow configuration shown in Figure 1.5 constitutes a test problem for examining the performance of numerical approximation to convection because of the extremely

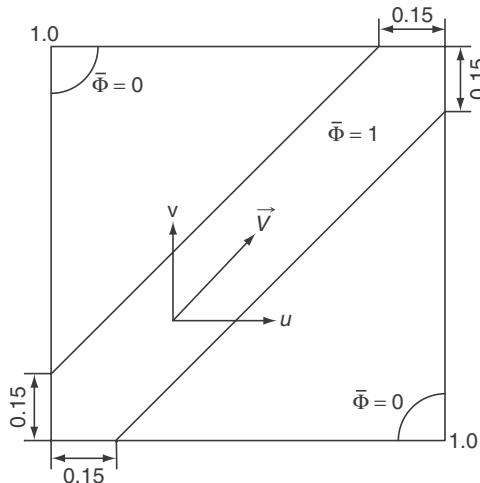


Figure 1.5. Pure convection of a box-shaped step by a uniform velocity field.

sharp gradient in a scalar. This is a linear problem in which the velocity field is prescribed. The calculations are performed with two different uniform meshes, 29×29 and 59×59 .

Comparisons of the numerical solutions obtained with the upwind, QUICK, and WACEB schemes are presented in Figure 1.6(a) and (b). It can be seen that the upwind scheme results in a quite falsely diffusive profile for the scalar even with the finer mesh. Although the QUICK scheme reduces such a false diffusion, it produces significant overshoots and undershoots. Unlikely, the WACEB predicts a fairly good steep gradient without introducing any overshoots or undershoots. Therefore, we conclude that the WACEB scheme resolves the boundedness problem while reserving a higher-order accuracy.

1.3.2 Sudden expansion of an oblique velocity field in a cavity

The geometry under consideration is depicted in Figure 1.7. The flow is assumed to be steady and laminar. At the inlet, U-velocity and V-velocity are given a constant value of U_{ref} . The boundary conditions at the outlet are $\partial U / \partial x = 0$ and $\partial V / \partial x = 0$. The calculations are performed on the uniform meshes (59×59). Figure 1.8 shows the comparison of U-velocity along the vertical central lines of the cavity for the Reynolds number 400. It is noticed that the upwind scheme cannot predict the secondary recirculation region well, which should appear near the upper side of the cavity and smears out the steep gradients of the velocity profile near the mainstream. We observe that both the WACEB and QUICK schemes distinctively predict this secondary recirculating region. Furthermore, it is noteworthy to observe that both produce very similar results. The streamline patterns predicted with the three schemes are all shown in Figure 1.9. It is clearly seen, again, that the upwind

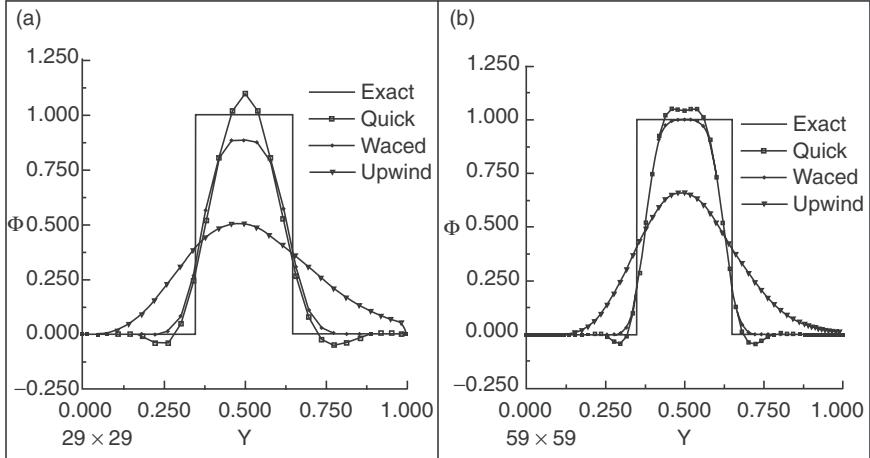


Figure 1.6. Scalar profiles along the center line.

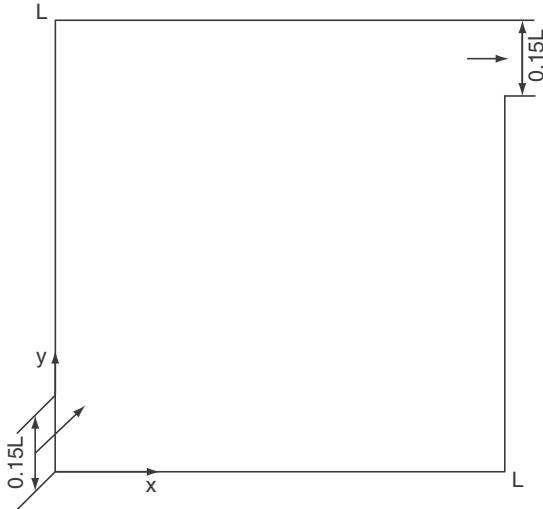


Figure 1.7. Geometry of a cavity.

scheme predicts a much smaller vortex on the upper left side of the cavity and much wider mainstream region than the QUICK and WACEB schemes. The computations were further extended to a higher Reynolds number up to 1,000. At this Reynolds number, the QUICK scheme produces a “wiggle solution.” Figure 1.10 shows streamline patterns predicted with the WACEB and upwind schemes. These two schemes give very different flow patterns; with the increase in the Reynolds number, the convection is enhanced and diffusion is suppressed and then the “dead water regions” should have less effect on the mainstream region. The results with

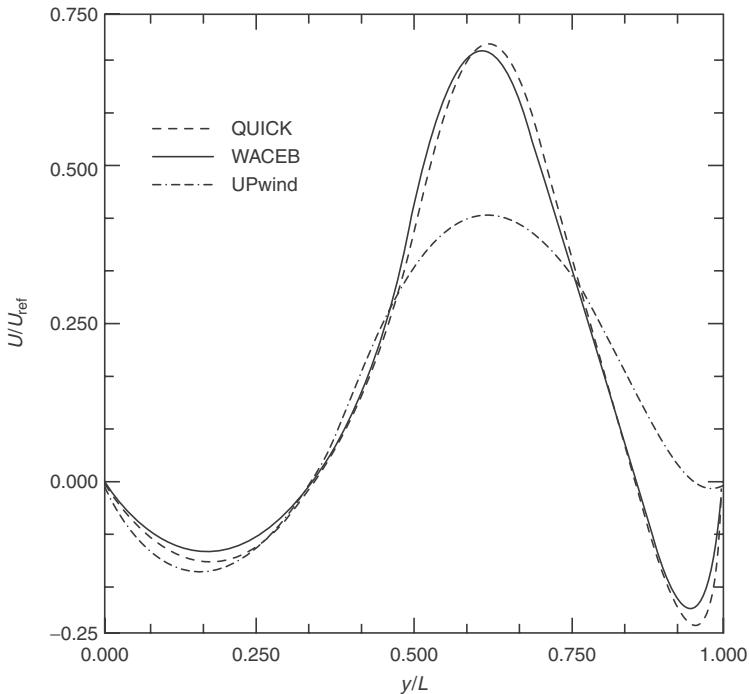


Figure 1.8. U-velocity profile along the vertical center line of the domain ($Re = 400$).

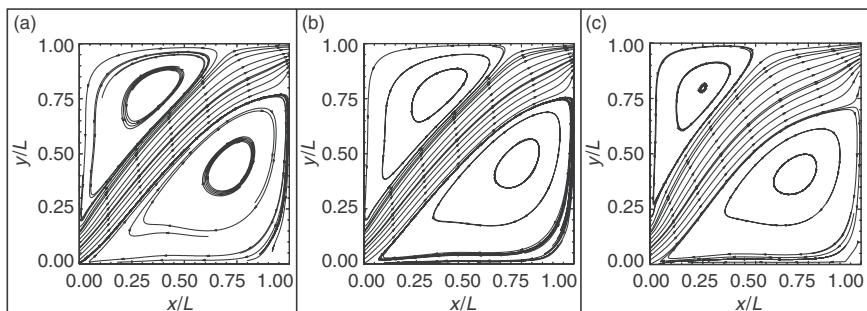


Figure 1.9. Streamlines for sudden expansion of an oblique velocity field ($Re = 400$): (a) QUICK; (b) WACEB; (c) upwind.

the WACEB scheme clearly show this trend. It is also noted that the WACEB scheme produces two additional vortices at the two corners of the cavity. However, the upwind scheme predicts only a very small additional vortex at the lower right corner and fails to capture the additional vortex at the upper left corner.

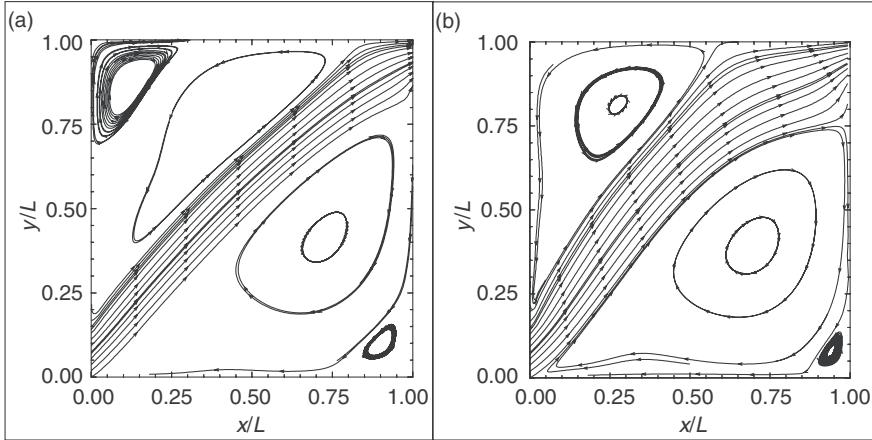


Figure 1.10. Streamlines for sudden expansion of an oblique velocity field ($Re = 1,000$): (a) WACEB; (b) upwind.

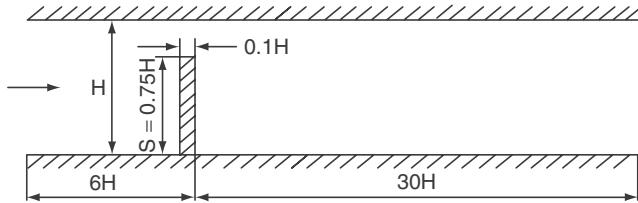


Figure 1.11. Geometry of flow over a fence.

From the above discussions, it is concluded that the solution with the WACEB scheme is comparable to that with the QUICK scheme. Even under highly convective conditions in which the unbounded QUICK scheme may produce “wiggle solutions,” the bounded WACEB scheme still produces a reasonable solution.

1.3.3 Two-dimensional laminar flow over a fence

A two-dimensional laminar flow over a fence (see Figure 1.11) with the Reynolds number based on the height of the fence, the mean axial velocity of 82.5, and the blockage ratio (s/H) of 0.75 is a benchmark case study. The boundary conditions at the inlet are prescribed as a parabolic profile for the axial velocity U and zero for the cross-flow velocity V . At the outlet, the boundary conditions are given as $\partial U / \partial x = 0$ and $\partial V / \partial x = 0$. The present study shows that the grid-independence results can be achieved with 150×78 uniform meshes for all the schemes.

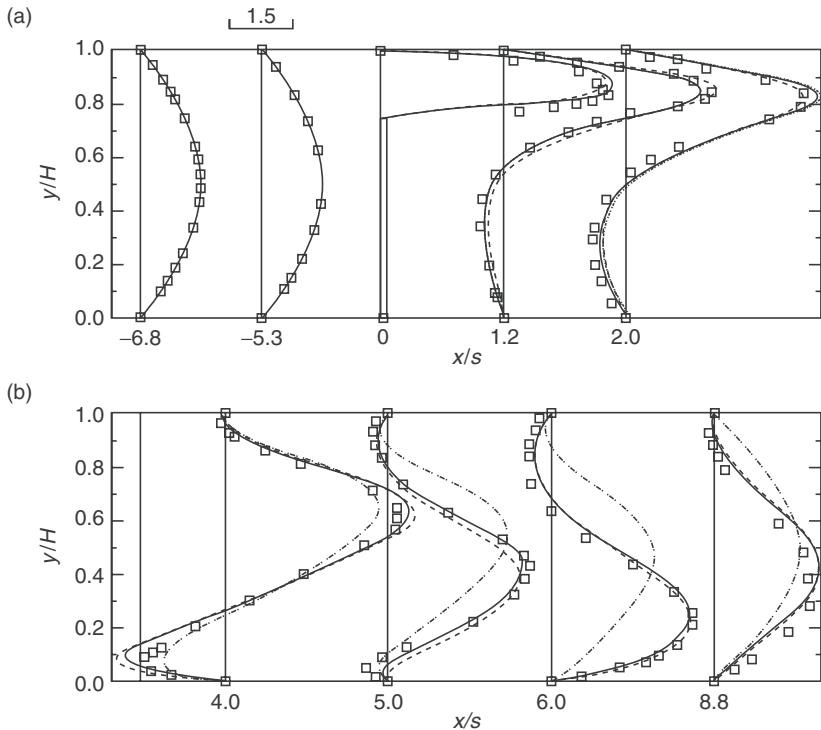


Figure 1.12. Comparison between prediction and measurement for flow over the fence ($\text{Re} = 82.5$) (square symbol, experimental data; solid line, WACEB; dashed line, QUICK; dash-dot line, upwind)

Figure 1.12 presents the axial velocity profiles at different locations (x/s) measured [11] and calculated with the QUICK, WACEB, and upwind schemes. We can observe that when x/s is less than 2, the results with the three schemes are nearly identical and are in good agreement with experimental data. However, when x/s is larger than 2, where the second separated flow on the top wall appears, the upwind scheme predicts very poor results and the QUICK and WACEB schemes give very satisfactory results in comparison with the experimental data [11]. These results verify the conclusion drawn from previous section.

1.4 Conclusions

By using normalized variable formulation and TVD constraints, the WACEB of the solution is determined and then a bounded scheme is presented in this chapter. This new scheme is tested for four different flow applications including a linear convection transport of a scalar, a sudden expansion of an oblique flow field, and a

laminar flow over a fence. The numerical tests show that the new WACEB scheme retains the ability of the QUICK to reduce the numerical diffusion without introducing any overshoots or undershoots. The scheme is very easy to implement, stable, and free of convergence oscillation and does not need to incorporate any under-relaxation treatment for weighted-average coefficient calculation.

Appendix

The cross-diffusion source term in equation (2) is defined as follows:

$$S^{\text{CD}} = \frac{1}{J} \frac{\partial}{\partial \xi} \left(\frac{\Gamma_\Phi}{J} (q_{21}\Phi_\eta + q_{31}\Phi_\zeta) \right) + \frac{1}{J} \frac{\partial}{\partial \eta} \left(\frac{\Gamma_\Phi}{J} (q_{12}\Phi_\xi + q_{32}\Phi_\zeta) \right) \\ + \frac{1}{J} \frac{\partial}{\partial \zeta} \left(\frac{\Gamma_\Phi}{J} (q_{13}\Phi_\xi + q_{23}\Phi_\eta) \right)$$

The transformation coefficients are defined as follows:

$$j_{11} = \frac{\partial y}{\partial \eta} \frac{\partial z}{\partial \zeta} - \frac{\partial y}{\partial \zeta} \frac{\partial z}{\partial \eta}, \quad j_{12} = \frac{\partial y}{\partial \zeta} \frac{\partial z}{\partial \xi} - \frac{\partial y}{\partial \xi} \frac{\partial z}{\partial \zeta}, \quad j_{13} = \frac{\partial y}{\partial \xi} \frac{\partial z}{\partial \eta} - \frac{\partial y}{\partial \eta} \frac{\partial z}{\partial \xi}, \\ j_{21} = \frac{\partial x}{\partial \zeta} \frac{\partial z}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial z}{\partial \zeta}, \quad j_{22} = \frac{\partial x}{\partial \xi} \frac{\partial z}{\partial \zeta} - \frac{\partial x}{\partial \zeta} \frac{\partial z}{\partial \xi}, \quad j_{23} = \frac{\partial x}{\partial \eta} \frac{\partial z}{\partial \xi} - \frac{\partial x}{\partial \xi} \frac{\partial z}{\partial \eta}, \\ j_{31} = \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \zeta} - \frac{\partial x}{\partial \zeta} \frac{\partial y}{\partial \eta}, \quad j_{32} = \frac{\partial x}{\partial \zeta} \frac{\partial y}{\partial \xi} - \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \zeta}, \quad j_{33} = \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \xi}$$

and

$$q_{ij} = \sum_{k=1}^3 j_{ki} j_{kj} \quad (i = 1, 3, \quad j = 1, 3)$$

Nomenclature

A	coefficients in equation (7)
F	total fluxes across the cell faces
H	height of channel
J	determinant of Jacobian
j_{ij}, q_{ij} ($i = 1, 3$ and $j = 1, 3$)	transformation factor
L	length of cavity
Re	Reynolds number
$S_\Phi, S_C, S^{\text{CD}}, S^{\text{DC}}$	source term
s	height of fence
U, V, W	contravariant velocity components
U_m	mean velocity in the channel
u, v, w	Cartesian velocity components
x, y, z	Cartesian coordinates

Greek Symbols

Γ	diffusion coefficient
κ	weighted-average coefficient
Φ	dependent variable
ξ, η, ζ	generalized curvilinear coordinates

Superscript

HO	term associated with higher-order scheme
UP	term associated with upwind scheme
n	iteration level
\sim	normalized value

Subscripts

f (=e, w, n, s, t, b)	value at the cell faces
F (=E, W, N, S, T, B)	value at the nodes

References

- [1] Price, H. S. Varga, R. S., and Warren, J. E. Application of oscillation matrices to diffusion-correction equations, *J. Math. Phys.*, 45, pp. 301–311, 1966.
- [2] Leonard, B. P. A stable and accurate convective modelling procedure based on quadratic upstream interpolation, *Comput. Methods Appl. Mech. Eng.*, 19, pp. 59–98, 1979.
- [3] Zhu, J., and Leschziner, M. A. A local oscillation-damping algorithm for higher-order convection schemes, *Comput. Methods Appl. Mech. Eng.*, 67, pp. 355–366, 1988.
- [4] Leonard, B. P. Simple high-accuracy resolution program for convective modelling of discontinuities, *Int. J. Numer. Methods Fluids*, 8, pp. 1291–1318, 1988.
- [5] Gaskell, P. H., and Lau, A. K. C. Curvature-compensated convective transport: SMART, a new boundedness preserving transport algorithm, *Int. J. Numer. Methods Fluids*, 8, pp. 617–641, 1988.
- [6] Zhu, J. On the higher-order bounded discretization schemes for finite volume computations of incompressible flows, *Comput. Methods Appl. Mech. Eng.*, 98, pp. 345–360, 1992.
- [7] Zhu, J. A low-diffusive and oscillation-free convection scheme, *Comm. Appl. Numer. Methods*, 7, pp. 225–232, 1991.
- [8] Patankar, S. V. *Numerical Heat Transfer and Fluid Flow*, McGraw-Hill, New York 1980.
- [9] Khosla, P. K., and Rubin, S. G. A diagonally dominant second-order accurate implicit scheme, *Comput. Fluids*, 2, pp. 207–209, 1974.
- [10] Rhie, C. M., and Chow, W. L. A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation, *AIAA J.*, 21, pp. 1525–1532, 1983.
- [11] Carvalho, M. G. Durst, F., and Pereira, J. C. Predictions and measurements of laminar flow over two-dimensional obstacles, *Appl. Math. Model.*, 11, pp. 23–34, 1987.

2 Higher-order numerical schemes for heat, mass, and momentum transfer in fluid flow

Mohsen M. M. Abou-Ellail, Yuan Li, and Timothy W. Tong

The George Washington University, Washington, DC, USA

Abstract

A novel numerical procedure for heat, mass, and momentum transfer in fluid flow is presented in this chapter. The new scheme is based on a non-upwind, interconnected, multigrid, overlapping (NIMO) finite-difference algorithm. In two-dimensional (2D) flows, the NIMO algorithm solves finite-difference equations for each dependent variable on four overlapping grids. The finite-difference equations are formulated using the control-volume approach, such that no interpolations are needed for computing the convective fluxes. For a particular dependent variable, four fields of values are produced. The NIMO numerical procedure is tested against the exact solution of two test problems. The first test problem is an oblique laminar 2D flow with a double-step abrupt change in a passive scalar variable for infinite Peclet number. The second test problem is a rotating radial flow in an annular sector with a single-step abrupt change in a passive scalar variable for infinite Peclet number. The NIMO scheme produces essentially the exact solution using different uniform and nonuniform square and rectangular grids for 45- and 30-degree angles of inclination. All other schemes are unable to capture the exact solution, especially for the rectangular and nonuniform grids. The NIMO scheme is also successful in predicting the exact solution for the rotating radial flow, using a uniform cylindrical-polar coordinate grid. The new higher-order scheme has also been tested against laminar and turbulent flow in pipes as well as recalculating flow behind a fence. The two laminar test problems are predicted with a very high accuracy. The turbulent flow in pipes is well predicted when the low Reynolds number k-e model is used.

Keywords: Discretization scheme, Non-upwind, Interconnected grids, Convective fluxes

2.1 Introduction

Finite-difference numerical simulations have suffered from false diffusion, which is synonymously referred to as numerical diffusion. This deficiency and other errors in computational fluid dynamics (CFD) are an inevitable outcome of the different interpolation schemes used for the convective terms. The interpolation schemes for the convective terms are classified as one-point schemes such as first-order upwind, two-point schemes such as central differencing (CD), and hybrid scheme, which is a combination of CD and upwind differencing [1–4]. Higher-order schemes, such as three-point second-order (CUI) [5], third-order quadratic interpolation for convective kinetics (QUICK [6] and QUICK-2D [7]), four-point third-order interpolation (FPTOI), and four-point fourth-order interpolation (FPFOI) [8], offer a route to improving the accuracy of the computations. The QUICK-2D scheme is an extension of the QUICK algorithm to enhance its stability in elliptic fluid flow problems [7]. It utilizes a six-point quadratic interpolation surface that favors the locally upstream points [7]. All of the above-mentioned schemes were unable to predict the exact profile along the y -axis on the mid-plane of the first test problem of the 45-degree oblique flow with infinite Peclet number [8–10]. These schemes produced uncertainties ranging from false diffusion and numerical instabilities to overshooting and undershooting [8]. Song et al. [9,10] introduced a higher-order bounded discretization algorithm (weighted-average coefficient ensuring boundedness, WACEB) to overcome overshooting and undershooting encountered in their previous FPTOI and FPFOI schemes [8]. They were able to remove the overshooting and undershooting in their numerical results of the 45-degree oblique flow. Even the higher-order schemes, mentioned above, were unable to predict the infinitely steep gradient of the scalar variable ϕ as it abruptly changes from 0 to 1 in the test problem [8–10]. Raithby [11] presented a skew differencing scheme that utilizes the upstream values prevailing along the local velocity vectors at the four faces of the 2D control volume surrounding each grid node. This skew differencing scheme would capture most of the details of the oblique flow if the grid is aligned along the local velocity vectors. Raithby obtained accurate results for the problem of step change of a passive scalar using a square uniform grid with dimensions 11×11 when the flow is inclined by a 45-degree angle [11]. In this case one of the diagonals of the control volume surrounding each node is aligned along the uniform velocity field, while the other diagonal is perpendicular to flow direction. Verma and Eswaran [12] used overlapping control volumes to discretize the physical solution domain. They obtained finite-difference equations that favor the upwind nodes from which the incoming flow emanates [12]. They tested their scheme using a 11×11 grid for the step change in passive scalar in an oblique flow problem. They were unable to capture the exact solution with a square grid for the 45-degree flow with step change in the passive scalar [12]. Interpolation for the convective transport is common to all of the above schemes, causing a varying degree of errors. Moreover, most of the above schemes involve upwind differencing, either explicitly or implicitly. However, the diffusive terms of the fluid flow governing equations are much easier and are more accurately modeled in most numerical schemes.

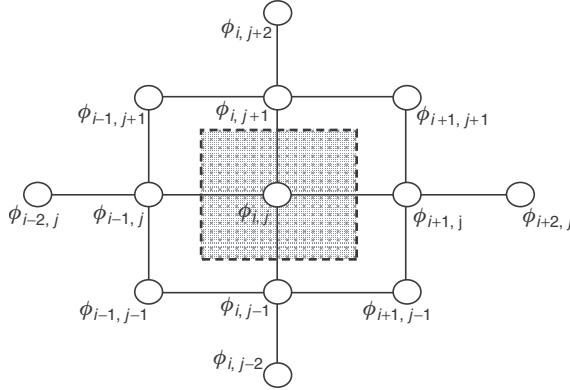


Figure 2.1. Control volume of single-grid schemes.

2.2 Single-Grid Schemes

Most of the published numerical procedures use a single grid. The main governing equations can be cast in one general form, namely,

$$\frac{\partial(\rho u_k \phi)}{\partial x_k} - \frac{\partial}{\partial x_k} \left(\Gamma_\phi \frac{\partial \phi}{\partial x_k} \right) = S_\phi \quad (1)$$

where u_k is the fluid velocity along coordinate direction x_k , while ϕ stands for any dependent variable such as mass fraction or dimensionless temperature. As explained by Abou-Ellaïl et al. [1], equation (1) can be formally integrated, over the control volume shown in Figure 2.1, to produce the following finite-difference equation:

$$\begin{aligned} & (d_{i+1,j} + d_{i-1,j} + d_{i,j+1} + d_{i,j-1} - S_p) \phi_{i,j} \\ &= d_{i+1,j} \phi_{i+1,j} + d_{i-1,j} \phi_{i-1,j} + d_{i,j+1} \phi_{i,j+1} + d_{i,j-1} \phi_{i,j-1} \\ &+ c_{i-1,j} \phi_{i-1/2,j} - c_{i+1,j} \phi_{i+1/2,j} + c_{i,j-1} \phi_{i,j-1/2} - c_{i,j+1} \phi_{i,j+1/2} + S_u \end{aligned} \quad (2)$$

where $d_{i+1,j}$ and $c_{i+1,j}$ are diffusion and convection coefficients to be computed at the center of the east face of the control volume, midway between the central node (i) and the east node ($i+1$). They are given as:

$$d_{i+1,j} = \left(\frac{A \Gamma_\phi}{\delta x} \right)_{at \ i+1/2,j} \quad (3)$$

$$c_{i+1,j} = (\rho u A)_{at i+1/2,j} \quad (4)$$

where ρ is the density, u the velocity along the x -axis, A the east face surface area, δx the distance between nodes i and $(i+1)$, Γ_ϕ the diffusion coefficient of ϕ , and S_u

and S_p are the coefficients of the integrated source term conveniently expressed as a linear expression. Equations similar to (3) and (4) apply to the west ($i - 1, j$), south ($i, j - 1$), and north ($i, j + 1$) nodes. Unlike the convective terms, the diffusion terms require no interpolation for intermediate values of ϕ . Convective terms, involving CV face values such as $\phi_{i+1/2,j}$, require interpolation between the neighboring grid nodes. The single-grid finite-difference equation can be written, for a central nodal point P and neighboring east–west–north–south nodes (E, W, N, S), as:

$$(a_E + a_W + a_N + a_S - S_p) \phi_p = a_E \phi_E + a_W \phi_W + a_N \phi_N + a_S \phi_S + S_u \quad (5)$$

The hybrid scheme defines the above finite-difference coefficients as follows:

$$a_E = \max \left[d_{i+1,j}, \left| \frac{1}{2} c_{i+1,j} \right| \right] - \frac{1}{2} c_{i+1,j} \quad (6)$$

$$a_W = \max \left[d_{i-1,j}, \left| \frac{1}{2} c_{i-1,j} \right| \right] - \frac{1}{2} c_{i-1,j} \quad (7)$$

In the above equations, $\max [\dots, \dots]$ represents the maximum value of the two values inside the bracket. Coefficients a_N and a_S have similar expressions in the hybrid scheme. The pure upwind differencing scheme has slightly different expressions for the finite-difference coefficients.

2.3 New Numerical Simulation Strategy

As mentioned above, most of the existing schemes, whether upwind or higher order, have a certain degree of false diffusion and/or over- and undershooting. All the above-mentioned schemes cannot produce an exact numerical solution to the first test problem of a 45-degree oblique flow with step changes in a passive scalar for square grids [8–10] and particularly for rectangular and nonuniform grids [11]. The present simulation strategy is based on removing all the ambiguity of interpolating for the CV face values of the scalar variable ϕ . This is done simply by superimposing four grids on the 2D solution domain. These grids are arranged in such a way that each grid uses the remaining grids to obtain directly, without any interpolations, the CV face values of the scalar variable used in computing the convective terms. The present new scheme essentially replaces the interpolation process, by finite-difference equations for the CV face values of ϕ . Therefore, the non-upwind, interconnected, multigrid, overlapping (NIMO) scheme eliminates most of the interpolation-based false diffusion that creeps into the numerical results. In addition to handling the scalar variables, the NIMO system can store the velocity components, pressure and its correction, on same space locations. In this case, the velocity components on one grid will still be located between the pressures, and their corrections, on the neighboring grids as preferred by the SIMPLE algorithm explained by Abou-Ellaïl et al. [1] and Patankar [2]. The NIMO interconnected grids share some features with the well-known staggered-grid method

[1,2]. Both methods have displaced grids, relative to a defined main grid. However, the NIMO scheme uses four overlapping grids for each dependent variable, while the staggered-grid method requires only one grid per dependent variable, as explained by Abou-Ellaïl et al. [1] and Patankar [2]. While the NIMO scheme needs no interpolations for all dependent variables, the staggered-grid method removes the need for interpolations only for the velocity components but not for the scalar variables [1,2].

2.4 Novel Multigrid Numerical Procedure

The new NIMO system is shown in Figure 2.2. The main grid defines the nodes where $\phi_{i,j}$ is located in space. Three other grids are shifted in space where $\phi_{i,j}^x$, $\phi_{i,j}^y$, and $\phi_{i,j}^{xy}$ are located midway between the main-grid nodes, as depicted in Figure 2.2. The superscripts x and y indicate shifting of grids midway with respect to the main-grid nodes. Moreover, the superscript xy indicates that the grid is shifted diagonally such that the shifted nodes occupy the center node between the neighboring four nodes of the main grid. The four-node arrowhead clusters shown in Figure 2.2 are used to indicate the common indices (e.g., i,j) affiliated with ϕ , ϕ^x , ϕ^y , and ϕ^{xy} . The spatial locations of ϕ , ϕ^x , ϕ^y , and ϕ^{xy} affiliated with cluster (i,j) , in the solution domain, are (x_i, y_j) , $(x_i + \Delta x_i/2, y_j)$, $(x_i, y_j + \Delta y_i/2)$, and $(x_i + \Delta x_i/2, y_j + \Delta y_i/2)$, respectively. This cluster technique simplifies greatly the finite-difference equations of NIMO. It also simplifies the computer coding of the system of equations of the NIMO scheme. The NIMO control volumes CV, CV^x , CV^y , and CV^{xy} are depicted

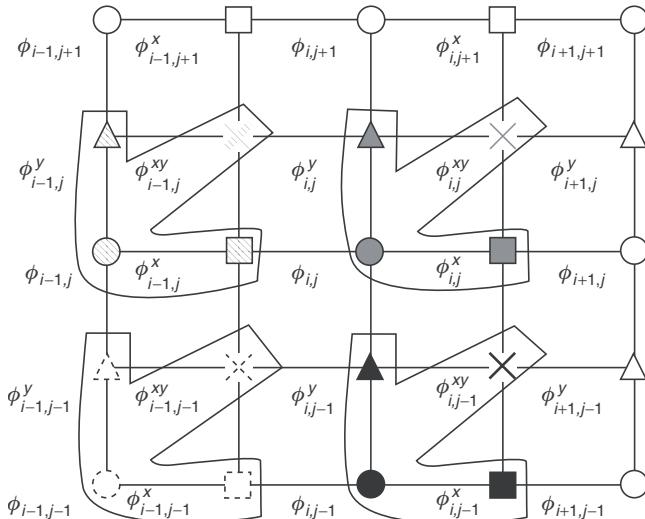


Figure 2.2. NIMO coordinate system, defining arrowhead clusters, of nodes with same indices, but differing in their spatial locations, in main, x , y , and xy grids.

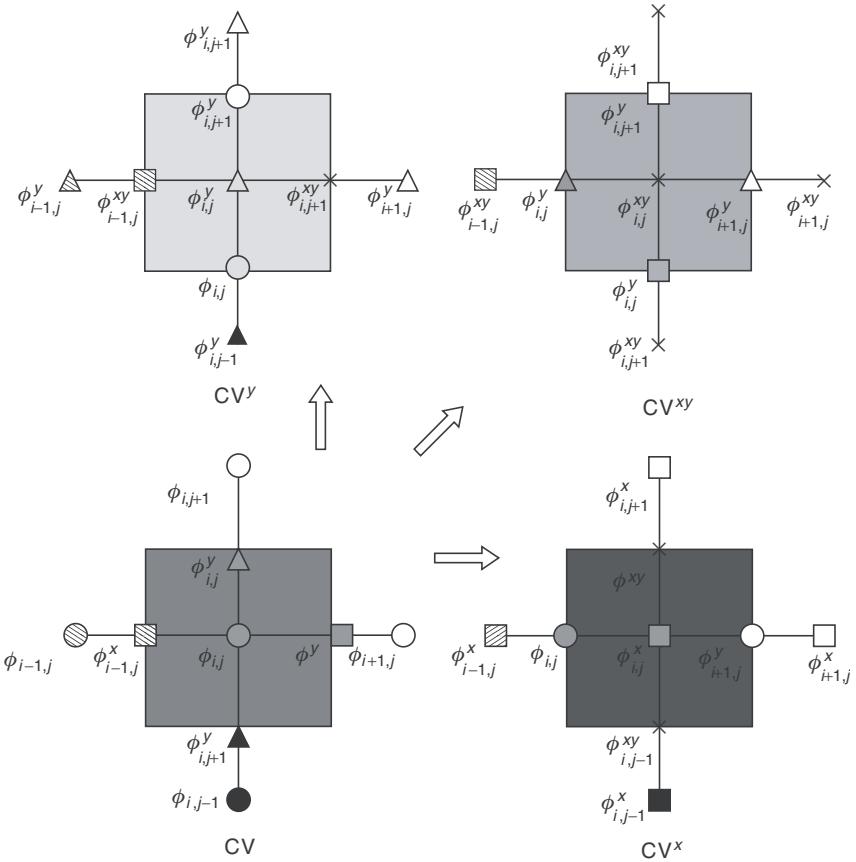


Figure 2.3. NIMO control volumes (CV , CV^x , CV^y , and CV^{xy}) surrounding the nodes of the main, x , y , and xy grids.

in Figure 2.3. The main-grid typical control volume CV is formed by the four planes bisecting the distances between the neighboring nodes and the central node (i,j) . Control volumes CV^x , CV^y , and CV^{xy} enclose nodal variables $\phi_{i,j}^x$, $\phi_{i,j}^y$, and $\phi_{i,j}^{xy}$. Along the x - or y -coordinate, the faces of these control volumes pass by the nearest neighboring nodes belonging to any of the four grids, as depicted in Figure 2.3. The acting nodal and face values used for convective fluxes of the scalar variable ϕ pertaining to each control volume surrounding each node are shown in the Figure. It should be mentioned here that the control-volume approach adopted here is similar to the finite- volume method [2]. However, in the finite-volume method, the solution domain is discretized into node-centered finite volumes [2]. With this arrangement, the computations of the convective fluxes pertaining to each control volume can be computed without the need for interpolation, even when using nonuniform grids. First, the main control volume CV is considered.

Equation (1) is formally integrated over the main control volume surrounding a typical node (i,j) where volume integrals are replaced by their surface integral counterparts performed over the four faces of the CV shown in Figure 2.3. The resulting finite-difference equation of the main grid can be written as follows:

$$\begin{aligned} & (d_{i+1,j} + d_{i-1,j} + d_{i,j+1} + d_{i,j-1} - S_{p_{i,j}})\phi_{i,j} \\ &= d_{i+1,j}\phi_{i+1,j} + d_{i-1,j}\phi_{i-1,j} + d_{i,j+1}\phi_{i,j+1} + d_{i,j-1}\phi_{i,j-1} \\ &+ c_{i-1,j}\phi_{i-1,j}^x - c_{i+1,j}\phi_{i,j}^x + c_{i,j-1}\phi_{i,j-1}^y - c_{i,j+1}\phi_{i,j}^y + S_{u_{i,j}} \end{aligned} \quad (8)$$

The finite-difference mass continuity equation of the main-grid nodes can be written as follows:

$$c_{i+1,j} - c_{i-1,j} + c_{i,j+1} - c_{i,j-1} = 0 \quad (9)$$

Equation (9) indicates, as it should, that the sum of the incoming mass fluxes is equal to the sum of the outgoing mass fluxes. Equations similar to (8) and (9) exist for control volumes CV^x , CV^y , and CV^{xy} of the other three grids. The convective and diffusive fluxes (e.g., $d_{i+1,j}$ and $c_{i+1,j}$) are still given by equations (3) and (4).

Equation (8), together with similar equations for CV^x , CV^y , and CV^{xy} , represents a closed set of finite-difference equations for ϕ , ϕ^x , ϕ^y , and ϕ^{xy} . Although they represent the same scalar variable (ϕ), they differ in their physical locations in space. However, the solution of these interconnected nonlinear equations is not easy, at least with the traditional methods, for example, the tri-diagonal matrix algorithm (TDMA). Even for a passive scalar, equation (8) has extra terms that must be included as source terms. In this case, the source terms represent passive scalar convective fluxes from the x - and y -grids to the main grid. Therefore, equation (9) is multiplied by $\phi_{i,j}$ and is used to modify equation (8). This modification is maneuvered such that the incoming fluxes of ϕ are added to the left-hand side, while the outgoing fluxes of ϕ are attached to the right-hand side of equation (8). This modification helps stabilize the solutions obtained using TDMA as part of a line-by-line alternating-direction algorithm (ADA). Since TDMA is very economical both in storage and in execution time demands, it has been favored over 2D-matrix-solver algorithms. Moreover, iterating between the four grids is inevitable as the solution of each grid is strongly dependent on the solutions of the other grids. The final NIMO finite-difference equations are as follows:

$$\begin{aligned} & (d_{i+1,j} + d_{i-1,j} + d_{i,j+1} + d_{i,j-1} - S_{p_{i,j}} + \hat{c}_{i+1,j} + \hat{c}_{i-1,j} + \hat{c}_{i,j+1} + \hat{c}_{i,j-1})\phi_{i,j} \\ &= d_{i+1,j}\phi_{i+1,j} + d_{i-1,j}\phi_{i-1,j} + d_{i,j+1}\phi_{i,j+1} + d_{i,j-1}\phi_{i,j-1} \\ &+ (c_{i-1,j}\phi_{i-1,j}^x + \tilde{c}_{i-1,j}\phi_{i,j}) + (\tilde{c}_{i+1,j}\phi_{i,j} - c_{i+1,j}\phi_{i,j}^x) \\ &+ (c_{i,j-1}\phi_{i,j-1}^y + \tilde{c}_{i,j-1}\phi_{i,j}) + (\tilde{c}_{i,j+1}\phi_{i,j} - c_{i,j+1}\phi_{i,j}^y) + S_{u_{i,j}} \end{aligned} \quad (10)$$

The terms \hat{c} and \tilde{c} are convective fluxes written in a general form to allow incoming fluxes to be transferred to the left-hand side while outgoing ones appear

only in the right-hand side of equation (10). The new convective fluxes are defined along the E–W directions as:

$$\hat{c}_{i-1,j} = \left| \frac{1}{2} c_{i-1,j} \right| + \frac{1}{2} c_{i-1,j} \quad (11)$$

$$\tilde{c}_{i-1,j} = \hat{c}_{i-1,j} - c_{i-1,j} \quad (12)$$

$$\hat{c}_{i+1,j} = \left| \frac{1}{2} c_{i+1,j} \right| - \frac{1}{2} c_{i+1,j} \quad (13)$$

$$\tilde{c}_{i+1,j} = \hat{c}_{i+1,j} + c_{i+1,j} \quad (14)$$

Similarly, along the N–S direction, \hat{c} and \tilde{c} are defined as:

$$\hat{c}_{i,j-1} = \left| \frac{1}{2} c_{i,j-1} \right| + \frac{1}{2} c_{i,j-1} \quad (15)$$

$$\tilde{c}_{i,j-1} = \hat{c}_{i,j-1} - c_{i,j-1} \quad (16)$$

$$\hat{c}_{i,j+1} = \left| \frac{1}{2} c_{i,j+1} \right| - \frac{1}{2} c_{i,j+1} \quad (17)$$

$$\tilde{c}_{i,j+1} = \hat{c}_{i,j+1} + c_{i,j+1} \quad (18)$$

It can be easily shown that the sum of the four components of \tilde{c} is equal to the sum of the four components of \hat{c} ; this is done by adding equations (12), (14), (16), and (18); that is,

$$\begin{aligned} \tilde{c}_{i-1,j} + \tilde{c}_{i+1,j} + \tilde{c}_{i,j-1} + \tilde{c}_{i,j+1} &= \hat{c}_{i-1,j} + \hat{c}_{i+1,j} + \hat{c}_{i,j-1} + \hat{c}_{i,j+1} \\ &\quad + (c_{i+1,j} - c_{i-1,j} + c_{i,j+1} - c_{i,j-1}) \end{aligned} \quad (19)$$

The sum of terms in parentheses on the right-hand side of equation (19) is equal to zero, as given by equation (9). Further simplification can be achieved if the diffusion fluxes and the convective fluxes are paired, namely,

$$\begin{aligned} &(a_{i+1,j} + a_{i-1,j} + a_{i,j+1} + a_{i,j-1} - S_{p_{i,j}}) \phi_{i,j} \\ &= d_{i+1,j} \phi_{i+1,j} + d_{i-1,j} \phi_{i-1,j} + d_{i,j+1} \phi_{i,j+1} + d_{i,j-1} \phi_{i,j-1} \\ &\quad + (c_{i-1,j} \phi_{i-1,j}^x + \tilde{c}_{i-1,j} \phi_{i,j}) + (\tilde{c}_{i+1,j} \phi_{i,j} - c_{i+1,j} \phi_{i,j}^x) \\ &\quad + (c_{i,j-1} \phi_{i,j-1}^y + \tilde{c}_{i,j-1} \phi_{i,j}) + (\tilde{c}_{i,j+1} \phi_{i,j} - c_{i,j+1} \phi_{i,j}^y) + S_{u_{i,j}} \end{aligned} \quad (20)$$

It should be mentioned that equation (20) is still equivalent to equation (10) or equation (8). The coefficient a on the left-hand side of equation (20) is similar to

the upwind differing coefficient, in as much as they both represent the net transport by convection and diffusion. However, the right-hand side has a completely different structure as it accepts the outgoing fluxes. These outgoing fluxes are reduced to zero in the pure upwind scheme. Equation (20) is found to have better convergence characteristics than the final NIMO equation (10). The above coefficients are defined as:

$$a_{i+1,j} = d_{i+1,j} + \hat{c}_{i+1,j} \quad (21)$$

$$a_{i-1,j} = d_{i-1,j} + \hat{c}_{i-1,j} \quad (22)$$

$$a_{i,j+1} = d_{i,j+1} + \hat{c}_{i,j+1} \quad (23)$$

$$a_{i,j-1} = d_{i,j-1} + \hat{c}_{i,j-1} \quad (24)$$

Similarly, the NIMO finite-difference equations for control volumes CV^x , CV^y , and CV^{xy} can be written as:

$$\begin{aligned} & (a_{i+1,j}^x + a_{i-1,j}^x + a_{i,j+1}^x + a_{i,j-1}^x - S_p^{xi,j})\phi_{i,j}^x \\ &= d_{i+1,j}^x \phi_{i+1,j}^x + d_{i-1,j}^x \phi_{i-1,j}^x + d_{i,j+1}^x \phi_{i,j+1}^x + d_{i,j-1}^x \phi_{i,j-1}^x \\ &+ (c_{i-1,j}^x \phi_{i,j}^x + \tilde{c}_{i-1,j}^x \phi_{i,j}^x) + (\tilde{c}_{i+1,j}^x \phi_{i,j}^x - c_{i+1,j}^x \phi_{i+1,j}^x) \\ &+ (c_{i,j-1}^x \phi_{i,j-1}^{xy} + \tilde{c}_{i,j-1}^x \phi_{i,j}^x) + (\tilde{c}_{i,j+1}^x \phi_{i,j}^x - c_{i,j+1}^x \phi_{i,j+1}^{xy}) + S_u^{xi,j} \end{aligned} \quad (25)$$

$$\begin{aligned} & (a_{i+1,j}^y + a_{i-1,j}^y + a_{i,j+1}^y + a_{i,j-1}^y - S_p^{yi,j})\phi_{i,j}^y \\ &= d_{i+1,j}^y \phi_{i+1,j}^y + d_{i-1,j}^y \phi_{i-1,j}^y + d_{i,j+1}^y \phi_{i,j+1}^y + d_{i,j-1}^y \phi_{i,j-1}^y \\ &+ (c_{i-1,j}^y \phi_{i,j}^{xy} + \tilde{c}_{i-1,j}^y \phi_{i,j}^y) + (\tilde{c}_{i+1,j}^y \phi_{i,j}^y - c_{i+1,j}^y \phi_{i+1,j}^{xy}) \\ &+ (c_{i,j-1}^y \phi_{i,j-1}^y + \tilde{c}_{i,j-1}^y \phi_{i,j}^y) + (\tilde{c}_{i,j+1}^y \phi_{i,j}^y - c_{i,j+1}^y \phi_{i,j+1}^y) + S_u^{yi,j} \end{aligned} \quad (26)$$

$$\begin{aligned} & (a_{i+1,j}^{xy} + a_{i-1,j}^{xy} + a_{i,j+1}^{xy} + a_{i,j-1}^{xy} - S_p^{xyi,j})\phi_{i,j}^{xy} \\ &= d_{i+1,j}^{xy} \phi_{i+1,j}^{xy} + d_{i-1,j}^{xy} \phi_{i-1,j}^{xy} + d_{i,j+1}^{xy} \phi_{i,j+1}^{xy} + d_{i,j-1}^{xy} \phi_{i,j-1}^{xy} \\ &+ (c_{i-1,j}^{xy} \phi_{i,j}^y + \tilde{c}_{i-1,j}^{xy} \phi_{i,j}^{xy}) + (\tilde{c}_{i+1,j}^{xy} \phi_{i,j}^{xy} - c_{i+1,j}^{xy} \phi_{i+1,j}^y) \\ &+ (c_{i,j-1}^{xy} \phi_{i,j-1}^x + \tilde{c}_{i,j-1}^{xy} \phi_{i,j}^{xy}) + (\tilde{c}_{i,j+1}^{xy} \phi_{i,j}^{xy} - c_{i,j+1}^{xy} \phi_{i,j+1}^x) + S_u^{xyi,j} \end{aligned} \quad (27)$$

The convective fluxes pertaining to each of the four NIMO grids are shown in the Figure 2.4. The proper locations of the fluxes entering or leaving CV , CV^x , CV^y , and CV^{xy} are shown clearly in the Figure. Interpretation of the above general NIMO equations is not easy. This interpretation becomes clear after reducing the above equation to the test problem, which is shown in Figure 2.5.

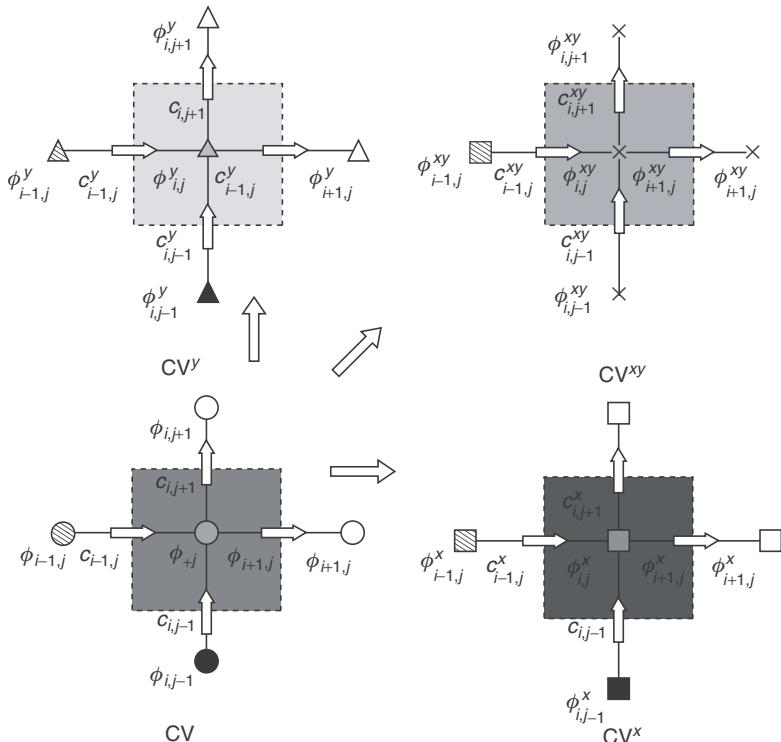


Figure 2.4. NIMO convective fluxes crossing the faces of CV, CV^x , CV^y , and CV^{xy} control volumes of the main, x , y , and xy grids.

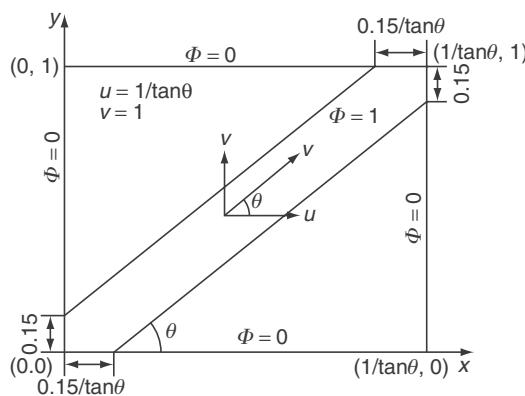


Figure 2.5. Computational domain of the test problem of inclined flow at angle θ with double-step change in passive scalar variable ϕ from 0 to 1.

2.5 The First Test Problem

The first test problem chosen to check the present NIMO scheme is shown in Figure 2.5. This oblique flow configuration was also used by Song et al. [8–10] to test their fourth-order schemes. A similar test problem with a single-step change in ϕ was utilized by Raithby [11] and by Verma and Eswaran [12]. The limiting case of the flow configuration of Figure 2.5 with infinite Peclet number has an exact analytical solution. In this case, with the absence of diffusion, the value of ϕ remains constant in the vicinity of the diagonal of the solution domain at a value of 1.0, while ϕ assumes a value of zero elsewhere, as shown in Figure 2.5. The variable ϕ is a passive scalar that has a vanishing source term. The test problem solution domain is $[(1/\tan\theta) \times 1]$, while the velocity components (u and v) are equal to $1/\tan\theta$ and 1.0, respectively. The density is also taken as constant and is equal to 1.0. The width along the x -coordinate, for $y = 0.0$ and $\phi = 1.0$, is equal to $(0.15/\tan\theta)$, while along the y -coordinate, at $x = 0.0$, it is 0.15, as shown in Figure 2.5. All the above values are in SI units. The boundary condition is also shown in Figure 2.5, with $\phi = 1.0$ in a districted zone and zero everywhere on the solution domain boundaries. The cell Peclet number at any node (i,j) is defined as:

$$\text{Pe}_{i,j} = |c_{i,j}|/d_{i,j} \quad (28)$$

For uniform convection and diffusion fluxes, the cell Peclet number is also uniform; otherwise it is grid-node dependent.

For the test problem of Figure 2.5, the values of source terms and the fluxes coefficients in equations (20), (25)–(27) are:

$$S_{p_{i,j}} = 0; \quad S_{u_{i,j}} = 0 \quad (29)$$

$$\hat{c}_{i-1,j} = c_{i-1,j}; \quad \hat{c}_{i,j-1} = c_{i,j-1} \quad (30)$$

$$\tilde{c}_{i-1,j} = 0; \quad \tilde{c}_{i,j-1} = 0 \quad (31)$$

$$\hat{c}_{i+1,j} = 0; \quad \hat{c}_{i,j+1} = 0 \quad (32)$$

$$\tilde{c}_{i+1,j} = c_{i+1,j}; \quad \tilde{c}_{i,j+1} = c_{i,j+1} \quad (33)$$

Similarly, the values of \hat{c} and \tilde{c} in the other three control volumes, CV^x , CV^y , and CV^{xy} , can be obtained. The NIMO finite-difference equations for the test problem can thus be obtained from equations (20), (25)–(27) as follows:

$$\begin{aligned} & [d_{i+1,j} + (d_{i-1,j} + c_{i-1,j}) + d_{i,j+1} + (d_{i,j-1} + c_{i,j-1})]\phi_{i,j} \\ &= d_{i+1,j}\phi_{i+1,j} + d_{i-1,j}\phi_{i-1,j} + d_{i,j+1}\phi_{i,j+1} + d_{i,j-1}\phi_{i,j-1} \\ & \quad + c_{i-1,j}\phi_{i-1,j}^x + c_{i+1,j}(\phi_{i,j} - \phi_{i,j}^x) + c_{i,j-1}\phi_{i,j-1}^y + c_{i,j+1}(\phi_{i,j} - \phi_{i,j}^y) \end{aligned} \quad (34)$$

$$\begin{aligned}
& [d_{i+1,j}^x + (d_{i-1,j}^x + c_{i-1,j}^x) + d_{i,j+1}^x + (d_{i,j-1}^x + c_{i,j-1}^x)]\phi_{i,j}^x \\
& = d_{i+1,j}^x \phi_{i+1,j}^x + d_{i-1,j}^x \phi_{i-1,j}^x + d_{i,j+1}^x \phi_{i,j+1}^x + d_{i,j-1}^x \phi_{i,j-1}^x + c_{i-1,j}^x \phi_{i,j}^x \\
& + c_{i+1,j}^x (\phi_{i,j}^x - \phi_{i+1,j}) + c_{i,j-1}^x \phi_{i,j-1}^{xy} + c_{i,j+1}^x (\phi_{i,j}^x - \phi_{i,j}^{xy}) \quad (35)
\end{aligned}$$

$$\begin{aligned}
& [d^{y_{i+1,j}} + (d^{y_{i-1,j}} + c^{y_{i-1,j}}) + d^{y_{i,j+1}} + (d^{y_{i,j-1}} + c^{y_{i,j-1}})]\phi^{y_{i,j}} \\
& = d^{y_{i+1,j}} \phi^{y_{i+1,j}} + d^{y_{i-1,j}} \phi^{y_{i-1,j}} + d^{y_{i,j+1}} \phi^{y_{i,j+1}} + d^{y_{i,j-1}} \phi^{y_{i,j-1}} + c^{y_{i-1,j}} \phi^{y_{i-1,j}} \\
& + c^{y_{i+1,j}} (\phi^{y_{i,j}} - \phi^{y_{i+1,j}}) + c^{y_{i,j-1}} \phi_{i,j} + c^{y_{i,j+1}} (\phi^{y_{i,j}} - \phi_{i,j+1}) \quad (36)
\end{aligned}$$

$$\begin{aligned}
& [d^{xy_{i+1,j}} + (d^{xy_{i-1,j}} + c^{xy_{i-1,j}}) + d^{xy_{i,j+1}} + (d^{xy_{i,j-1}} + c^{xy_{i,j-1}})]\phi^{xy_{i,j}} \\
& = d^{xy_{i+1,j}} \phi^{xy_{i+1,j}} + d^{xy_{i-1,j}} \phi^{xy_{i-1,j}} + d^{xy_{i,j+1}} \phi^{xy_{i,j+1}} + d^{xy_{i,j-1}} \phi^{xy_{i,j-1}} \\
& + c^{xy_{i-1,j}} \phi^{y_{i,j}} + c^{xy_{i+1,j}} (\phi^{xy_{i,j}} - \phi^{y_{i+1,j}}) + c^{xy_{i,j-1}} \phi^{x_{i,j}} + c^{xy_{i,j+1}} (\phi^{xy_{i,j}} - \phi^{x_{i,j+1}}) \quad (37)
\end{aligned}$$

It should be mentioned here that all the above convective fluxes are positive in the case of the test problem shown in Figure 2.5. This is because all the velocity components in the solution domain are positive, as depicted in the Figure.

The above equations share one feature with the pure upwind method. This shared feature is the left-hand sides of the above equations. However, the right-hand sides of the NIMO equations are essentially different, as they accommodate the outgoing fluxes that are ignored in the upwind scheme. Each NIMO control volume receives incoming convective fluxes from two other grids; for example the main CV receives two incoming convective fluxes from the x - and y -grids. In addition to the incoming upwind flux effects, the outgoing fluxes have a unique perturbation. They leave traces behind in the control volumes, as can be seen from the last terms of equations (34)–(37). These traces are the outcome of the differences between the CV center ϕ and the leaving ϕ values. Positive and negative traces are possible. While the incoming fluxes of ϕ tend to increase the CV central values, the traces may increase or decrease ϕ at the center of the control volume of the grid in question. Each grid receives fluxes from neighboring grids as well as traces resulting from the leaving fluxes of ϕ to the same neighboring grids, as can be seen from the above set of equations defining the NIMO scheme. The net gain in each control volume is redistributed by the diffusion mechanism, as dictated by the NIMO finite-difference equations above. The interpretations of the reduced NIMO finite-difference equations, reflecting the test problem flow conditions, apply equally to their general counterparts.

2.6 Numerical Results of the First Test Problem

The solution of the test problem is obtained numerically. The TDMA, line-by-line, ADA is adopted [1]. Equations (34)–(37) are modified at the nodes adjoining the

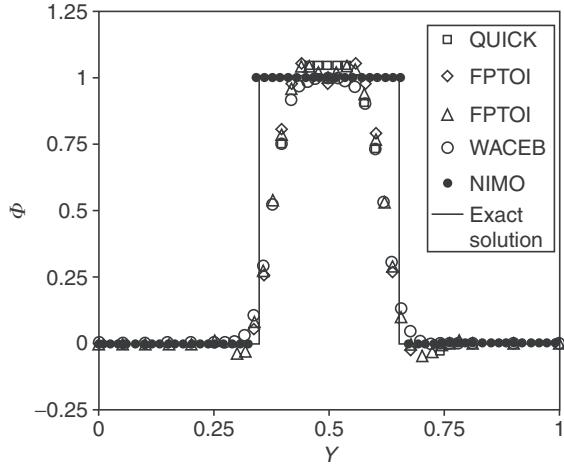


Figure 2.6. Profiles of ϕ for previous numerical schemes and NIMO at $x = 0.5$ and $\text{Pe} = \infty$ for 59×59 square grid and $\theta = 45$ degrees.

boundary to satisfy the conditions imposed there. The value of the cell Peclet number is fixed at a very large number to simulate $\text{Pe} = \infty$ of the limiting case of the test problem. A value of $\text{Pe} > 10^4$ is found sufficient enough to essentially suppress diffusion transport from equations (34)–(37). The NIMO simulation results depicted in Figure 2.6 are obtained for a fixed value of $\text{Pe} = 10^6$. The NIMO solution procedure computes the diffusive fluxes from the given Peclet number and equation (28). Different grids are used for the 45-degree oblique flow for the four grids of the NIMO scheme. These grids are as follows: one square grid (59×59), two rectangular grids (59×69 and 59×53), and one nonuniform grid (59×69). However, a uniform 99×59 grid is utilized for the 30-degree oblique flow. These different meshes are used to explore the behavior of the NIMO scheme under condition imposed by square, rectangular, and nonuniform grids. Under-relaxation factors (URFs) in the range of 0.3 to 0.5 for the 45-degree oblique flow and 0.6 for the 30-degree oblique flow are used for the finite-difference equations of each grid. The above ranges are convenient to control the conversion processes of the interlinked variables ϕ , ϕ^x , ϕ^y , and ϕ^{xy} of the NIMO scheme. The number of iterations and the residual errors of each grid are discussed below. Figure 2.6 depicts the results of Song et al. [8–10] for different single-grid schemes and the present results of the hybrid single-grid and NIMO schemes, for $\theta = 45$ degrees. The exact solution is also depicted in the Figure. The present results of the multigrid system (NIMO) gave a numerical solution that exactly fits the analytical solution of the limiting case, when $\text{Pe} = \infty$. No overshooting, undershooting, or false diffusion can be detected from the present results. It can be noticed that the NIMO scheme essentially predicts the proper infinite gradient of ϕ at $y = 0.35$ and 0.65 , as depicted in Figure 2.6. Moreover, the hybrid scheme, with its dominant false diffusion, gave

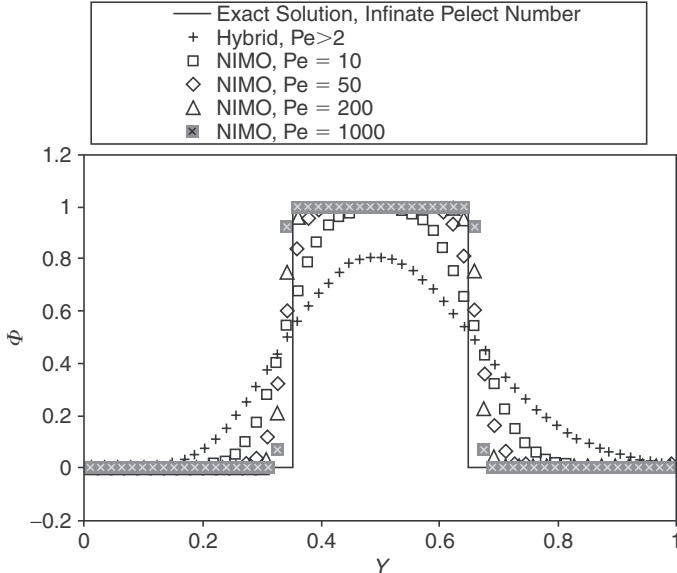


Figure 2.7. Profiles of ϕ at $x = 0.5$ for $Pe = 10, 50, 200$, and $1,000$ and exact solution for $Pe = \infty$ for 59×59 square grids and $\theta = 45$ degrees.

the least agreement with the analytical solution. The FPFOI, FPTOI, and WACEB schemes are much better than the hybrid scheme. However, FPFOI and FPTOI schemes [8] suffer from overshooting and undershooting and were unable to predict the infinite ($d\phi/dy$) at $y = 0.35$ and 0.65 . Although the WACEB scheme [9, 10] results are properly bounded between 0 and 1, it still could not predict the infinite gradient of ϕ at $y = 0.35$ and 0.65 , as can be seen from Figure 2.6.

It seemed interesting to see how NIMO works for values of Peclet number less than infinity. Numerical results of ϕ are also obtained for the same boundary conditions and using the 45-degree oblique flow. The NIMO scheme profiles of ϕ along the y -axis and at $x = 0.5$ for $Pe = 10, 50, 200$, and $1,000$ are depicted in Figure 2.7. The hybrid results for $Pe > 2$ and the analytical solution for $Pe = \infty$ are replotted in Figure 2.7 for comparison. It can be seen that as Pe increases, $(d\phi/dy)$, at $y = 0.35$ and 0.65 , increases. When $Pe = 1,000$, $(d\phi/dy)$ is so steep it almost reaches infinity.

The four-grid values of the passive scalar variable, that is, ϕ , ϕ^x , ϕ^y , and ϕ^{xy} , are plotted in Figure 2.8. The profiles of the scalar variable are plotted at $x = 0.5$ for a value of Peclet number equal to 5. The values of ϕ^x and ϕ^{xy} at $x = 0.5$ used in Figure 2.8 have been obtained by interpolating the converged final solution due to the shifting of the x - and xy -grids with respect to $x = 0.5$ plane. The four profiles fall on top of each other, indicating that the fields of ϕ , ϕ^x , ϕ^y , and ϕ^{xy} converge together to essentially the same values. The accuracy of each grid of NIMO is accessed by computing the sum, over all nodes, of the absolute residual errors.

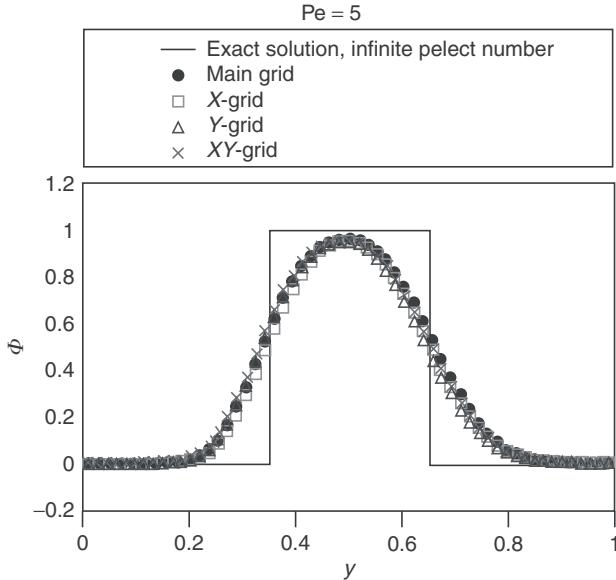


Figure 2.8. Profiles of ϕ , ϕ^x , ϕ^y , and ϕ^{xy} at $x = 0.5$ for $\text{Pe} = 5$ for 59×59 square grids and $\theta = 45$ degrees.

The residual error at a node of each grid is computed as the imbalance of the pertinent finite-difference equation.

The sums of absolute residual errors of the main grid, x -grid, y -grid, and xy -grid are plotted as a percentage of the incoming total fluxes of ϕ versus the number of iterations (n) in Figure 2.9, for $\text{Pe} = 5$ and $\text{URF} = 0.3$. The iterations are stopped when the maximum error of any grid is less than 0.1%. This condition is satisfied when $n = 975$. The above conversion criterion is also almost satisfied near $n = 600$. However, the larger number of iterations is favored to ensure complete conversion. The pattern of changes of the error curves is very interesting. The main-grid and xy -grid errors are similar and decrease monotonically. However, the x -grid and y -grid errors are nearly identical, as can be seen from Figure 2.9. They converge faster at the beginning and then suddenly diverge slightly only to converge to the desired accuracy at 975 iterations. Part of the faster convergence at the beginning is attributed to the fact that four sweeps per iteration were used for the x -grid and y -grid, while only two sweeps per iteration proved to be sufficient for the main grid and the xy -grid. It is interesting to notice that the four grids interact in such a way that the main grid and the xy -grid have no direct link. However, they are linked indirectly through their interactions with both the x -grid and y -grid. On the other hand, the x -grid and y -grid establish their indirect link by interacting directly with both the main grid and the xy -grid. The interlinking of the grids explains the pairing of the sum of the absolute residual errors of the main grid and the xy -grid as well as the pairing of the x -grid and y -grid. It should be mentioned that the initial

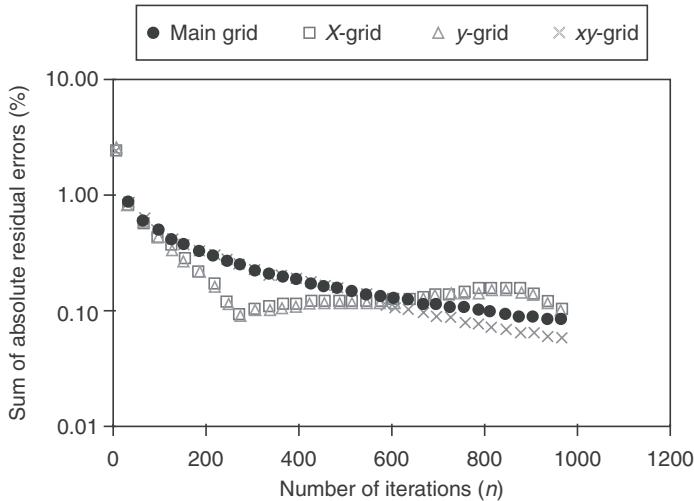


Figure 2.9. Percentage sum of absolute residual errors of the finite-difference equations of main, x -, y -, and xy -grids of NIMO for $\text{Pe} = 5$ for 59×59 square grids and $\theta = 45$ degrees.

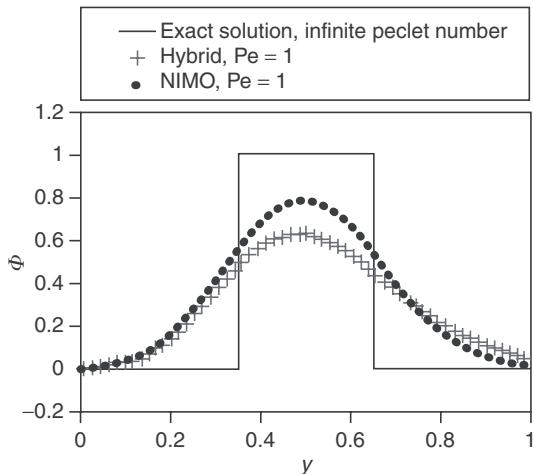


Figure 2.10. Profiles of ϕ for NIMO and hybrid schemes at $x = 0.5$ for $\text{Pe} = 1$ for 59×59 grids and $\theta = 45$ degrees.

error, which is approximately equal to 3.4, arises from the initial guess. The exact analytical solution of the test problem is used as the initial guess for ϕ , ϕ^x , ϕ^y , and ϕ^{xy} for all runs reported in the present work.

NIMO and hybrid profiles of ϕ are depicted in Figure 2.10, for $\text{Pe} = 1.0$ and $x = 0.5$. In this case, ϕ is equally transported by convection and by diffusion.

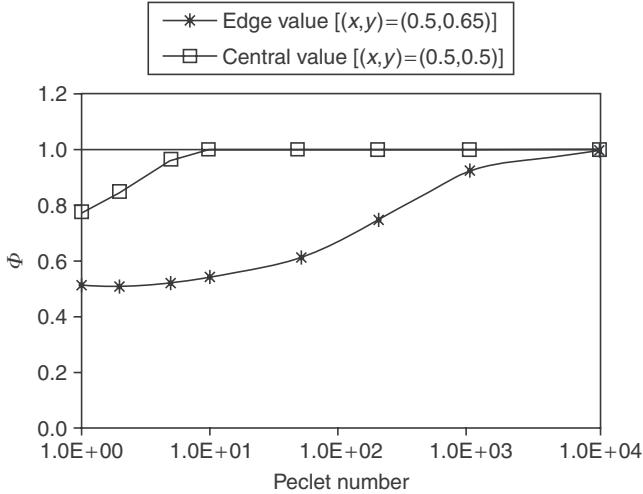


Figure 2.11. NIMO values of ϕ at edge $[(x,y)=(0.5,0.65)]$ and center of the solution domain $[(x,y)=(0.5,0.5)]$ versus Peclet number for 59×59 square grids and $\theta = 45$ degrees.

Figure 2.10 shows that the hybrid scheme produced skewed profiles of ϕ along the y -axis, while NIMO properly predicts symmetrical profiles. The skewed part of ϕ is an outcome of the inevitable false diffusion of the hybrid scheme, which results from blowing wind along the positive direction of the y -axis. This can also be detected from the hybrid scheme profile of ϕ , for $\text{Pe} > 2.0$, which is shown in Figure 2.7. Similar skewed profiles of ϕ were also obtained by Song et al. [8–10], for $\text{Pe} > 2.0$. It can be seen from Figure 2.10 that the central value of ϕ is higher in the case of NIMO. This is because the hybrid scheme suffers from extranumerical diffusion, as shown in Figure 2.10.

It is also interesting to determine the minimum Peclet number that produces essentially a flat profile of ϕ along the y -axis for $0.35 < y < 0.65$. This is achieved in Figure 2.11 by plotting the value of ϕ along the edge and center of the diagonal edge versus Pe . A value of 10^4 is just sufficient for Pe to produce nearly diffusion-free results.

The ϕ contours that produced Figure 2.11 are depicted in Figure 2.12. The skewed ϕ profile of the hybrid scheme along the y coordinate direction is also obvious from Figure 2.12a. However, the hybrid scheme contours appear to be nearly symmetrical around the diagonal of the solution domain, as this is the true flow direction. Moreover, the NIMO results indicate that ϕ is more contained inside the zone bounded by $y=0.35$ and $y=0.65$, as can be seen from Figure 2.12b.

More contours are plotted in Figure 2.13. The additional NIMO contours are for $\text{Pe} = 10, 50, 200$, and $1,000$. Figure 2.13a shows hybrid scheme ϕ contours for $\text{Pe} > 2.0$. The hybrid contours show that ϕ remains close to unity in a very

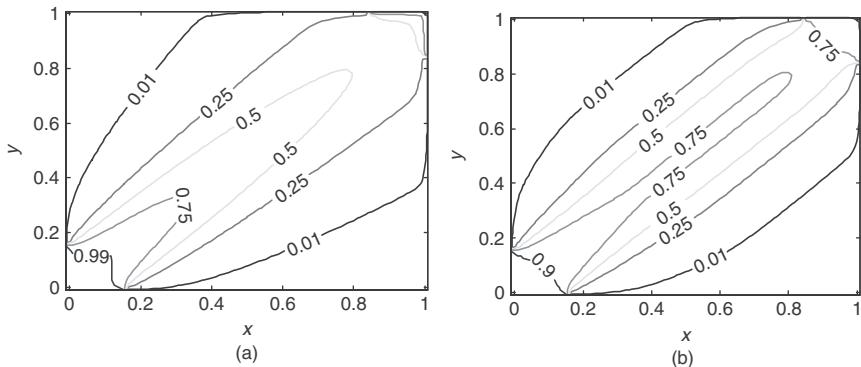


Figure 2.12. Contour plots of ϕ for unity Peclet number for 59×59 square grids and $\theta = 45$ degrees. (a) Hybrid, $\text{Pe} = 1$. (b) NIMO, $\text{Pe} = 1$.

small area next to the inlet section where ϕ assumes a value of 1.0. However, the NIMO contours show that the area where ϕ remains uncontaminated increases as the Peclet number increases. Finally at $\text{Pe} = 1,000$, most of the core along the diagonal of the solution domain is filled with unity ϕ fluid, as can be seen from Figure 2.13e. Moreover, the widths of the narrow strips, bounded by $\phi = 0.99$ and 0.01, that surround the diagonal core decrease as Peclet number increases. These narrow strips become nearly nonexistent as the Peclet number increases beyond 1,000. In order to find out how much NIMO is dependent on the boundary conditions, along the exit sections of the solution domain, a few computer runs were performed. These runs had zero ϕ gradients at the exit section instead of the fixed- ϕ -value boundary condition. These runs, although not reported here, showed that the NIMO scheme is not sensitive to this change in boundary condition for the flow configuration of Figure 2.5, except for the nodes very near to the exit boundary of the solution domain. For these nodes, insignificant differences on the order of $\pm 0.1\%$, between zero-gradient and fixed-value boundary conditions, were encountered.

In order to check the consistency of the NIMO scheme, a number of meshes were used to compute the oblique flow of Figure 2.5, for $\theta = 45$ degrees. All the grids have 59 nodes along the x -coordinate, while 53, 59, and 69 nodes are used along the y -coordinate. One of these grids is nonuniform (59×69), while the other three are uniform (59×53 , 59×59 , and 59×69). The scalar variable profiles, along the $x = 0.5$ plane, are shown in Figures 2.14 and 2.15, for two nominal values of the Peclet number, namely, 50.0 and 1,000. Only the 59×59 square grid has uniform Peclet number. However, the rectangular uniform grids have Peclet number values along the y -axis higher or lower than the nominal value along the x -axis. This is because Δx does not equal Δy for the rectangular grids, while the fluid physical properties are uniform and $u = v$ for the 45-degree flow. Moreover, the nonuniform grid has Peclet numbers that are space dependent. For a nominal Peclet number of 50.0, the profiles of ϕ at $x = 0.5$ for the above-mentioned grid dimensions are in

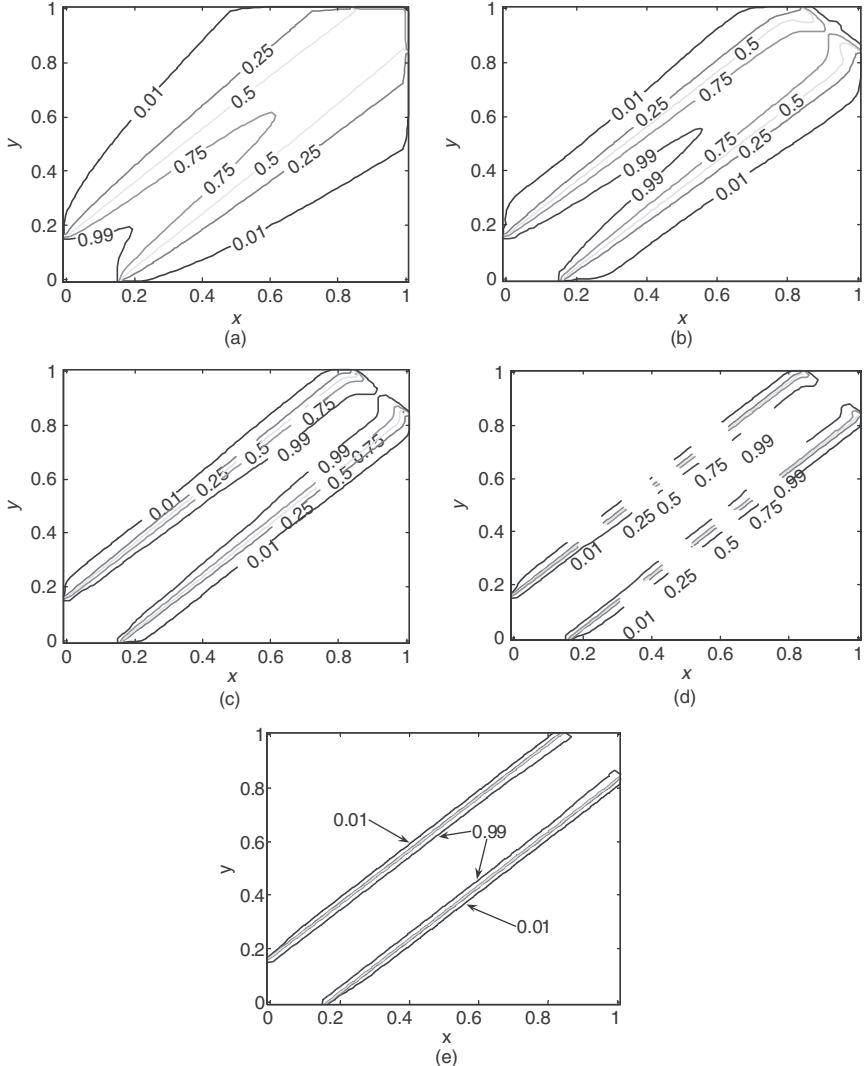


Figure 2.13. Contour plots of ϕ for hybrid scheme ($\text{Pe} > 2$) and NIMO system (59×59 square grids, $\theta = 45$ degrees and $\text{Pe} = 10, 50, 200$ and $1,000$).
 (a) Hybrid, $\text{Pe} > 2$. (b) NIMO, $\text{Pe} = 10$. (c) NIMO, $\text{Pe} = 50$. (d) NIMO, $\text{Pe} = 200$. (e) NIMO, $\text{Pe} = 1,000$.

good agreement with each other, as depicted in Figure 2.14. The minor differences between them arise from the nonuniformity of the Peclet number of the rectangular and nonuniform grids, as can be seen from the Figure.

The profiles of ϕ for different grid geometries and dimensions are depicted in Figure 2.15, for a nominal Peclet number of 1,000 and $\theta = 45$ degrees. The 59×69

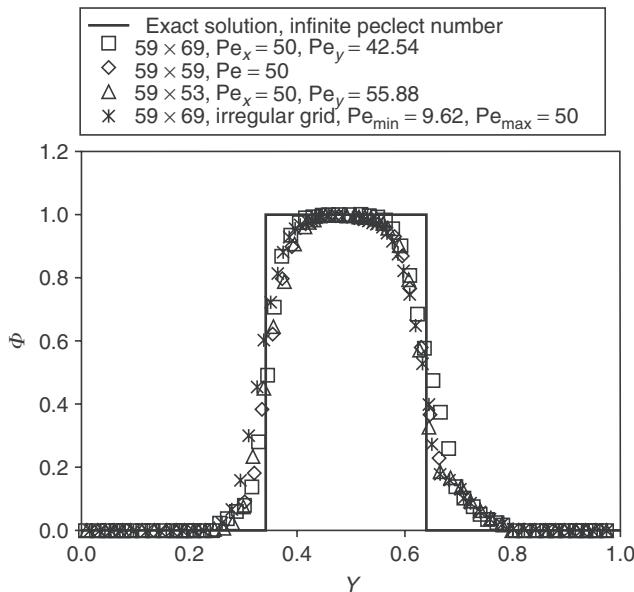


Figure 2.14. Profiles of ϕ at $x = 0.5$, for different grid dimensions and geometries for Peclet numbers along the x -axis equal to 50 and $\theta = 45$ degrees.

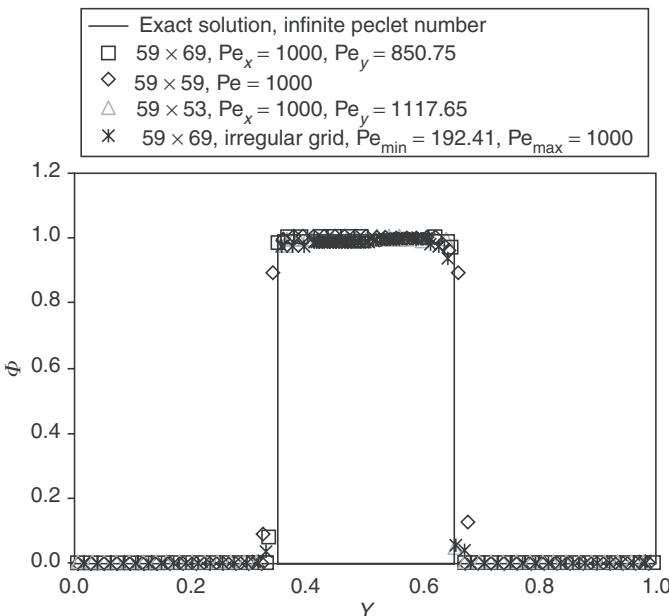


Figure 2.15. Profiles of ϕ at $x = 0.5$, for uniform and irregular different grid dimensions, Peclet numbers along the x -axis equal to 1,000 and $\theta = 45$ degrees.

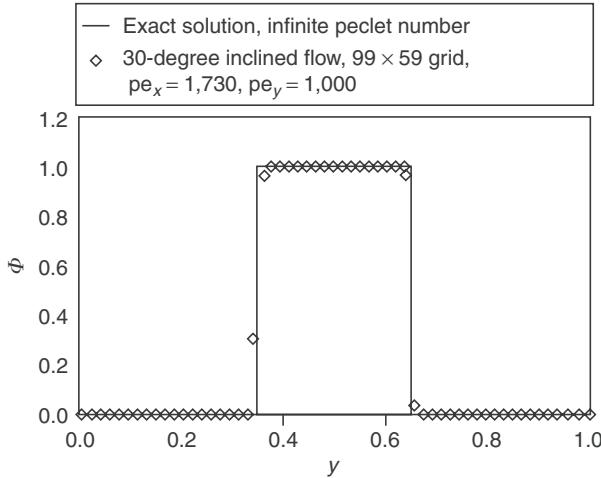


Figure 2.16. Profiles of ϕ , at $x = 0.5/\tan\theta$, for 99×59 grids, Peclet number along y -axis = 1,000, and $\theta = 30$ degrees.

nonuniform grid is uniform along the x -coordinate and nonuniform along the y direction. The 59×69 grid contracts along the y -coordinate from the outer boundaries toward the center, where $y = 0.5$. For this grid, the ratio of the maximum increment to the minimum increment along the y -coordinate is 4.5. The profiles essentially capture the analytical solution for infinite Peclet number and $\theta = 45$ degrees, although they were computed for different grid geometries and dimensions. It can be concluded that irrespective of the dimensions or geometry of the computational grid, the NIMO scheme is capable of computing the exact solution. However, the skewed upstream-differencing scheme of Raithby [11] can capture the exact solution only for square grids where the diagonals of the computational cells are parallel and normal to the flow direction.

Another test is introduced in Figure 2.16 for an angle of inclination of 30 degrees. In this case, the oblique flow passes diagonally through a rectangular solution domain. The distance of the solution domain and the velocity component along the x -axis are approximately equal to 1.73. The grid used for the profiles in the Figure has dimensions of 99×59 . The Peclet numbers along the x - and y -axes are 1,730 and 1,000, respectively. Here also none of the computational cell diagonals is parallel or normal to the flow direction. However, the profile of ϕ captures the exact analytical solution for infinite Peclet numbers, as can be seen from Figure 2.16.

2.7 The Second Test Problem

The second test problem is chosen in the cylindrical-polar domain to check the accuracy and applicability of the NIMO numerical scheme in coordinates with

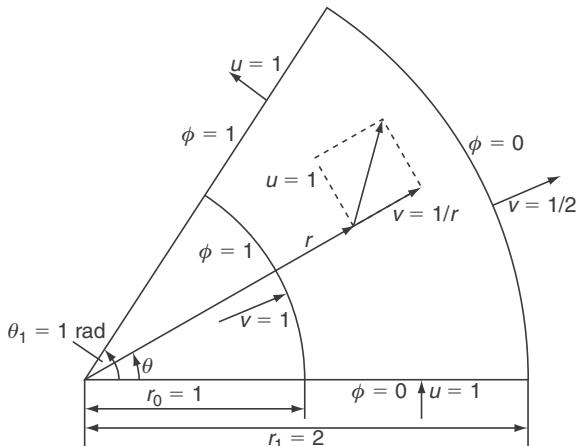


Figure 2.17. Cylindrical-polar computational domain of the test problem of rotating radial flow in an annular sector with single-step change in passive scalar variable ϕ from 0 to 1.

curvature. The configuration is that of a rotating radial flow in an annular sector. The velocity field is prescribed as shown in Figure 2.17. The tangential velocity u is taken as uniform and is equal to unity, while the radial velocity v is inversely proportional to the radial distance r . This velocity configuration satisfies the mass continuity of the constant-density rotating radial flow in an annular sector.

The annular sector is bounded by $r_0 = 1.0$, $r_1 = 2.0$, and $\theta = 0.0$ and 1.0 , along the radial and tangential directions, respectively. The passive scalar variable ϕ is equal to 0.0 and 1.0 at the two inlet sections defined by $\theta = 0.0$ and by $r = 1.0$. For infinite Peclet numbers, the exit sections have values of ϕ equal to 1.0 and 0.0 for $\theta = 1.0$ and $r = 2.0$, respectively. For this case the exact solution for the streamline that divides the sector into a $\phi = 0.0$ zone and another zone with $\phi = 1.0$ is shown in Figure 2.18. The streamline, which runs from point $(1,0)$ to point $(2,1)$, is shown in the Figure, superimposed on the prescribed velocity vector field. The velocity vectors vary in direction and magnitude, which is a good test to the extent of the accuracy of the NIMO numerical scheme in conditions where false diffusion is not uncommon. Equations (20), (25), (26), and (27) are applicable to 2D cylindrical-polar coordinates flows. The superscripts x , y , and xy , are replaced by θ , r , and $r\theta$, while the distances between the central node and the neighboring nodes become $r\delta\theta$ and δr , along the tangential and radial directions, respectively. For this test problem, a 59×59 uniform grid is superimposed on the solution domain of an annular sector. The distances along the tangential (angular) direction, which are involved in the NIMO equations, increase in the radial direction, although the grid itself is uniform.

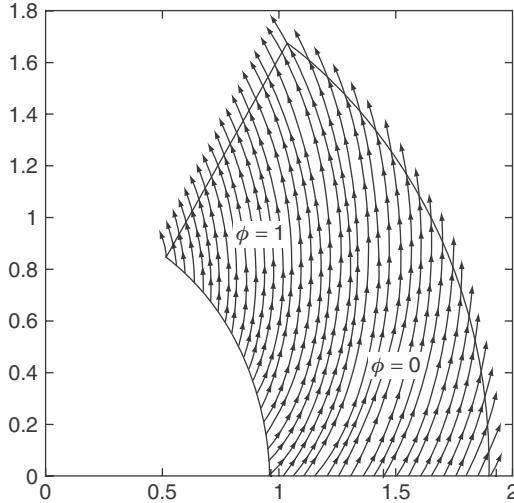


Figure 2.18. Velocity vector field for rotating radial flow in an annual sector, showing the boundary curve between regions of $\phi = 1$ and 0 in $r\theta$ plane, for infinite Peclet number.

2.8 Numerical Results of the Second Test Problem

The numerical solution of the second test problem is also obtained with the TDMA, line-by-line, ADA. The value of 10^4 for the cell Peclet number at point $(r,\theta) = (1,0)$ is found sufficient to produce nearly diffusion-free profiles of the passive scalar variable. The URF is taken as 0.5, which produced converged solutions after 500 iterations with an error less than 0.1%. The local cell Peclet number along the tangential (angular) direction increases from 10^4 to 2×10^4 , while along the radial direction it decreases from 10^4 to $5 \times 1,000$. This is an outcome of the variations along the radial direction of the radial velocity and the tangential distance.

Figure 2.19 shows the angular profiles of the passive scalar variable ϕ computed numerically and the corresponding step function exact solution. The exact solution is obtained for the limiting case of infinite Peclet number. The agreement between the two profiles is excellent. The main differences occur in a very narrow angular zone, where ϕ changes abruptly from 1.0 to 0.0. However, outside this narrow zone, ϕ is properly computed as zero on one side and unity on the other side.

Similarly, the radial profiles of the passive scalar are depicted in Figure 2.20. The NIMO results show an excellent agreement with the exact solution of a step function change in the passive scalar variable. The gradient of the passive scalar along the radial direction ($\delta\phi/\delta r$) is higher than that along the tangential direction ($\delta\phi/r\delta\theta$), as can be seen from Figures 2.19 and 2.20. This is an outcome of the higher mean value of the Peclet number along the radial direction relative to the corresponding value along the tangential direction.

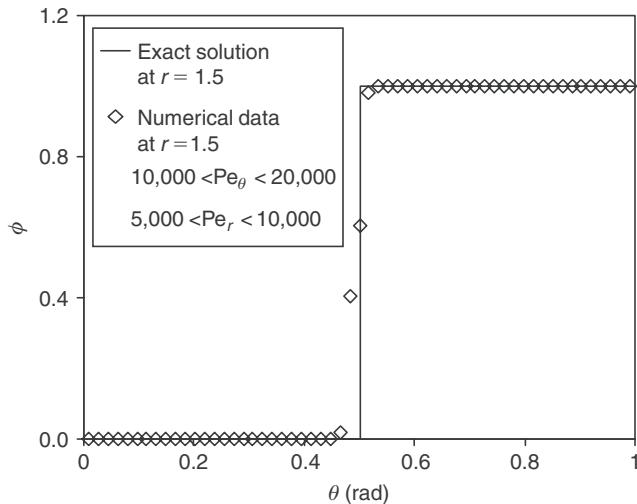


Figure 2.19. Angular profiles of ϕ , at $r = 1.5$, for Peclet numbers ranging from 5,000 to 10,000 (angular) and from 10,000 to 20,000 (radial), using 59×59 cylindrical-polar grid.

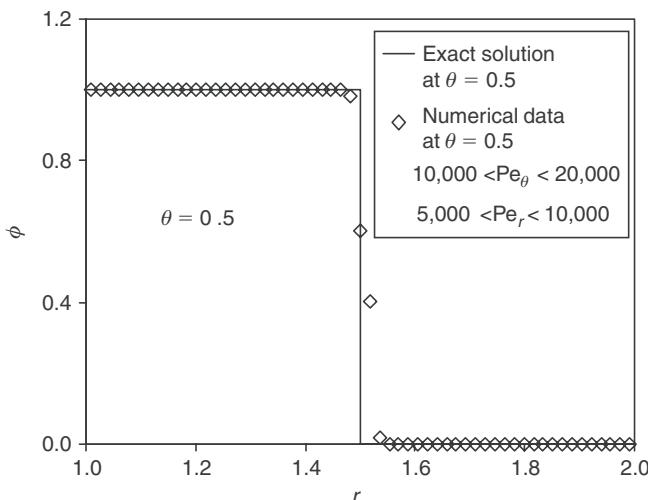


Figure 2.20. Profiles of ϕ , at $\theta = 0.5$ rad, for Peclet numbers ranging from 5,000 to 10,000 (angular) and from 10,000 to 20,000 (radial), using 59×59 cylindrical-polar grid.

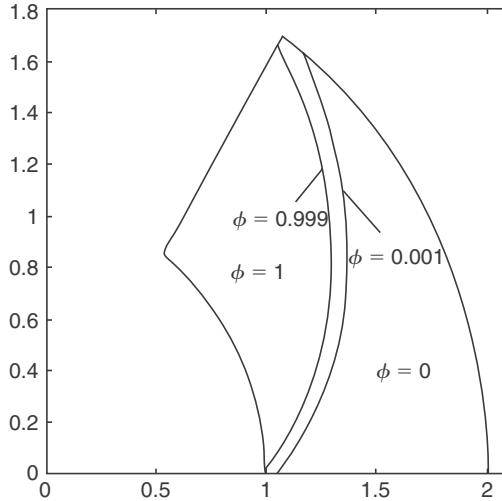


Figure 2.21. Open contours of the passive scalar variable ϕ , for a $59 \times 59 r\theta$ grid.

Figure 2.21 depicts the open contours of ϕ in the rotating radial flow in an annular sector. A thin curved strip, where ϕ varies from 0.001 to 0.999, is computed using the NIMO scheme. Outside this narrow strip the passive scalar variable changes abruptly from 0.0 to 1.0. It is interesting to notice the resemblance between the shape of the narrow strip and the exact solution curve that divides the annular sectors to $\phi = 0.0$ and 1.0 zones (Figure 2.18).

The system of finite-difference equations (equations (20), (25)–(27) can further be simplified if the central node (i,j) is replaced by the subscript P , while the neighboring nodes are denoted by the subscripts E, W, N, and S. The actual position in the xy space is determined by the superscript on the main variable ϕ to reflect the different control volumes, CV , CV^x , CV^y , and CV^{xy} of the NIMO scheme. Therefore, the NIMO equations are simplified as:

$$a_P \phi_P = \sum_n a_n \phi_n + S_N + S_u \quad (38)$$

$$a_P^x \phi_P^x = \sum_n a_n^x \phi_n^x + S_N^x + S_u^x \quad (39)$$

$$a_P^y \phi_P^y = \sum_n a_n^y \phi_n^y + S_N^y + S_u^y \quad (40)$$

$$a_P^{xy} \phi_P^{xy} = \sum_n a_n^{xy} \phi_n^{xy} + S_N^{xy} + S_u^{xy} \quad (41)$$

where the summation \sum_n is over the four neighboring nodes (E, W, N, and S) of the central node P of each grid of the NIMO scheme. The coefficients a_P , a_P^x , a_P^y , and

a_P^{xy} are defined as follows:

$$a_P = \sum_n a_n - S_P \quad (42)$$

$$a_P^x = \sum_n a_n^x - S_P^x \quad (43)$$

$$a_P^y = \sum_n a_n^y - S_P^y \quad (44)$$

$$a_P^{xy} = \sum_n a_n^{xy} - S_P^{xy} \quad (45)$$

In the above equations, S_P and S_u are the coefficients of the linearized source terms of the original governing equations. Moreover, S_N , S_N^x , S_N^y , and S_N^{xy} are extra source terms resulting from the NIMO scheme. They are given explicitly as:

$$\begin{aligned} S_N = & (\tilde{c}_W \phi_P + c_W \phi_W^x - \hat{c}_W \phi_W) + (\tilde{c}_E \phi_P - c_E \phi_P^x - \hat{c}_E \phi_E) \\ & + (\tilde{c}_S \phi_P + c_S \phi_S^y - \hat{c}_S \phi_S) + (\tilde{c}_N \phi_P - c_N \phi_P^y - \hat{c}_N \phi_N) \end{aligned} \quad (46)$$

$$\begin{aligned} S_N^x = & (\tilde{c}_W^x \phi_P^x + c_W^x \phi_P - \hat{c}_W^x \phi_W^x) + (\tilde{c}_E^x \phi_P^x - c_E^x \phi_E - \hat{c}_E^x \phi_E^x) \\ & + (\tilde{c}_S^x \phi_P^x + c_S^x \phi_S^{xy} - \hat{c}_S^x \phi_S^x) + (\tilde{c}_N^x \phi_P^x - c_N^x \phi_P^{xy} - \hat{c}_N^x \phi_N^x) \end{aligned} \quad (47)$$

$$\begin{aligned} S_N^y = & (\tilde{c}_W^y \phi_P^y + c_W^y \phi_W^{xy} - \hat{c}_W^y \phi_W^y) + (\tilde{c}_E^y \phi_P^y - c_E^y \phi_P^{xy} - \hat{c}_E^y \phi_E^y) \\ & + (\tilde{c}_S^y \phi_P^y + c_S^y \phi_S^y - \hat{c}_S^y \phi_S^y) + (\tilde{c}_N^y \phi_P^y - c_N^y \phi_N - \hat{c}_N^y \phi_N^y) \end{aligned} \quad (48)$$

$$\begin{aligned} S_N^{xy} = & (\tilde{c}_W^{xy} \phi_P^{xy} + c_W^{xy} \phi_P^y - \hat{c}_W^{xy} \phi_W^{xy}) + (\tilde{c}_E^{xy} \phi_P^{xy} - c_E^{xy} \phi_E^y - \hat{c}_E^{xy} \phi_E^{xy}) \\ & + (\tilde{c}_S^{xy} \phi_P^{xy} + c_S^{xy} \phi_S^x - \hat{c}_S^{xy} \phi_S^{xy}) + (\tilde{c}_N^{xy} \phi_P^{xy} - c_N^{xy} \phi_N^x - \hat{c}_N^{xy} \phi_N^{xy}) \end{aligned} \quad (49)$$

In the above equations the finite-difference coefficients a_E , a_W , a_N , and a_S are still given, respectively, by equations (21)–(24). It should be mentioned here that the exact location of a particular grid node is determined by both the subscript and the superscript attached to the variable in question. For example, ϕ_P^x is the dependent variable at the central node P of the control volume CV^x reflecting the shifting of the x -grid with respect to the main grid, as illustrated in Figure 2.3.

2.9 Application of NIMO Scheme to Laminar Flow Problems

The NIMO scheme is now applied to steady flow problems. For constant density and viscosity laminar flow problems, the governing equations for momentum and

mass are written in tensor form as follows:

$$\frac{\partial(\rho u_i u_j)}{\partial x_i} - \frac{\partial}{\partial x_i} \left(\mu \frac{\partial u_j}{\partial x_i} \right) = - \frac{\partial p}{\partial x_j} \quad (50)$$

$$\frac{\partial(\rho u_i)}{\partial x_i} = 0 \quad (51)$$

where μ is the laminar viscosity, u_j is the velocity component along the x_j coordinate, and p is the fluid pressure. For 2D flow $u_j = (u_1, u_2)$ while $x_j = (x_1, x_2)$. On the other hand, if $x - y$ plane is used then $u_j = (u, v)$ and $x_j = (x, y)$. In this case, invoking equations (38)–(41), the finite-difference equations for u and v are given as:

$$a_{uP} u_P = \sum a_{un} u_n + A_{uE}(p_P^x - p_E^x) + S_{uN} \quad (52)$$

$$a_{vP} v_P = \sum a_{vn} v_n + A_{vS}(p_S^y - p_S^y) + S_{vN} \quad (53)$$

where subscripts u and v denote coefficients pertaining to the velocity components u and v . Also, A_u and A_v are areas of main-grid control-volume CV surfaces normal to u and v , respectively. Similarly, for CV^x , CV^y , and CV^{xy} the momentum finite-difference equations are as follows:

$$a_{uP}^x u_P^x = \sum a_{un}^x u_n^x + A_{uE}^x(p_P - p_E) + S_{uN}^x \quad (54)$$

$$a_{vP}^x v_P^x = \sum a_{vn}^x v_n^x + A_{vS}^x(p_S^{xy} - p_P^{xy}) + S_{vN}^x \quad (55)$$

$$a_{uP}^y u_P^y = \sum a_{un}^y u_n^y + A_{vW}^y(p_W^{xy} - p_P^{xy}) + S_{uN}^y \quad (56)$$

$$a_{vP}^y v_P^y = \sum a_{vn}^y v_n^y + A_{vN}^y(p_P - p_N) + S_{vN}^y \quad (57)$$

$$a_{uP}^{xy} u_P^{xy} = \sum a_{un}^{xy} u_n^{xy} + A_{uE}^{xy}(p_P^y - p_E^y) + S_{uN}^{xy} \quad (58)$$

$$a_{vP}^{xy} v_P^{xy} = \sum a_{vn}^{xy} v_n^{xy} + A_{vN}^{xy}(p_P^x - p_N^x) + S_{vN}^{xy} \quad (59)$$

Equations (52)–(59) define the eight velocity components of the four grids of the NIMO scheme. Additional equations for p , p^x , p^y , and p^{xy} are needed to close the above equations. This is done by writing the pressure correction equations for the NIMO scheme control volume CV , CV^x , CV^y , and CV^{xy} . These pressure correction equations are the counterparts of the mass continuity equations for CV , CV^x , CV^y , and CV^{xy} control volumes.

The pressure correction equation of CV can be obtained from the corrected velocity components u_P^x , u_W^x , v_P^y , and v_S^y , which are used in the mass continuity finite-difference equation of control volume CV . The corrected velocity components are given as:

$$u_P^x = u_P^{x*} + D u_E^x (p'_P - p'_E) \quad (60)$$

$$u_W^x = u_W^{x*} + Du_E^x(p'_W - p'_P) \quad (61)$$

$$v_P^y = v_P^{y*} + Dv_N^y(p'_P - p'_N) \quad (62)$$

$$v_S^y = v_S^{y*} + Dv_S^y(p'_S - p'_P) \quad (63)$$

where u_P^x , u_W^x , v_P^y , and v_S^y are the corrected velocity components pertaining to CV. Corresponding to the above corrected velocity components, the pressure at each grid node is corrected to $(p^* + p')$. Moreover, the asterisk superscript indicates momentum-based velocity components and pressure. In equations (60)–(63), the coefficients Du_E^x and Dv_N^y are defined as:

$$Du_E^x = \left(\frac{\partial u}{\partial p} \right)_E = \frac{A_{uE}^x}{a_{uP}^x} \quad (64)$$

$$Dv_N^y = \left(\frac{\partial v}{\partial p} \right)_N = \frac{A_{vN}^y}{a_{uP}^y} \quad (65)$$

Similar expressions exist for Du_W^x and Dv_S^y . Substituting equations (60)–(63) in the mass continuity equation of CV, the corresponding pressure correction equation can be obtained as follows:

$$a_P p'_P = \sum a_n p'_n - \delta_{mp} \quad (66)$$

where p' is the pressure correction of the main grid, while δ_{mp} is the excess mass leaving control volume CV per unit time. It is defined explicitly as:

$$\delta_{mp} = \rho_E A_{uE}^x u_P^x - \rho_W A_{uW}^x u_W^x + \rho_N A_{vN}^y v_P^y - \rho_S A_{vS}^y v_S^y \quad (67)$$

The finite-difference coefficient a_n stands for a_E , a_W , a_N , and a_S , and can be defined as:

$$a_E = \rho_E A_{uE}^x Du_E^x \quad (68)$$

$$a_N = \rho_N A_{vN}^y Dv_N^y \quad (69)$$

Similar expressions exist for a_W and a_S . The remaining pressure correction equations for CV^x , CV^y , and CV^{xy} are similarly obtained as:

$$a_P^x p'^x_P = \sum a_n^x p'^x_n - \delta_{mp}^x \quad (70)$$

$$a_P^y p'^y_P = \sum a_n^y p'^y_n - \delta_{mp}^y \quad (71)$$

$$a_P^{xy} p'^{xy}_P = \sum a_n^{xy} p'^{xy}_n - \delta_{mp}^{xy} \quad (72)$$

Equations (52)–(57), (66), and (70)–(72) represent a complete set of finite-difference equations for u , v , u^x , v^x , u^y , v^y , u^{xy} , v^{xy} , p' , p'^x , p'^y , and p'^{xy} of the NIMO scheme. The above finite-difference pressure correction equations apply to turbulent flow as well as to laminar flow. However, the momentum finite-difference equations for the turbulent flow have extra source terms than their laminar counterparts. The NIMO higher-order scheme will now be applied to steady laminar flow in pipes and 2D steady laminar flow over a fence.

2.9.1 Steady laminar flow in pipes

Equations (50) and (51) are the governing equations for steady laminar flow in pipes with constant physical properties. The exact analytical solution for this type of flow is a parabolic profile for the fully developed axial velocity distribution along the radial coordinate. It is given as:

$$\frac{u}{u_0} = 2[1 - (r/R)^2] \quad (73)$$

where u_0 is the uniform axial velocity at the pipe entrance, r is the radial distance from the pipe axis, and R is the pipe inner radius. Equation (73) is used here to test the accuracy of the NIMO scheme. Equations (52)–(59), (67), and (70)–(72) represent the NIMO finite-difference equations applicable to the laminar flow in pipes. The pipe radius R is taken as 1.0 cm and the inlet uniform axial velocity is 1.0 m/s. The flowing gas is air at atmospheric pressure and 300 K. For this flow, Reynolds number is 1,300. The dimensionless pipe length (L/D) is 50. The pressure correction normal gradient, for each grid of the NIMO scheme, is equal to zero for the four boundaries of the solution domain. This boundary condition is valid for all the test problems reported below. Zero normal gradients of all dependent variables, at the exit section, are imposed. Again this boundary condition is common to all test problems reported below. The inlet axial velocity is 1.0 m/s, while the radial velocity is equal to zero. At the pipe wall the velocity components are equal to zero. However, at the pipe axis the axial velocity gradient is equal to zero, while the radial velocity itself is equal to zero. For a uniform (80×80) grid, the number of iterations is 1,000, which gives an error less than 0.1 percent in the finite-difference equations. Figure 2.22 depicts the radial profiles of the dimensionless axial velocity for axial distances $(x/D) = 2, 10$, and 20 . The axial velocity radial profiles for the main grid and the x-grid are consistent, showing continuous increase in the centerline velocity. The fully developed axial velocity radial profiles, shown in Figure 2.23 for the grids of the NIMO scheme, are in excellent agreement with the exact analytical solution given by equation (73). In Figures 2.22 and 2.23, u_m is the mean axial velocity. The radial velocity profiles are depicted in Figures 2.24 and 2.25 for dimensionless axial distances of $2, 10, 20$, and 50 . It is interesting to notice that the maximum negative radial velocity occurs at $(x/D) = 2$; then it decreases sharply as the exit section is approached. Figure 2.26 shows the variations in the pressure along the axial distance. The numerical results indicate that the pressure is uniformly declining along the pipe length to overcome the laminar wall shear stresses.

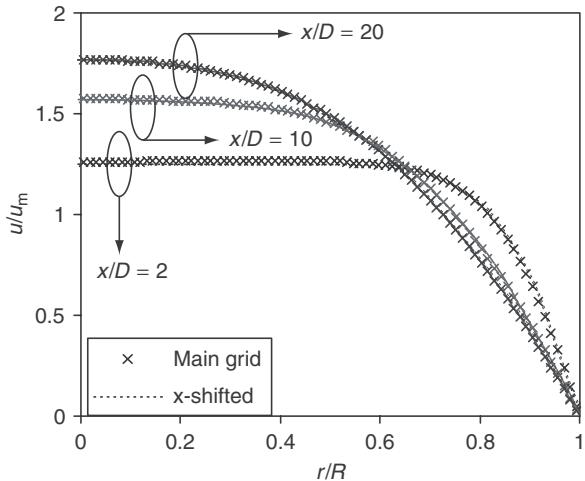


Figure 2.22. Dimensionless axial velocity profiles at different axial locations for the laminar flow in pipes with $\text{Re} = 1,300$.

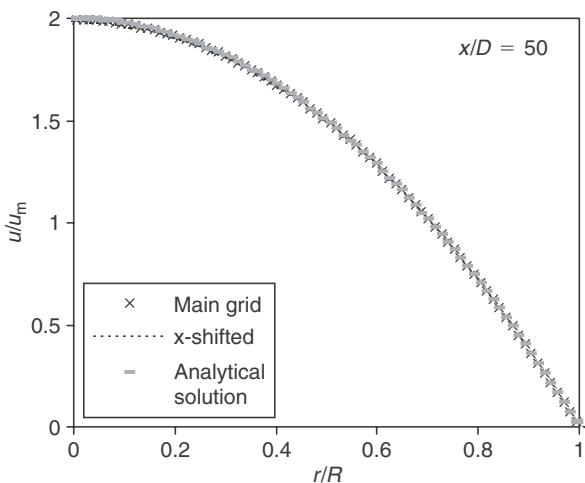


Figure 2.23. Dimensionless axial velocity profile for laminar flow in a pipe with $\text{Re} = 1,300$.

2.9.2 Steady laminar flow over a fence

This is a 2D steady laminar flow over a single fence in a rectangular channel. The finite-difference equations used in the laminar pipe flow problem are also applicable to flow over a fence. Excellent experimental results were obtained, using laser-Doppler anemometry, by Carvalho et al. [13]. The flow geometry used by

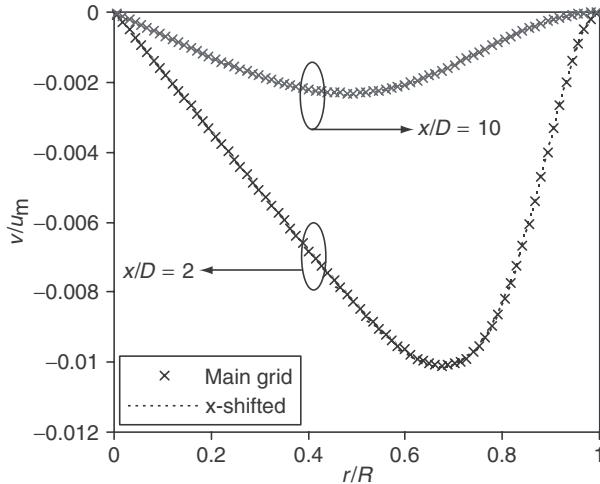


Figure 2.24. Dimensionless radial velocity profiles at different axial locations for laminar flow in pipes with $\text{Re} = 1,300$.

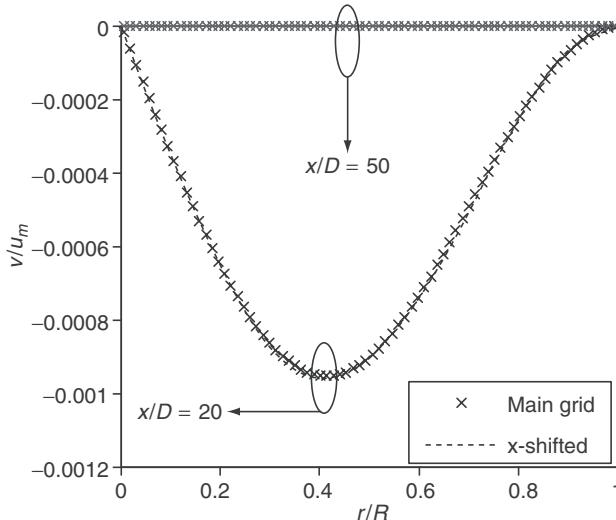


Figure 2.25. Dimensionless radial velocity at different axial locations for laminar flow in pipes with $\text{Re} = 1,300$.

Carvalho et al. [13] is shown in Figure 2.27. The inlet velocity is prescribed as a fully developed profile for laminar flow between two parallel plates, namely,

$$u_i/u_0 = 1.5 \left[1 - \left(\frac{H - 2y}{H} \right)^2 \right] \quad (74)$$

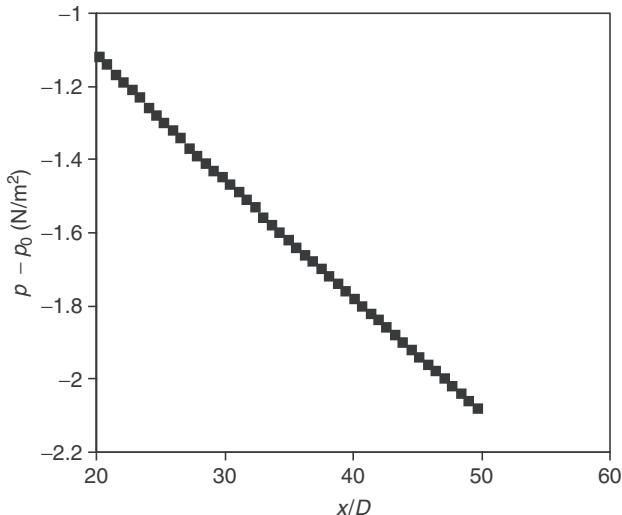


Figure 2.26. Centerline axial pressure profile in the laminar flow in pipes with $\text{Re} = 1,300$.

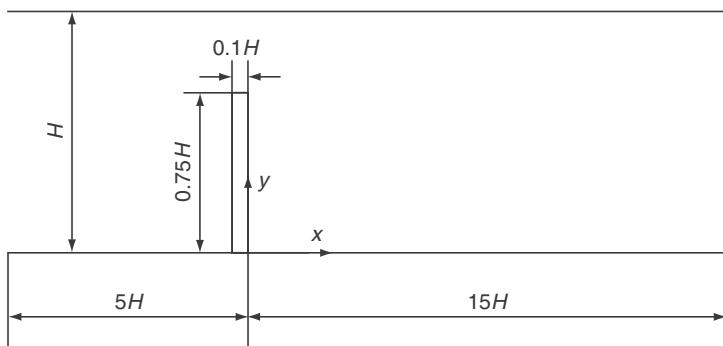


Figure 2.27. Geometry of flow over a fence.

where u_i and u_0 are the velocity along the x -axis at inlet and its mean value, respectively. The distance from the lower plate is denoted by y , while the vertical distance between the two plates is given as H . In the above equation, y ranges between 0.0 and H . The value of height H is equal to 1.0 cm, and u_0 is equal to 0.175 m/s. The flowing gas is air at 300 K. The inlet Reynolds number (Re) is equal to 82.5, which is based on u_0 and the fence height $S (= 0.75H)$. The thickness of the fence is 0.1 cm. The inlet flow velocity is computed from equation (74), while the transverse velocity (v) is equal to zero. Pressure correction normal gradients, u and v , are equal to zero on all solid surfaces of the flow domain. For this laminar flow problems two grids were used, namely, (100×100) and (150×150) . To obtain numerical errors less than 0.1

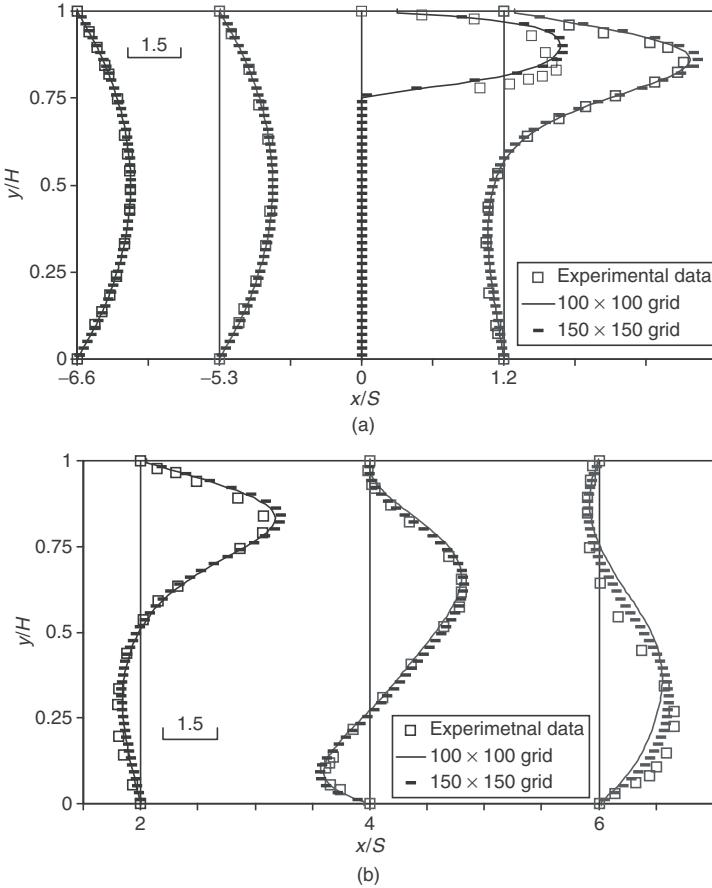


Figure 2.28. Comparison between predictions and measurements of dimensionless velocity along x -axis (u/u_0) [13] for flow over fence, $\text{Re} = 82.5$.

percent, 1,757 and 4,000 iterations are used for the (100×100) and the (150×150) grids, respectively. This shows that the number of iterations for complete conversion is more than doubled in the case of the finer grid compared to the coarse one.

The transverse profiles of the dimensionless velocity component along the x -axis are depicted in Figure 2.28 for $(x/S) = -6.6, -5.3, 0.0, 1.2, 2.0, 4.0$, and 6.0 . The axial distance is measured from the fence. For $(x/S) < 0.0$ the (100×100) and the (150×150) grids gave perfect fit to the experimental data. However, for $(x/S) < 0.0$ the finer grid gave excellent agreement with the experimental data. However, grids finer than (150×150) could not give any higher accuracy. Figure 2.28 shows that the flow direction is downward for $0.0 < (x/S) < 4.0$; then the gas flows upward for higher values of the distance along the x -axis. Recirculation zones are thus created near the lower wall for $(x/S) < 4.0$ followed by recirculation zones at the

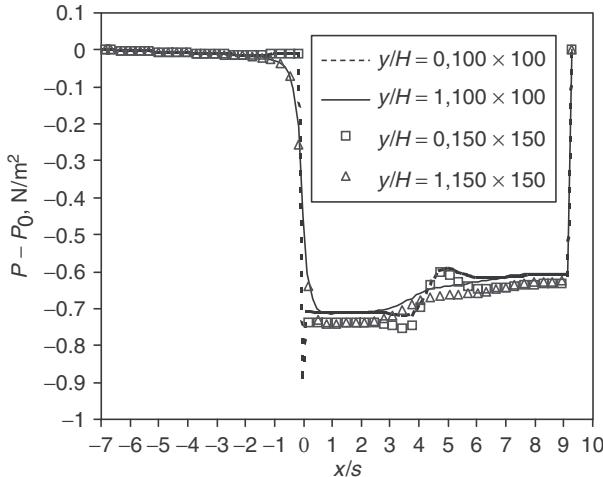


Figure 2.29. Axial pressure profiles near the upper and lower walls for the flow over a fence with 100×100 and 150×150 grids, $Re = 82.5$.

upper wall for higher values of (x/S) . The radial pressure profiles along the x-axis are plotted in Figure 2.29 for the (100×100) and (150×150) grids. The profiles are chosen at (y/H) equal to approximately 0.0 and 1.0, respectively, near the lower and upper walls. It can be seen from this Figure that sudden pressure drop occurs at the fence. However, it is smoother near the upper wall. Moreover, (100×100) grid predicts lower pressure at the fence with respect to the (150×150) grid. It is interesting to notice the sudden increase in pressure as the flow changes its direction in the vicinity of (x/H) ranging between 4 and 5.

2.10 Application of the NIMO Higher-Order Scheme to the Turbulent Flow in Pipes

For constant-density steady turbulent flow in pipes, the modeled time-mean momentum equation can be written as follows:

$$\frac{\partial(\rho u_i u_j)}{\partial x_i} - \frac{\partial}{\partial x_i} \left[(\mu_t + \mu) \frac{\partial u_j}{\partial x_i} \right] = - \frac{\partial p}{\partial x_j} + \frac{\partial}{\partial x_i} \left[(\mu_t + \mu) \frac{\partial u_i}{\partial x_j} - \frac{2}{3} \rho k \delta_{ij} \right] \quad (75)$$

where u_i , u_j , and p are time-mean values. The turbulent viscosity and turbulent kinetic energy are denoted by μ_t and k ; in addition to time-mean momentum equations, the time-mean mass continuity must be included.

$$\frac{\partial(\rho u_i)}{\partial x_i} = 0 \quad (76)$$

The turbulent viscosity can be computed with the help of the two-equation model of turbulence for the kinetic energy of turbulence (k) and its dissipation rate (ε). The governing equation for the kinetic energy of turbulence and its dissipation rate are written here such that they can be applied to the standard k - ε model (Lauder & Spalding [14]) and its low Reynolds number version of Launder and Sharma [15].

$$\frac{\partial}{\partial x_i}(\rho u_i k) - \frac{\partial}{\partial x_i} \left(\frac{\mu_e}{\sigma_k} \frac{\partial k}{\partial x_i} \right) = G - \rho(\varepsilon + D) \quad (77)$$

$$\frac{\partial}{\partial x_i}(\rho u_i \varepsilon) - \frac{\partial}{\partial x_i} \left(\frac{\mu_e}{\sigma_\varepsilon} \frac{\partial \varepsilon}{\partial x_i} \right) = (C_1 f_1 G - C_2 f_2 \rho \varepsilon) \frac{\varepsilon}{k} + \rho E \quad (78)$$

where $G \left(= \mu_t \frac{\partial u_j}{\partial x_i} \left[\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right] \right)$ is the turbulent kinetic energy generation term and is given by Launder and Spalding [14]. The turbulent viscosity μ_t is thus given as:

$$\mu_t = C_\mu f_u \rho k^2 / \varepsilon \quad (79)$$

The high Reynolds number standard k - ε model is specified with the effective viscosity $\mu_e = \mu_t$, $D = E = 0.0$, and $f_1 = f_2 = f_\mu = 1.0$. Moreover, the high Reynolds number k - ε model logarithmic wall functions that prescribe at the wall the shear stress, the generation term (G), and the value of ε are given by Launder and Spalding [14].

However, the low Reynolds number k - ε model of Launder and Sharma [15] takes the solution right to the pipe wall. The common constants between the two versions of the k - ε model are given by Launder and Sharma [15] as $c_\mu = 0.09$, $c_1 = 1.44$, $c_2 = 1.92$, $\sigma_k = 1.0$, and $\sigma_\varepsilon = 1.3$. Moreover, the low Reynolds number additional functions are defined as follows [15]:

$$f_\mu = \exp[-3.4/(1 + R_t/50)^2] \quad (80)$$

$$f_1 = 1.0 \quad (81)$$

$$f_2 = 1 - 0.3 \exp(-R_t^2) \quad (82)$$

$$D = 2(\mu/\rho) \left[\frac{\partial(k)^{\frac{1}{2}}}{\partial r} \right]^2 \quad (83)$$

$$E = 2(\mu/\rho)(\mu_t/\rho) \left[\frac{\partial^2 u}{\partial r^2} \right]^2 \quad (84)$$

where the turbulent Reynolds number R_t is defined as [15]:

$$R_t = \rho k^2 / (\mu \varepsilon) \quad (85)$$

Equations (83) and (84) are given specifically for the turbulent flow in pipes where r is the radial distance and u is the axial velocity.

In the low Reynolds number model, μ_e is defined as $\mu + \mu_t$ and the wall boundary condition is defined as $\partial p'/\partial r = u = v = k = \varepsilon = 0.0$. The boundary condition at the centerline is defined as $\partial \phi/\partial r = v = 0$, where ϕ stands for all dependent variables except v . At the exit section, the axial gradients of all dependent variables are equal to zero. At the inlet section, $u = u_0$ and $v = \partial p'/\partial x = 0$, where u_0 is the inlet uniform axial velocity.

The assessment of the accuracy of the results of the NIMO scheme is achieved with the help of the experimental data of Laufer [16] at $Re = 500,000$ and the semi-empirical relation obtained by Schlichting [17] for a large number of experimental data for the fully developed turbulent pipe flow. The Schlichting 1/7th power law is given as [17]:

$$\frac{u}{u_{CL}} = \left(\frac{R - r}{R} \right)^{1/7} \quad 0 < r < 0.96R \quad (86)$$

where u_{CL} is the fully developed centerline axial velocity.

The numerical predictions for turbulent flow in pipes, using the NIMO higher-order scheme, are shown in Figures 2.30–34. Air at 1.0 bar and 300 K enters the pipe at a uniform axial velocity of 29 m/s. The pipe diameter is 24.7 cm and its dimensionless length (L/D) is equal to 30. For this flow, the Reynolds number is 500,000. For the low and high Reynolds number turbulent models, a (100×100) grid is used. The grid of the high Reynolds number model is uniform, while that for the low Reynolds number model converses toward the pipe wall. For both models the number of iterations approximates 1,000, giving errors less than 0.1 percent in the difference equations.

The developing turbulent flow in pipes is depicted in Figure 2.30 for axial distances (x/D) equal to 2 and 5. The dimensionless axial velocity (u/u_m) numerical results of the main grid are essentially indistinguishable from those of the x -grid, as can be seen from Figure 2.30. The near-wall region is shown in Figure 2.30b, to facilitate the comparison of the wall effect as predicted by the low and high Reynolds number models. The wall effect of the low Reynolds number model propagates into the core of the turbulent flow in pipes, while the high Reynolds number model has its wall effect restricted to $(r/R) > 0.95$ for $(x/D) < 5$. More developing flow numerical results are shown in Figure 2.31 for (x/D) equal to 10 and 20. The high Reynolds number $k-\varepsilon$ model wall effect is still confined to $(r/R) > 0.75$, while the low Reynolds number counterpart has its wall effect all over the turbulent flow for $(x/D) = 20$. The dimensionless radial velocity (v/u_m) profiles are depicted in Figure 2.32, at $(x/D) = 5$ and 20 and at the exit section. The low Reynolds number model predicts higher absolute radial velocities, while both models predict a vanishing radial velocity at the fully developed flow near the exit section of $(x/D) = 30$. The fully developed turbulent time-mean axial velocity radial profiles are depicted in Figure 2.33, for the high and low Reynolds number models. On the same figure, the measured fully-developed dimensionless axial

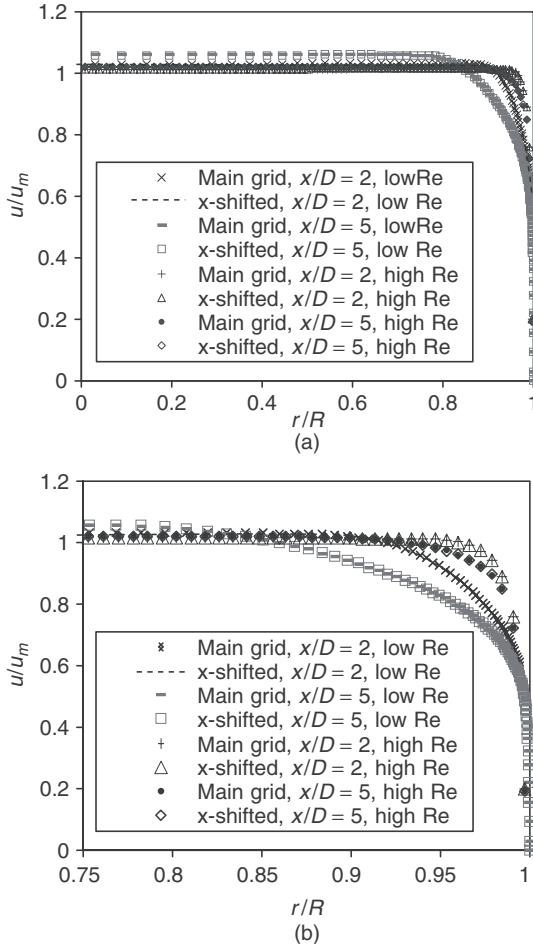


Figure 2.30. Dimensionless axial velocity (u/u_m) at different axial locations for turbulent flow in pipes with $\text{Re} = 500,000$. (a) r/R from 0 to 1. (b) r/R from 0.75 to 1.

velocity [16] and the similar profile that can be calculated by the semi-empirical relation of Schlichting [17] are also plotted. The NIMO scheme predicts excellent agreement with the fully developed axial velocity profile given by the 1/7th power law of Schlichting [17] when the low Reynolds number $k-\varepsilon$ model is used. However, the present numerical results for u/u_{CL} fall nicely between the experimental data of Laufer [16] and the Schlichting's 1/7th power law [17]. On the other hand, the high Reynolds number model poorly predicts the fully developed turbulent pipe flow and hence should not be adopted for this type of flow. Finally, the pressure axial distributions, predicted using the low and high Reynolds number $k-\varepsilon$ models, are shown in Figure 2.34. This figure shows clearly that the absolute axial pressure

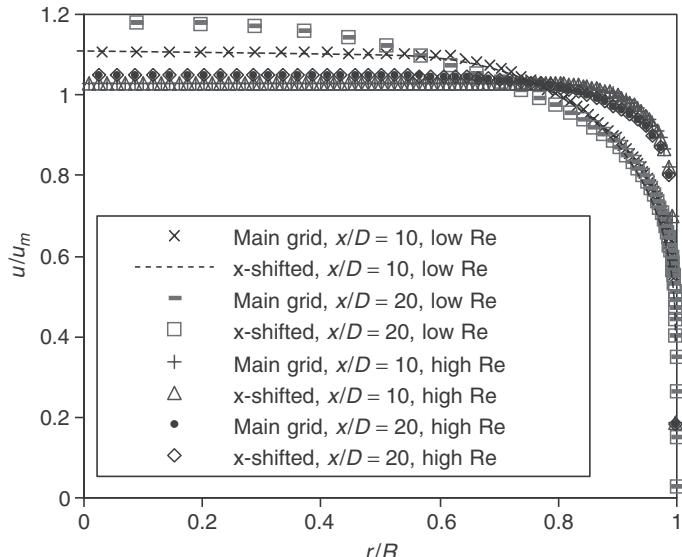


Figure 2.31. Dimensionless axial velocity at different axial locations for turbulent flow in pipes, $\text{Re} = 500,000$.

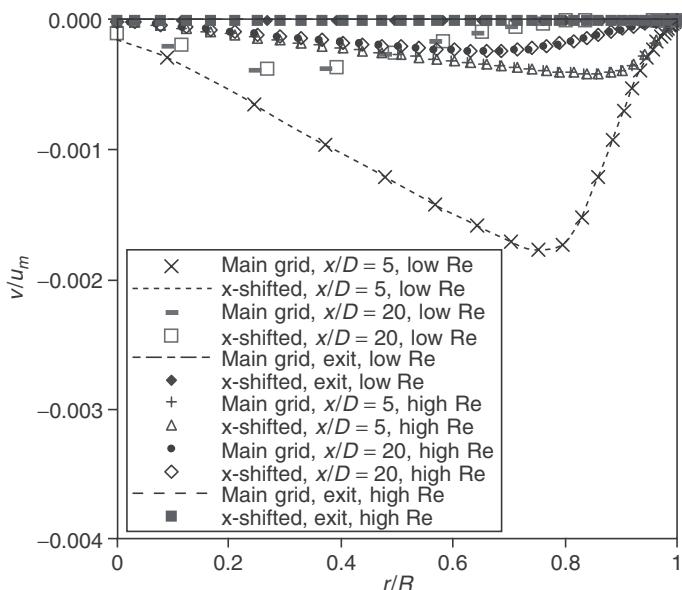


Figure 2.32. Dimensionless radial velocity profiles at different axial locations for turbulent pipe flows, $\text{Re} = 500,000$.

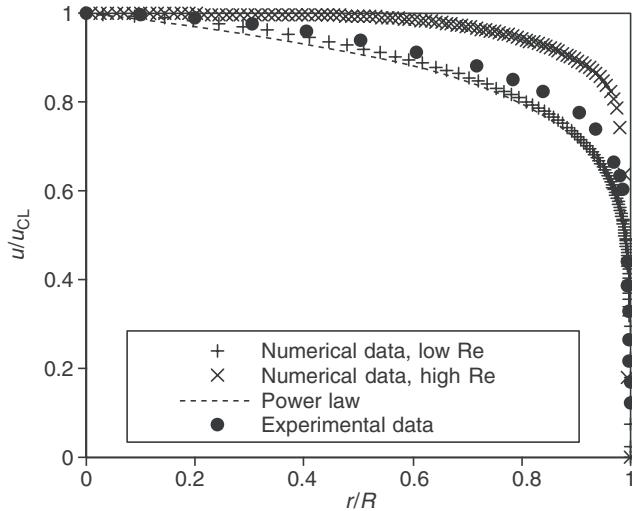


Figure 2.33. Dimensionless time-mean axial velocity for turbulent pipe flows using low and high Reynolds number models, $Re = 500,000$.

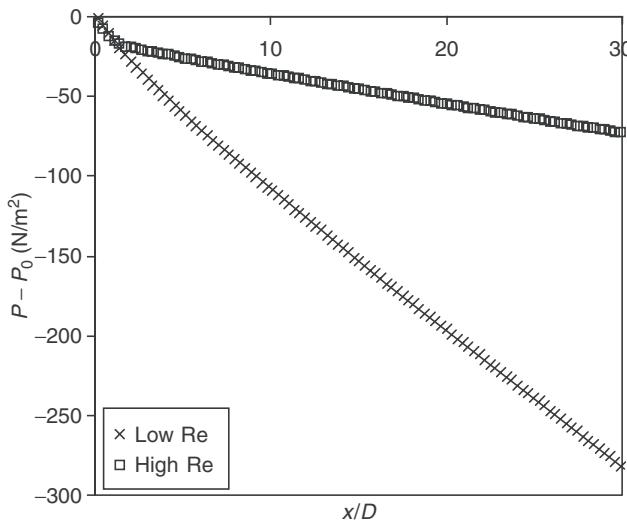


Figure 2.34. Centerline axial pressure profiles for turbulent flow in pipes, $Re = 500,000$.

gradient of the low Reynolds number model is approximately five times higher than that of the high Reynolds number model. Alternatively, the wall shear stress in the low Reynolds number model is five times that of the high Reynolds number $k-\varepsilon$ model.

2.11 Conclusions

This chapter describes a novel numerical procedure—the NIMO finite-difference procedure. The NIMO numerical procedure is free from any interpolations of convective fluxes that normally produce high levels of false diffusion in most known finite-difference schemes. The NIMO system involves four grids in 2D flows. The four grids are located in space such that each grid is displaced midway with respect to one of the remaining grids. The main grid and the xy -grid interconnect indirectly through the x - and y -grids and vice versa. Similar interconnection between the four NIMO grids is valid in the cylindrical-polar solution domain. The percentage sum of the absolute residual errors of the main grid and xy -grid behaves in the same fashion, which is different from the similar trend of the other two grids. The NIMO scheme is applied to two test problems. The first test problem is the oblique flow in the Cartesian coordinates, while the second test problem is represented by a rotating radial flow in an annular sector. The NIMO finite-difference scheme can produce numerical profiles of ϕ that capture the analytical solution of the test problem for 45-degree and 30-degree oblique flows for high Peclet numbers. Moreover, NIMO can essentially capture the analytical solution of the two test problems for square, rectangular, and nonuniform grids. The second test problem is also computed with a very high accuracy, which confirms the validity of the NIMO scheme to the cylindrical-polar coordinates as well as to the Cartesian coordinates. Most of the single-grid numerical procedures utilizing rectangular or nonuniform grids could not reach that level of accuracy due to false diffusion, overshooting, or undershooting. This could be concluded from the comparison of the data of these schemes with the corresponding present numerical results. The NIMO finite-difference equations contain convective terms that slightly resemble the upwind schemes. However, unlike the upwind scheme, the incoming convective fluxes to a particular grid are supplied by two of the remaining three grids of NIMO. A new feature emerges that allows the outgoing fluxes to leave behind positive or negative traces in a particular grid control volume, which is completely ignored in the upwind scheme. These outgoing convective fluxes constitute the incoming convective fluxes to two of the remaining three grids of NIMO. Results obtained for different values of the Peclet number showed that the diagonal core of the test problem is surrounded by narrow strips bounded by $\phi = 0.99$ and 10^{-2} . These narrow strips become essentially nonexistent as the Peclet number is increased to 1,000. The converged numerical solutions of the four NIMO grids for a passive scalar variable at a fixed value of the Peclet number are consistent and identical. The NIMO higher-order scheme has further been tested against laminar and turbulent flow in pipes as well as recirculating flow behind a fence. The laminar flow in pipes is predicted with a very high accuracy for a (100×100) grid. However, the recirculating laminar flow behind a fence can be predicted accurately with a finer (150×150) grid. The turbulent flow in pipes is well predicted when the low Reynolds number k-e model is used.

Nomenclatures

A	surface area of control volume
a	<i>finite-difference coefficients</i>
c	convective transport rates
\hat{c}, \tilde{c}	modified convective transport rates
$c_\mu, c_1, c_2, \sigma_k, \sigma_e$	common coefficients in high and low Reynolds number models
D, E, f_1, f_2, f_μ	coefficients in low Reynolds number model
d	diffusive transport rate
k	turbulent kinetic energy
n	number of iterations
Pe	Peclet number
p'	pressure correction
R_t	turbulent Reynolds number
S_u, S_p	source term coefficients in linear form
u	velocity component along x direction
u_{CL}	fully developed centerline axial velocity
u_k	fluid velocity along coordinate direction x_i
v	velocity component along y direction
x_k	coordinate direction
δx	distance between central node and neighboring node
ε	dissipation rate of turbulent kinetic energy
ϕ	scalar variable
Γ_ϕ	molecular diffusivity of ϕ
μ_e	effective viscosity
μ_t	turbulent viscosity
ρ	density

Superscript

E, W	east and west directions
p	value at center of control volume
S, N	south and north directions
t	turbulent flow
i, j	grid node indices

Superscript

x	x -grid
y	y -grid
xy	xy -grid
*	uncorrected values

References

- [1] Abou-Ellail, M. M. M., Gosman, A. D., Lockwood, F. C., and Megahed, I. E. A. Description and validation of a three-dimensional procedure for combustion chamber flows. *AIAA Journal of Energy*, 2, pp. 71–80, 1978.
- [2] Patankar, S. V. *Numerical Heat Transfer and Fluid Flow*. McGraw Hill: New York, 1980.
- [3] Roache, P. J. *Computational Fluid Dynamics*. Hermosa: Albuquerque, NM, 1972.
- [4] Prince, H. S., Varga, R. S., and Warren, J. E. Application of oscillation matrices to diffusion-convection equations. *Journal of Mathematics and Physics*, 45, pp. 301–311, 1966.
- [5] Leonard, B. P. A survey of finite differences of opinions on numerical modeling of incomprehensible defective confusion equation. *ASME Applied Mechanics Division*, 34, pp. 59–98, 1979.
- [6] Leonard, B. P. Stable and accurate convective modeling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering*, 19, pp. 59–98, 1979.
- [7] Leonard, B. P. A stable, accurate, economical and comprehensible algorithm for the Navier–Stokes and scalar transport equations. In: *Numerical Methods in Laminar and Turbulent Flow*, C. Taylor and B. A. Schrefler, eds., *Proceedings of the Second International Conference*, pp. 543–553. Venice, Italy, 1981.
- [8] Song, B., Liu, G. R., Lam, K. Y., and Amano, R. S. Four-point interpolation schemes for convective fluxes approximation. *Numerical Heat Transfer, Part B*, 35, pp. 23–39, 1999.
- [9] Song, B., Liu, G. R., Lam, K. Y., and Amano, R. S. On a higher-order bounded discretization scheme. *International Journal for Numerical Methods in Fluids*, 32, pp. 881–897, 2000.
- [10] Song, B., Liu, G. R., and Amano R. S. Applications of a higher-order bound numerical scheme to turbulent flows. *International Journal for Numerical Methods in Fluids*, 35, pp. 371–394, 2001.
- [11] Raithby, G. D. Skew upstream differencing schemes for problems involving fluid flow. *Computer Methods in Applied Mechanics and Engineering*, 9, pp. 153–164, 1976.
- [12] Verma, A. K., and Eswaran, V. Overlapping control volume approach for convection-diffusion problems. *International Journal for Numerical Methods in Fluids*, 23, pp. 865–882, 1996.
- [13] Carvalho, G. M., Durst, F., and Pereira, C. F. Predictions and measurements of laminar flow over two-dimensional obstacles. *Applied Mathematical Modeling*, 11, pp. 23–34, 1987.
- [14] Launder, B., and Spalding, D. The numerical computation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering*, 3, pp. 269–289, 1974.
- [15] Launder, B., and Sharma, B. Application of the energy-dissipation model of turbulence to the calculation of flow near a spinning disc., *Letters in Heat and Mass Transfer*, 1, pp. 131–137, 1974.
- [16] Laufer, J. The structure of turbulence in fully developed pipe flow. NACA Report no. 1174, 1954.
- [17] Schlichting, H. *Boundary-Layer Theory*, Chapter 20, McGraw-Hill: New York, 1979.

3 CFD for industrial turbomachinery designs

C. Xu and R. S. Amano

University of Wisconsin, Milwaukee, WI, USA

Abstract

An implicit method was introduced to solve the viscous turbulent flow in a time-efficient way. This chapter presents how to use the numerical method for applying the airfoil blade design improvements in turbomachinery. This step includes blade aerodynamic design system with optimization approaches. The design method developed by the authors are introduced that employs a discrete vortex method to calculate the blade pressure distributions after the airfoil section was designed, and a fast N-S solver developed in this study was used to further check the blade performance. With the approaches described in this chapter, the whole blade aerodynamic design procedure can provide a fast way to design the blade profile that meets the blade design requirements. The present Navier-Stokes (N-S) code can well handle both subsonic and supersonic flows, which can be further connected to an optimization code to performance three-dimensional blade design optimizations.

Keywords: CFD, Compressor, Finite-volume method, Implicit method, Optimization, Turbomachinery

3.1 Introduction

3.1.1 Computational methods in turbomachinery

Flows in turbomachinery are among the most complex flows encountered in fluid dynamic practices as in most cases, they are three dimensional, with all three flow regimes (laminar, transition, and turbulent) with flow separations. Flows may be incompressible, compressible, subsonic, transonic, or supersonic; some turbomachinery flows include all of these flow regimes. Equations are strongly coupled and complex boundary conditions are often unavoidable. A large number of flow and thermal parameters are encountered: Reynolds number, Mach number, rotational speed, Prandtl number, and flow incidence, with highly complex geometrical parameters. This complex flow and geometrical variables dictate the nature of governing equations and flow solvers to be used.

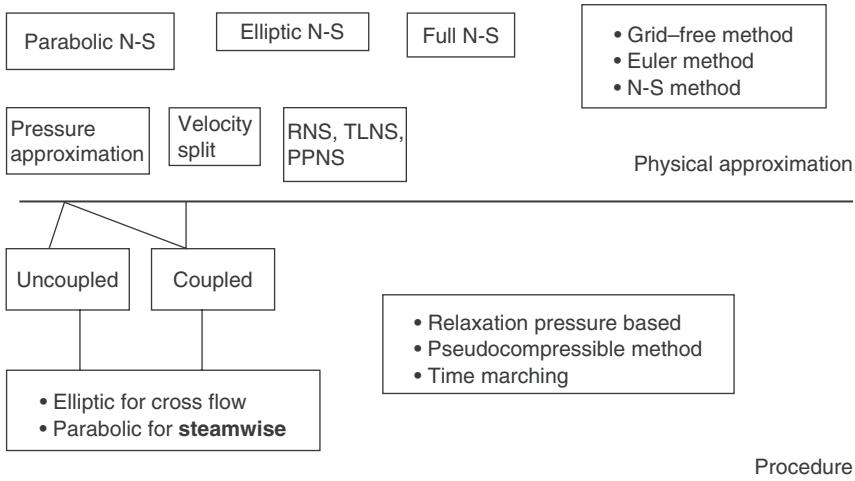


Figure 3.1. Classifications of governing equations.

With the rapid development of computer hardware, the use of computational fluid dynamics (CFD) in turbomachinery design and analysis has been ever increasing. Aerodynamic processes associated with the flow of fluid through turbomachines pose one of the toughest challenges to the computational fluid dynamicists. Although with the recent development of computer performance scientists and engineers can look more and more flow phenomena in turbomachine designs, applications of full three-dimensional multistage simulations in the aerodynamic design are still a challenge and so is, the design of effective schemes that can be used for turbine design [1–6].

There are several types of approximations in the numerical solution of governing equations for turbomachinery flow simulations. Governing equations are approximated depending on the physics of flowfield. Further approximations are in numerical discretization and technique in solving these equations. The classifications for governing equations are shown in Figure 3.1. The numerical procedures classified according to the physical approximations made are inviscid flows, parabolized Navier–Stokes techniques, and thin-layer Navier–Stokes and full Navier–Stokes solutions. According to the flow equations to be solved, the numerical method can be categorized into three types: i.e. grid-free vortex method [7–14], incompressible Euler equation and Navier–Stokes simulation method [2–6], and compressible Euler equation and Navier–Stokes simulation method [14–36]. Discussions of the numerical methods here are mainly on the Navier–Stokes equations. Brief discussions of the grid-free vortex method and incompressible simulation method are also provided. The main focus will be on the discussions of the compressible simulation method with emphasis on the Navier–Stokes method, which is directly related to the present study.

3.1.2 Grid-free vortex method

Vortex methods are based on the frequency-domain source-doublets and vortex panels that can predict steady and unsteady flows in turbomachines. In this approach, airfoils of a turbomachine are discretized in two several vortex panels and source-doublets to represent boundaries. Vortex methods can handle not only steady flows but also unsteady flows [7–14]. Most of these methods are based on the inviscid assumption. The vortex methods are still limited in irrotational flow, and they still have some difficulties in handling heat transfer problem. A grid-free vortex method can be adapted and extended to use as part of blade design procedure. In the vortex method, the rotor/stator stage is decoupled into two separate blade rows. Each blade row interacts only with the immediate downstream or upstream of a blade row. The influences from other blade rows are assumed to be much smaller than that from the blade from the immediate neighboring blade rows. The wake profile can be an analytical or measured time or spatially dependent velocity distribution. The velocity distributions are generally not the simple harmonic type. Harmonic amplitudes and related phase angles of this nonsimple harmonic wake-field can be obtained through a Fourier decomposition of the velocities. The wakes of the blades are usually calculated based on some empirical wake models. The unsteady response on the downstream blade row is calculated in a typical blade-wake interaction manner. This method can predict the interactions between blades and blades as well as blades and wake. The panel method can predict the pressure distribution accurately. The computational time using this method is more than 100 times faster than a typical Navier–Stokes solution. This method can be coupled to a boundary layer code to predict viscous effects and losses. The main limitations of this method are based on the assumption that flow is irrotational and incompressible. The three-dimensional turbine flows are almost invariably rotational, and the existence of potential flow is an exception.

The detailed formulation is described in Refs. [7–10]. The flow around a turbine airfoil can be represented by vortex sheets and/or source sheets of unknown strength clothing the whole airfoil. The location of the sheet is measured clockwise around the body perimeter from the trailing edge. The airfoil profile is divided into N planar elements denoted by (x_q, y_q) (where $q = 1, 2, \dots, N$) with elements 1 and N being adjacent to the trailing edge. The calculation assumes that a uniform vortex distribution of an initially unknown vorticity is placed on each element q . The Biot–Savart law provides a means of calculating the velocities in the flow field. The boundary condition on the airfoil surface is such that the resulting flow must be parallel to the body surface at all control points.

The condition of a zero internal tangential velocity at the p th collocation point (where $p = 1, 2, \dots, N$) is given by

$$(Q + Q_\infty) \cdot \tau|_p = 0 \quad (1)$$

where Q is the velocity induced by each vortex element. While the free-stream tangential velocity component can be calculated directly, it can be moved to the

right-hand side of equation (1) and be given as follows:

$$(Q \cdot \tau)|_p = -(Q\infty \cdot \tau)|_p = -U\infty \cos \beta_p - V\infty \sin \beta_p \quad (2)$$

where $U\infty$ and $V\infty$ represent the two components of uniform stream velocity parallel to the x - and y -axes, respectively. β_p is the slope of the body profile and is defined by

$$\beta_p = \tan^{-1} \left(\frac{y_p - y_{p-1}}{x_p - x_{p-1}} \right) \quad (3)$$

Let Δl_q be the length of the q th element. On using the boundary condition, equation (1) can be expressed as a set of linear equations

$$\sum_{q=1}^N K_{pq} \gamma q = -U\infty \cos \beta_p - V\infty \sin \beta_p \quad (4)$$

where K_{pq} denotes the coupling coefficients linking p and q and is given by

$$K_{pq} = \frac{\Delta l_q}{2\pi} \frac{[(y_p - y_q) \cos \beta_p - (x_p - x_q) \sin \beta_p]}{[(y_p - y_q)^2 + (x_p - x_q)^2]} \quad (5)$$

After the vortex strength of each element is calculated, the velocity vector field and velocity potential field can be obtained. The calculated velocity potential and velocity can then be used to obtain the pressure and the isentropic Mach number distribution. To validate the code, a NACA 0012 airfoil with 4-degree incidence was used.

3.2 Numerical Methods for Incompressible Flow

From the numerical method point of view, it is needed to point out that the techniques typically used for the solution of Euler and Navier–Stokes equations are the same, even though there are some differences in the behavior of the technique with and without the viscous terms. The techniques typically used by both Euler and Navier–Stokes equations are discussed in the following sections. To predict and improve the performance of turbomachine, it is important to understand the flow and heat transfer through the turbine blade and blade passages from the standpoint of the design of blade geometry. For the flow inside the turbine stages from the first stage to the last one, the flow character may change from incompressible potential flow to compressible flow. The incompressible solution is also an important part of the turbomachinery flow simulations. In most of the early studies [6,10], time-average incompressible flow equations were considered. In those methods the first and second derivative terms did not contain cross-derivatives that can be treated

implicitly. The high-order artificial dissipation terms can be used in the calculation in order to increase the accuracy when the solutions meet the stability requirement. To calculate the pressure, those schemes need to incorporate the pressure correction equations.

Pressure-based methods are one of the major methods for incompressible flow analysis. In this method, the assumed pressure is updated using an auxiliary equation for pressure, and a multipass procedure is adapted to help converge both the pressure and the velocity field. Most of the pressure-based methods are extensions of the single-pass pressure correction methods. In this technique, a separate equation for pressure correction is developed for both the bulk pressure correction and the pressure correction in the transverse momentum equation, relating the source term in the Poisson equation to the pressure correction in the velocity field through the continuity equation (10). The pressure correction requires a staggered grid to eliminate oscillations in the solution of the pressure correction equation. Both finite-difference and finite-volume technique can be used in these methods. Application of these techniques to turbomachinery flows and subsequent modifications to improve accuracy and efficiency have been carried out. The author has modified and improved convergence of the pressure correction method [10]. The pressure-based methods are one of the most sophisticated methods. They are highly recommended for incompressible turbomachinery flow simulation.

3.3 Numerical Methods for Compressible Flow

For the numerical method for compressible flow, the time-marching technique is one of the most important methods. The principle of a time-marching method is to consider the solution of a stationary problem as the solution after a sufficiently long calculation time of the time-dependent equations describing the particular problem. The attraction of the time-marching method is the ability to compute mixed subsonic and supersonic flows. The equations may be solved by either finite-difference or finite-volume form. In the former, it is usual to transform the computational domain into a uniform rectangular grid and to express the derivatives of the flow variables in terms of values at the nodes of this grid. In the finite-volume method the equations are regarded as these for the conservation of mass, energy, and momentum applied to a set of interlocking control volumes formed by a grid in the physical plan. When equations are solved in this way, it is easier to ensure conservation of mass and momentum than in the differential approach, but numerical schemes are necessary to ensure stability.

The advantages of the time-marching methods are as follows: (a) the same code can be used for the solution of all flow regimes: subsonic to supersonic, viscous or inviscid equations; (b) the technique is easy to implement and vectorize to take an advantage of supercomputing capabilities. The equations are solved as a coupled system. The scalar variables (pressure, velocities, density, enthalpy, entropy, and temperature) are solved simultaneously; (c) the code is flexible so that it can employ

both external and internal flow calculations; (d) for complex flow geometry and flow fields, the technique is comparable in other methods. The major disadvantages are as follows: (a) since the technique is time iterative and often requires very small time steps, the computational efficiency is low. This shortage stimulates the author to develop a new kind of economical time-marching algorithm; (b) the technique is not suitable for incompressible flows. This also stimulates the author to solve the eigenvalue stiffness problem.

Since the flows in turbomachinery components contains both incompressible and compressible parts, the development of the methods that can be used for both incompressible and compressible flows is very important. The pressure-based methods solve a Poisson equation for pressure, and the momentum equations in an uncoupled manner by iterating until a divergence-free flow field is satisfied. On the other hand, it is evident from the material presented in the earlier section that very efficient and accurate algorithms have been developed for solving hyperbolic systems governing compressible flow using time-iterative methods. Many attempts have been made to combine these two methods and hope to form a kind of new method that can be used in both incompressible and compressible flows.

Two methods have been proposed. One involves recasting the incompressible flow equation as hyperbolic systems, which is called pseudo-compressibility technique. The first method is to premultiply the time derivative by a suitable matrix and the other is to use a perturbed form of the equations in which specific terms are dropped such that the physical acoustic waves are replaced by pseudo-acoustic modes. This technique is called preconditioning method [6].

Pseudo-compressibility technique is based on making the governing equations behave like a hyperbolic system through adding the time derivative of a pressure term. In the steady state, these equations and solutions are exactly those of steady incompressible flow, even though the temporal solutions are not accurate. The continuity equation after modification introduces spurious pressure waves of finite speed into the flowfield through the pseudo-compressibility parameter. Most researchers linearized the equations using a truncated Taylor series and discretized them by using a two-point central difference in the spatial directions. The most popular solution techniques are the standard implicit approximate factorization schemes. The pseudo-compressibility technique has been used in the turbine blade flow calculation. However, the pseudo-compressibility parameter is difficult to determine. It is strongly influenced by the flow structure.

The preconditioning method, on the other hand, is based on modifying the time-dependent Navier–Stokes equations. The preconditioning matrix was added in the time derivative term. The basic idea of the preconditioning procedure is to start with the time derivatives expressed in terms of the viscous dependent variables that appear in the diffusion terms. The procedures modify the eigen structure of the inviscid components of the set of the Navier–Stokes equations such that the condition number remains bounded as the freestream Mach number becomes small. Provided that the numerical discretization is modified to reflect the new eigen system, the use of time-derivative preconditioning can allow a single algorithm to provide accurate, efficient solutions to the Navier–Stokes equations at all flow speeds.

For obtaining the stable scheme, most of the researchers used the implicit scheme to yield a promising method for rapid calculations of fully coupled flows at all speeds. However, the selection of the preconditioning matrix is according to experience and the flow characteristics. The limitations of current methods encourage us to investigate the new methods. Authors [2 and 3] proposed a timing matching method through modifying the governing equations to prevent eigenvalue stiffness when it uses in incompressible flow range.

There are two classes of methods for solving the time-dependent equations: explicit and implicit. Explicit methods, where spatial derivatives are evaluated using the known conditions at the old time level, are simpler and more easily vectorizable. The implementation of the boundary conditions is straightforward. The coding can be easily extended to include a time-dependent calculation. Their major disadvantages lie in the conditional stability dictated by courant-Friedrichs-lewy limit. The convergence is usually slower and requires more computational time than an implicit method. For implicit methods, on the other hand, the unknown variables are derived from a simultaneous solution of a set of equations. Implicit methods usually allow larger time steps and faster convergence, and are attractive for both steady and unsteady flows. Both explicit and implicit schemes are widely employed in the computations of turbomachinery flows. However, both of these methods are still under development.

The most widely used explicit methods in turbomachinery flow are Lax-Wendroff scheme, MacCormack scheme, and Runge–Kutta scheme [6]. The Runge–Kutta scheme allows larger time step than that with other schemes. Artificial dissipation must be purposefully added to these schemes, usually after each stage in the updating procedure. All of these schemes provide inherent artificial dissipation due to truncation error. The most widely used implicit techniques in turbomachinery flow are approximate factorization and upwind schemes. The literature shows that operational CFL number in the order of 10 for viscous turbomachinery flow calculations provided the best error damping properties although linear stability analysis provided that the schemes are unconditionally stable in two dimensions [1–3]. The implicit scheme enables large time steps but may introduce factorization error. The results for a turbine cascade indicate that all techniques provide nearly identical pressure distribution. The major differences lie in the CFL number and CPU time for convergence. Both explicit and implicit techniques are widely used, with explicit techniques more commonly used in industry. Both techniques can incorporate local time stepping. The maximum time step for a four-stage Runge–Kutta method is $CFL = 2.8$, and typical CFL numbers for the implicit lower upper scheme and alternating direction implicit scheme are between 5 to 10, [6]. Therefore, the choice of explicit versus implicit technique and the time accuracy of a given algorithm may have less to do with the accuracy of the predicted results than the nature of a numerical damping and an application grid.

In the turbomachinery flow field simulation, most researchers [20–32] focused on the flow field analysis. In the flow and heat transfer analysis, the governing equations may be solved in either finite-difference or finite-volume form. The finite-difference scheme usually provides a good convergent rate. The finite-volume

scheme provides a very simple and automatic conservation of mass and momentum scheme, and it also has good physical properties because the scheme is obtained from working in a physical grid. The argument as to whether finite-difference or finite-volume schemes are preferable is not resolved, and both types are still widely used. Either scheme is written in the form of finite-difference scheme. The finite-difference equations used to simulate flow and heat transfer must be numerically stable, spatially and temporally accurate and efficient, conservative so that flow discontinuities can be removed from every grid without any distortion, and easily applicable in generalized coordinates.

Aerodynamic designers have been using the CFD approach as one of the design tools to generate engineering design data. The potential of CFD approaches to revolutionize a turbomachinery analysis is exemplified by impressive calculations such as those of flows around turbine stators and rotors. A general method for CFD in the design of turbomachinery, which would include turbulence viscous effects with minimum approximations and high efficiencies, could provide a major tool for both researchers and design engineers to improve turbine performance and durability. One of the motivations of this part has led to a major effort in developing cascade flow analysis of various degrees of sophistication. Despite significant gains in the computational speed of modern computers, most numerical schemes still require prohibitive amounts of computational time to obtain solutions of the Navier-Stokes equations. Despite the challenges of the three-dimensional complicated flow simulation, a two-dimensional flow simulation is still an interesting topic of a research scientist [1,4].

Although considerable progress has been made in the past 20 years in the numerical simulation of steady two-dimensional turbine cascade flows, these numerical results were still in general not reliable enough [1]. Further improvement is still needed in the turbulence modeling, grid generation, and efficiency of the numerical scheme. The fact is that turbulence models themselves have not always performed well. The choice of the numerical method might also be a crucial factor for the successful implementation of a numerical scheme. The present study demonstrates the efficiency of the numerical scheme and potential usage of this scheme to solve cascade flow and heat transfer problems.

Time-marching algorithms [2,14] have been used primarily in gas turbine cascade flow and heat transfer analysis. The basic principle of a time-marching method is to consider the solution of a stationary problem as the solution after sufficiently long calculation time of the time-dependant equations. The computation starts with a rough perturbation that develops under certain boundary conditions until it reaches a convergent state. In this approach, the governing equations are replaced by a robust time-difference approximation with which steady or time-dependent flows of interest can be solved. At each time level, a system of algebraic equations is solved. The time-marching algorithms can be categorized into two types: explicit and implicit schemes. An explicit method, in general, is easier to program and to vectorize, allowing incorporation of boundary conditions and turbulence models in a simpler way than an implicit method. A main shortcoming of the explicit method is its susceptibility to numerical stability, which constrains the time-step size rather

severely. The implicit scheme, on the other hand, is generally unconditionally stable, so that a relatively large time step can be adapted. In addition to these algorithms, there exists a family of hybrid algorithms, which is widely used in the cascade-flow simulations. The hybrid algorithms possess the positive features of the explicit and implicit algorithms, providing a relatively rapid convergence process and having a less restricted stability constraint. The essence of this algorithm is the use of explicit and implicit finite-difference formulae at alternative computational mesh points. However, the nature of the basic formulae for the hybrid algorithm remains the same as for the traditional implicit and explicit schemes.

In viscous calculations, dissipating properties always present due to the existence of diffusive terms. Away from the shear layer regions, however, the physical diffusion is generally not sufficient to prevent the numerical oscillation of the schemes. Thus, the artificial dissipation terms are usually used in order to stabilize and permit a higher CFL number in most of the interactive schemes for the compressible flow computation. As is commonly recognized, the artificial dissipation will influence the accuracy of the numerical results. It has been shown that essentially grid-converged solutions for the high-speed viscous flows over aerodynamic shapes can be obtained if sufficiently fine meshes for the computation are provided. However, the computation based on fine meshes is very time consuming, making it difficult to assess the numerical accuracy of grid-converged solutions. The techniques to properly treat dissipation terms are still an important topic of the computational fluid. An attempt was made by Turkel and Vatsa [15] to improve the accuracy of the solutions on a given grid in order to reduce the required number of grid points for obtaining a specified level of accuracy. The essential mechanism used in their method [15] is to reduce the level of the artificial viscosity by replacing the scalar form of the artificial viscosity with a matrix form. It has been shown that the numerical accuracy of the Navier–Stokes solutions is improved through the use of the matrix-valued dissipation model. In this chapter, the dissipation terms were not treated through modifying convective fluxes as in traditional methods. Instead, the dissipation terms were incorporated into the time-derivative terms to form a new time discretization scheme to reduce the level of artificial viscosity. And also in this way, the eigenvalue-stiffness problem associated with the time-derivative term can be prevented when the calculations are performed in a low Mach number flow. This characteristic can partly control the effect of eigenvalue stiffness, which is well known on the convergence of both explicit and implicit schemes. Furthermore, the present scheme is a hybrid scheme that combines the advantages of the implicit and explicit schemes, and is a second-order time and spatial derivative scheme that allows more accurate calculation. Because the higher CFL number can be used with the present method, it is more economic than the ordinary implicit scheme.

The effect of eigenvalue stiffness on the convergence of both explicit and implicit schemes is well known, and normally two distinct methods have been suggested for controlling the eigenvalues to enhance convergence. There are two kinds of methods that can solve this problem. One of the methods to overcome this problem is to premultiply the time derivative by a suitable matrix, and the other is to use a perturbed form of the equations in which specific terms are dropped such that

the physical acoustic waves are replaced by pseudo-acoustic modes. The present method follows the first idea and incorporates the matrix-valued dissipation terms in time-derivative terms. These terms prevent the eigenvalues of the time-derivative terms from becoming stiff when the calculations are performed in the flow where the Mach number is relatively small. This characteristic can partly control the effect of eigenvalue stiffness that is well known on the convergence of both explicit and implicit schemes. Furthermore, the present scheme employs a hybrid algorithm that combines the advantages of the implicit and explicit schemes. The present scheme also processes a spectral radio technique to simplify the calculation and avoid the approximate factorization, thus increasing the stability. This new method is a second-order time and spatial derivative scheme permitting a high CFL number.

3.4 Governing Equations for Two-Dimensional Flow

The two-dimensional time-dependent Reynolds-averaged Navier–Stokes equations can be written as:

$$Q_t + E_x + F_y = R_x + S_y \quad (6)$$

where

$$Q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix}, \quad E = \begin{bmatrix} \rho u \\ p + \rho u^2 \\ \rho u v \\ (e + p)u \end{bmatrix}, \quad F = \begin{bmatrix} \rho v \\ \rho u v \\ p + \rho v^2 \\ (e + p)v \end{bmatrix}, \quad R = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \beta_x \end{bmatrix}, \quad S = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \beta_y \end{bmatrix} \quad (7)$$

and where

$$\begin{aligned} \tau_{xx} &= 2\mu u_x + \lambda(u_x + v_y) \\ \tau_{xy} &= \mu(u_x + v_y) \\ \tau_{yy} &= 2\mu u_y + \lambda(u_x + v_y) \\ \beta_x &= u\tau_{xx} + v\tau_{xy} + \gamma\mu P_r^{-1} e_x \\ \beta_y &= u\tau_{xy} + v\tau_{yy} + \gamma\mu P_r^{-1} e_y \\ e &= p/[\rho(\gamma - 1)] \\ \lambda &= -2\mu/3 \\ \mu &= \mu_t + \mu_l \end{aligned}$$

The equation of state for a perfect gas is:

$$p = \rho RT \quad (8)$$

And the definition of the static enthalpy supplementing the equations of motion can be written as:

$$h = C_p T \quad (9)$$

Because the turbine cascade flow always accompanies high temperature, the flow properties change with the temperature variation. In this calculation, Sutherland's Law is used to express the relation of molecular viscosity and temperature:

$$\mu = \mu_0 \left[\left(\frac{T}{T_0} \right)^{1.5} \frac{T_0 + S}{T + S} \right] \quad (10)$$

where $S = 100$ K for air and the variation of thermal conductivity with temperature is determined by:

$$k = k_0 \left[\left(\frac{T}{T_0} \right)^{1.5} \frac{T_0 + S'}{T + S'} \right] \quad (11)$$

where $S' = 194$ K for air. Making the variable transformation for the governing equations from physical domain (x, y) to computational domain (ξ, η) , we have

$$\xi = \xi(x, y) \quad (12)$$

$$\eta = \eta(x, y) \quad (13)$$

The Navier–Stokes equation can be written as:

$$\overline{Q_t} + \overline{E_\xi} + \overline{F_\eta} = \overline{R_\xi} + \overline{S_\eta} \quad (14)$$

where

$$\begin{aligned} \overline{Q} &= \frac{1}{J} \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix}, \quad \overline{E} = \frac{1}{J} \begin{bmatrix} \rho \bar{u} \\ p\xi_x + \rho u \bar{u} \\ p\xi_y + \rho u \bar{v} \\ (e + p) \bar{u} \end{bmatrix}, \quad \overline{F} = \frac{1}{J} \begin{bmatrix} \rho \bar{v} \\ p\eta_x + \rho u \bar{v} \\ p\eta_y + \rho v \bar{v} \\ (e + p) \bar{v} \end{bmatrix} \\ \overline{R} &= \frac{1}{J} \begin{bmatrix} 0 \\ \tau_{xx}\xi_x + \tau_{xy}\xi_y \\ \tau_{xy}\xi_x + \tau_{yy}\xi_y \\ \beta_x\xi_x + \beta_y\xi_y \end{bmatrix}, \quad \overline{S} = \frac{1}{J} \begin{bmatrix} 0 \\ \tau_{xx}\eta_x + \tau_{xy}\eta_y \\ \tau_{xy}\eta_x + \tau_{yy}\eta_y \\ \beta_x\eta_x + \beta_y\eta_y \end{bmatrix} \end{aligned}$$

where the contravariant velocities are given by:

$$\bar{u} = u\xi_x + v\xi_y$$

$$\bar{v} = u\eta_x + v\eta_y$$

Equation (14) can be written in the similar form to equation (6) as:

$$Q_t + E_\xi + F_\eta = R_\xi + S_\eta \quad (15)$$

A satisfactory turbulence model should meet the requirements of accuracy and compatibility [1–5]. Although the flow around the turbine cascade is extremely complicated, the boundary layers comprise three regions: boundaries near the suction side, those near the pressure side, and the wake region behind the trailing edge. In most of the blade shapes the curvature of the suction side is fairly low and the pressure and velocity gradients are not very large compared to the boundary-layer thickness. This kind of turbulent boundary layer can be simulated using zero-equation turbulence models just as well as with a more complicated and less efficient one- or two-equation models of turbulence. For the wake region near the trailing edge, there are no widely used and successful models to simulate this complex flow. The Baldwin–Lomax model [32] is very compatible and has several good features, such as its usefulness in separated flows with small separation, its relatively smooth continuous transition of the length scale of the boundary layer into the length scale of the far wake, and its ability to accurately predict the wall effect near the trailing edge. For this reason, the Baldwin–Lomax turbulence model was employed in this study to handle the turbulent flow around the turbine cascade flows.

3.5 Decomposition of Flux Vector

3.5.1 Governing equations

In general, there are two ways to split the flux. One is the faster convergent scheme called the Flux Vector Splitting (FVS), and the other is a more accurate scheme called the Flux Difference Splitting (FDS). Of these, FDS has been shown to be the more accurate one. One-fifth to one-half as many points were required to resolve a shear layer using fluxes in a first-order scheme compared to using other fluxes in second- or even third-order schemes. However, in the FDS scheme, the exact Jacobian of the Roe matrix is expensive to compute, whereas approximated Jacobians have observed to give poor convergence compared to FVS.

The basic Roe scheme could produce entropy-violating solutions, and it has a slower convergence compared to FVS. In the present study, the FVS scheme is used to evaluate the Jacobians in matrix equations. The flux vector can be decomposed into two subvectors according to the positive and negative eigenvalues. Through a forward straight manipulation, one can compute the Jacobians $A = \partial E / \partial Q$ and $B = \partial F / \partial Q$ through the following expressions:

$$E = AQ \quad (16)$$

$$F = BQ \quad (17)$$

Equation (15) can be rewritten as

$$Q_t + AQ_\xi + BQ_\eta = R_\xi + S_\eta \quad (18)$$

Note that E and F are homogeneous functions of degree 1 in “ Q .” The Jacobians have the following transformation relations:

$$P^{-1}AP = \Lambda_A \quad (19)$$

$$P^{-1}BP = \Lambda_B \quad (20)$$

where Λ_A and Λ_B are the diagonal matrices and can be written as:

$$\begin{aligned} \Lambda_A &= \text{diag}[\tilde{u}, \tilde{u}, \tilde{u} + aL(\xi), \tilde{u} - aL(\xi)] \\ &= \text{diag}[\lambda_{E1}, \lambda_{E2}, \lambda_{E3}, \lambda_{E4}] \end{aligned} \quad (21)$$

$$\begin{aligned} \Lambda_B &= \text{diag}[\tilde{v}, \tilde{v}, \tilde{v} + aL(\eta), \tilde{v} - aL(\eta)] \\ &= \text{diag}[\lambda_{F1}, \lambda_{F2}, \lambda_{F3}, \lambda_{F4}] \end{aligned} \quad (22)$$

where λ 's are the eigenvalues of flux vectors E and F , a denotes the speed of sound in the physical domain, and

$$\tilde{u} = u\partial\xi/\partial x + v\partial\xi/\partial y \quad (23)$$

$$\tilde{v} = u\partial\eta/\partial x + v\partial\eta/\partial y \quad (24)$$

$$L(\xi) = \sqrt{\xi_x^2 + \xi_y^2} \quad (25)$$

$$L(\eta) = \sqrt{\eta_x^2 + \eta_y^2} \quad (26)$$

According to the vector splitting method, the eigenvalues of the flux vector can be expressed using positive and negative forms in order to split the flux vectors into two subvectors, that is,

$$E = (A^+ + A^-)Q = E^+ + E^- \quad (27)$$

$$F = (B^+ + B^-)Q = F^+ + F^- \quad (28)$$

The subvectors E^+ , E^- , F^+ , and F^- can be expressed according to the positive and negative eigenvalues as:

$$E^\pm = \frac{J\rho}{2\gamma} \begin{bmatrix} 2(\gamma-1)\lambda_{E1} + \lambda_{E3} + \lambda_{E4} \\ [2(\gamma-1)\lambda_{E1} + \lambda_{E3} + \lambda_{E4}]u + a(\lambda_{E3} - \lambda_{E4})\xi_x/L(\xi) \\ [2(\gamma-1)\lambda_{E1} + \lambda_{E3} + \lambda_{E4}]v + a(\lambda_{E3} - \lambda_{E4})\xi_y/L(\xi) \\ [2(\gamma-1)\lambda_{E1} + \lambda_{E3} + \lambda_{E4}](u^2 + v^2)/2 + a(\lambda_{E3} - \lambda_{E4})[u\xi_x/L(\xi) \\ + v\xi_y/L(\xi)] + (\lambda_{E3} + \lambda_{E4})a^2/(\gamma-1) \end{bmatrix} \quad (29)$$

$$F^\pm = \frac{J\rho}{2\gamma} \begin{bmatrix} 2(\gamma-1)\lambda_{F1} + \lambda_{F3} + \lambda_{F4} \\ [2(\gamma-1)\lambda_{F1} + \lambda_{F3} + \lambda_{F4}]u + a(\lambda_{F3} - \lambda_{F4})\eta_x/L(\eta) \\ [2(\gamma-1)\lambda_{F1} + \lambda_{F3} + \lambda_{F4}]v + a(\lambda_{F3} - \lambda_{F4})\eta_y/L(\eta) \\ [2(\gamma-1)\lambda_{F1} + \lambda_{F3} + \lambda_{F4}](u^2 + v^2)/2 + a(\lambda_{F3} - \lambda_{F4})[u\eta_x/L(\eta) \\ + v\eta_y/L(\eta)] + (\lambda_{F3} + \lambda_{F4})a^2/(\gamma-1) \end{bmatrix} \quad (30)$$

The FVS form of equation (7) can be written as:

$$Q_t + E_\xi^+ + E_\xi^- + F_\eta^+ + F_\eta^- = R_\xi + S_\eta \quad (31)$$

3.5.2 Special treatment of the artificial dissipation terms and numerical algorithm

In this section, the new way to treat the artificial dissipation terms is proposed. According to the idea proposed by Turkel and Vatsa [15], the dissipation terms can be written in the matrix-valued dissipation. However, in the present method, the dissipation term was not incorporated into the convection flux term as proposed by Turkel and Vatsa. The idea to overcome the effect of eigenvalue stiffness on the convergence of a compressible flow calculation was incorporated into the artificial dissipation terms. The dissipation terms were incorporated into the time-derivative terms. The dissipation terms can play a role in relatively enlarging the eigenvalue of the time-derivative term. The dissipation terms that required stabilizing the scheme is implemented in a convenient manner by modifying the convective fluxes as follows:

$$F_{\xi+1/2,\eta} = 0.5 * (F_{\xi\eta} + F_{\xi+1,\eta}) - D_{\xi+1/2,\eta} \quad (32)$$

$$E_{\xi\eta+1/2} = 0.5 * (E_{\xi\eta} + E_{\xi\eta+1}) - D_{\xi,\eta+1/2} \quad (33)$$

The terms $D_{\xi+1/2,\eta}$ and $D_{\xi,\eta+1/2}$ represent the dissipative terms in ξ and η directions, respectively. Two forms of artificial dissipation models can be used. One is the scalar dissipation model and another is the matrix-valued dissipation model.

The concept of the matrix-valued dissipation model was adapted and modified to form a new numerical scheme. The spectral radii for the ξ direction is:

$$\lambda_\xi = |\tilde{u}| + aL(\xi) \quad (34)$$

To form the present scheme and optimize the dissipation model in the sense that the same dissipation scaling is used for all the governing equations in a given coordinate direction, the matrix-valued dissipation term in ξ direction can be written as:

$$D_E = \Lambda_E \cdot \frac{\partial Q}{\partial t} \Delta t \quad (35)$$

Following the matrix-valued dissipation concept, matrix Λ in ξ direction can be written as:

$$\begin{aligned}\Lambda F = \lambda_3 I + & \left(\frac{\lambda_1 + \lambda_2}{2} - \lambda_3 \right) \times \left(\frac{\gamma - 1}{a^2} E_1 + \frac{E_2}{L(\xi)^2} \right) \\ & + \frac{\lambda_1 - \lambda_2}{2aL(\xi)} \times [E_3 + (\gamma - 1)E_4]\end{aligned}\quad (36)$$

where

$$\begin{aligned}E_1 &= [1, u, v, c_p T]^T [q^2/2, -u, -v, 1] \\ E_2 &= [0, \xi_x, \xi_y, \tilde{u}]^T [-\tilde{u}, \xi_x, \xi_y, 0] \\ E_3 &= [1, u, v, c_p T]^T [-\tilde{u}, \xi_x, \xi_y, 0] \\ E_4 &= [0, \xi_x, \xi_y, \tilde{u}]^T [q^2/2, -u, -v, 1]\end{aligned}\quad (37)$$

where q is the total velocity defined as:

$$q_2 = u_2 + v_2 \quad (38)$$

and

$$\begin{aligned}\lambda_1 &= \tilde{u} + aL(\xi) \\ \lambda_2 &= \tilde{u} - aL(\xi) \\ \lambda_3 &= \tilde{u}\end{aligned}\quad (39)$$

However, in the calculation, we cannot choose λ_1 , λ_2 , and λ_3 according to equation (39) because near the stagnation points, λ_3 approach zero, whereas near sonic lines λ_1 and λ_2 approach zero. For solving this problem, we limit these values in the following manner:

$$\begin{aligned}\lambda_1 &= \max(\tilde{u} + aL(\xi), \tilde{u}/M_r^2) \\ \lambda_2 &= \max(\tilde{u} - aL(\xi), \tilde{u}/M_r^2) \\ \lambda_3 &= \max(\tilde{u}, \tilde{u}/M_r^2)\end{aligned}\quad (40)$$

where

$$M_r = \begin{cases} 0.001 & M \leq 0.001 \\ M^2 & 0.001 < M \leq 1 \\ 1 & M > 1 \end{cases}\quad (41)$$

and where M is the calculated Mach number.

The matrix Λ_E can be represented in the form Λ_E^\pm according to the positive or negative value of λ_1 , λ_2 , and λ_3 , that is,

$$\Lambda_E = (\Lambda_E^+ - \Lambda_E^-)/2 \quad (42)$$

Similar expression can be derived for D_F by replacing the contravariant velocity \tilde{u} by \tilde{v} and ξ by η in equations (35)–(42). By adding the matrix form of the dissipative terms in the governing equation (31), we obtain:

$$\left[\Delta t \left(\frac{\partial \Lambda_A^+}{2\partial \xi} - \frac{\partial \Lambda_A^-}{2\partial \xi} + \frac{\partial \Lambda_B^+}{2\partial \eta} - \frac{\partial \Lambda_B^-}{2\partial \eta} \right) + I \right] Q_t + E_\xi^+ + E_\xi^- + F_\eta^+ + F_\eta^- = R_\xi + S_\eta \quad (43)$$

where I represents the unit matrix, and Λ_A^+ , Λ_A^- , Λ_B^+ , and Λ_B^- denote the matrices corresponding to the eigenvalues of the convection flux matrices.

By using the second-order time-difference algorithm, we have

$$\frac{Q^{n+1} - Q^n}{\Delta t} = \frac{1}{2} \left[\left(\frac{\partial Q}{\partial t} \right)^{n+1} + \left(\frac{\partial Q}{\partial t} \right)^n \right] \quad (44)$$

$$\begin{aligned} & \left[\Delta t \left(\frac{\partial \Lambda_A^+}{2\partial \xi} - \frac{\partial \Lambda_A^-}{2\partial \xi} + \frac{\partial \Lambda_B^+}{2\partial \eta} - \frac{\partial \Lambda_B^-}{2\partial \eta} \right) + I \right] \frac{Q^{n+1} - Q^n}{\Delta t} \\ &= - \left(\frac{\partial E^+}{\partial \xi} + \frac{\partial E^-}{\partial \xi} + \frac{\partial F^+}{\partial \eta} + \frac{\partial F^-}{\partial \eta} \right)^{n+1} + \left[\frac{\partial R}{\partial \xi} + \frac{\partial S}{\partial \eta} \right]^n \end{aligned} \quad (45)$$

Recalling that

$$\left(\frac{\partial E}{\partial \xi} \right)^{n+1} = \left(\frac{\partial E}{\partial \xi} \right)^n + \frac{\partial [A^n (Q^{n+1} - Q^n)]}{\partial \xi} \quad (46)$$

$$\left(\frac{\partial F}{\partial \eta} \right)^{n+1} = \left(\frac{\partial F}{\partial \eta} \right)^n + \frac{\partial [B^n (Q^{n+1} - Q^n)]}{\partial \eta} \quad (47)$$

Considering the stability, the forward and backward scheme were adopted in the difference of flux splitting. We can define that:

$$\Delta Q_{i,j}^{n+1} = -\Delta t \left[\frac{\nabla_\xi E^+}{\Delta \xi} + \frac{\Delta_\xi E^-}{\Delta \xi} + \frac{\nabla_\eta F^+}{\Delta \eta} + \frac{\Delta_\eta F^-}{\Delta \eta} \right]^n + \Delta t \left(\frac{\Delta_c R}{\Delta \xi} + \frac{\Delta_c S}{\Delta \eta} \right)^n \quad (48)$$

$$\delta Q_{i,j}^{n+1} = Q_{i,j}^{n+1} - Q_{i,j}^n \quad (49)$$

where the backward operator, for example in ξ for the E term, can be written as $\nabla_\xi E = 0.5 (3E_\xi - 4E_{\xi-1} + E_{\xi-2})$, whereas the forward operator $\Delta_\xi E = -0.5 (3E_\xi - 4E_{\xi+1} + E_{\xi+2})$ and $\Delta_\xi = 0.5 (R_{\xi+1} - R_{\xi-1})$.

By substituting equations (46) and (47) into equation (45) and considering the definition in equations (48) and (49), we can obtain the following difference scheme:

$$d_W \delta Q_{i-1,j}^{n+1} + d_S \delta Q_{i,j-1}^{n+1} + d_P \delta Q_{i,j}^{n+1} + d_N \delta Q_{i,j+1}^{n+1} + d_E \delta Q_{i+1,j}^{n+1} = \Delta Q_{i,j}^{n+1} \quad (50)$$

where

$$\begin{aligned} d_W &= -\frac{\Delta t}{2\Delta\xi} (\Lambda_A^+)_i^n - \frac{\Delta t}{2\Delta\xi} (A_{i-1,j}^+)^n \\ d_S &= -\frac{\Delta t}{2\Delta\eta} (\Lambda_B^+)_j^n - \frac{\Delta t}{2\Delta\eta} (B_{i,j-1}^+)^n \\ d_P &= 1 + \frac{\Delta t}{2\Delta\xi} (\Lambda_A^+)_i^n - \frac{\Delta t}{2\Delta\xi} (\Lambda_A^-)_i^n + \frac{\Delta t}{2\Delta\eta} (\Lambda_B^+)_j^n - \frac{\Delta t}{2\Delta\eta} (\Lambda_B^-)_j^n \\ &\quad + \frac{\Delta t}{2\Delta\xi} (A_{i,j}^+)^n - \frac{\Delta t}{2\Delta\xi} (A_{i,j}^-)^n + \frac{\Delta t}{2\Delta\eta} (B_{i,j}^+)^n - \frac{\Delta t}{2\Delta\eta} (B_{i,j}^-)^n \\ d_N &= \frac{\Delta t}{2\Delta\eta} (\Lambda_B^-)_j^n + \frac{\Delta t}{2\Delta\eta} (B_{i,j+1}^-)^n \\ d_E &= \frac{\Delta t}{2\Delta\xi} (\Lambda_B^-)_i^n + \frac{\Delta t}{4\Delta\xi} (A_{i+1,j}^-)^n \end{aligned}$$

In the above expressions, d_W , d_S , d_P , and d_E are scalars, which can be calculated directly from flux vectors. It is shown that the present scheme mainly includes both implicit and explicit parts. Equation (50) is the implicit part of the present scheme, which can improve the numerical stability when a large time step is used in the calculation. Because of the application of flux vectors, the computational efforts can be greatly reduced as compared to other coefficient matrix implicit schemes for any time step [2]. Equation (48) is the explicit part of the scheme. We found that the implicit part, equation (50), can be solved by using a two-sweep method, which is similar to the hopscotch-sweep method but along i (i.e., ξ) and j (i.e., η) directions. Equation (50) can then be written in the form of the two-sweep form as follows:

$$d_S \delta Q_{i,j-1}^{n+1} + d_P \delta Q_{i,j}^{n+1} + d_N \delta Q_{i,j+1}^{n+1} = \Delta Q_{i,j}^{n+1} - d_W \delta Q_{i-1,j}^{n+1} - d_E \delta Q_{i+1,j}^{n+1} \quad (51)$$

Calculation starts from $i = 1$ to i_{max} along the i -direction. At each station i of the time step $n + 1$, the variables $\delta Q_{i,j-1}$, $\delta Q_{i,j}$, and $\delta Q_{i,j+1}$ can be obtained along the j -direction. It is clearly shown that the present scheme consists of an explicit part of $\Delta Q_{i,j}$ at any time step “ n ” and an implicit $\delta Q_{i,j}$ at time step “ $n + 1$.”

3.6 Stability Analysis

Stability is one of the important aspects in developing a numerical scheme, and only a stable scheme can be used to simulate the differential equations. In this

section, the stability analysis of the present numerical scheme is addressed. One of the most common methods of stability analysis is the von Neumann stability analysis. With this method, a finite Fourier series expansion of the solution to a model equation is made, and the decay or amplification of each mode is separately considered to determine whether the method is stable or not. Analysis showed that the typical stability limitation for the explicit scheme equation (49) is the CFL number restriction. Following the von Neumann analysis, it was shown that the present implicit scheme is unconditionally stable.

Nonetheless, the determination of stability condition for the present hybrid scheme is quite difficult due to the complication of the equation systems employed for the computation. To simplify the analysis, let all η derivatives in equation (47) equal to zero and all the eigenvalues be greater than zero so that in ξ -direction we have $|A^-| = 0$ and $|A^-| = |P_E^+ P^{-1}|$. In the stability analysis, only one of the two velocity components was performed. The stability analysis of the other variables is the same. The experience shows that, if each variable in the equation system is stable, the entire equation system in general is stable. Suppose that ρ_A and ρ_M are, respectively, the spectral radii of the matrices A and M , where

$$M = \frac{\partial R}{\partial \left(\frac{\partial Q}{\partial \xi} \right)} \quad (52)$$

The numerical scheme of equations (48) and (49) can be written in the following scalar forms:

$$\Delta u_i^{n+1} = -\Delta t_\xi \left[\frac{\nabla_\xi E^+}{\Delta \xi} \right]^n + \Delta t_\xi \left(\frac{\Delta_c R}{\Delta \xi} \right)^n \quad (53)$$

$$\left(-\frac{\Delta t_\xi \rho_A}{\Delta \xi} \right) (u_{i-1}^{n+1} - u_{i-1}^n) + \left(1 + \frac{\Delta t_\xi \rho_A}{\Delta \xi} \right) (u_i^{n+1} - u_i^n) = \Delta u_i^{n+1} \quad (54)$$

Upon substitution of the Fourier component of the solution into equations (53) and (54), we have:

$$u_{i+1}^{n+1} = V^{n+1} e^{(i+1)\theta \sqrt{-1}} \quad (55)$$

The following amplification factor for equations (53) and (54) can be obtained:

$$|G|^2 = \frac{\left\{ \Delta \xi + \frac{\Delta t_\xi}{\Delta \xi} \rho_M [-2 + 2 \cos(m \Delta \xi)] \right\}^2 + [\Delta t_\xi \rho_A \sin(m \Delta \xi)]^2}{[\Delta \xi + \Delta t_\xi \rho_A - \Delta t_\xi \rho_A \cos(m \Delta \xi)]^2 + [\Delta t_\xi \rho_A \sin(m \Delta \xi)]^2} \quad (56)$$

The stability condition of the scheme is that the absolute value of G is less than or equal to unity for all wave numbers m . From equation (56), the following stability condition can be obtained:

$$\Delta t_\xi \leq \frac{\Delta \xi}{\rho_A + \frac{2\rho_M}{\Delta \xi}} \quad (57)$$

From a similar analysis, letting all ξ derivatives in equation (50) equal to zero and letting all the eigenvalues be greater than zero, we can obtain:

$$\Delta t_\eta \leq \frac{\Delta\eta}{\rho_B + \frac{2\rho_N}{\Delta\eta}} \quad (58)$$

where ρ_B and ρ_N represent the spectral radii of the matrices B and N , respectively, and N is defined as:

$$N = \frac{\partial S}{\partial \left(\frac{\partial Q}{\partial \eta} \right)} \quad (59)$$

It is clearly shown that the stability constraint of equations (57) and (58) is due to the explicit part. Therefore, we can conclude that the present implicit scheme is unconditionally stable.

By solving the spectral radii of the flux Jacobian matrices for the convective terms and diffusion terms, we can obtain the following stability conditions:

$$\Delta t_\xi \leq \frac{\Delta\xi}{|\tilde{u}| + a\sqrt{\xi_x^2 + \eta_y^2} + \frac{2\omega}{\Delta\xi\rho}(\xi_x^2 + \eta_y^2)} \quad (60)$$

$$\Delta t_\eta \leq \frac{\Delta\eta}{|\tilde{v}| + a\sqrt{\xi_x^2 + \eta_y^2} + \frac{2\omega}{\Delta\eta\rho}(\xi_x^2 + \eta_y^2)} \quad (61)$$

where

$$\omega = \max(\mu, \lambda + 2\mu, \mu\gamma/\text{Pr})$$

The stability analysis shows that the stability restraint arises from the explicit part for the present scheme. The implicit part of the scheme is unconditionally stable. However, from the calculation point of view, the explicit part must be treated within a certain time step. When the time step increases, the implicit scheme is activated. From this point of view, the present overall scheme can be judged to be unconditionally stable. From the practical point of view, however, many factors such as smoothing of the implicit calculation, local linearization of the nonlinear equations, and complex geometry, all will constrict the time step. Therefore, the time step must be selected according to the CFL number as:

$$\Delta t = \min(\Delta t_\xi, \Delta t_\eta) \quad (62)$$

where

$$\Delta t_\xi \leq \frac{\text{CFL} \cdot \Delta\xi}{|\tilde{u}| + a\sqrt{\xi_x^2 + \eta_y^2} + \frac{2\omega}{\Delta\xi\rho}(\xi_x^2 + \eta_y^2)} \quad (63)$$

$$\Delta t_\eta \leq \frac{\text{CFL} \cdot \Delta\eta}{|\tilde{v}| + a\sqrt{\xi_x^2 + \eta_y^2} + \frac{2\omega}{\Delta\eta\rho}(\xi_x^2 + \eta_y^2)} \quad (64)$$

From equations (63) and (64), it is easily found that when $\text{CFL} \leq 1$, the time step constriction is the restriction of the typical explicit scheme. For the present scheme the CFL number can be chosen to be around 15. Anyhow it is clear that the present scheme is more economical than a pure explicit scheme.

3.7 Applications in Turbine Cascade

In a practical use of the code, the convergence criterion is one of the important factors of the CFD program. There are several different kinds of convergence criteria to check the convergent state of the numerical solutions. In this work, two convergent criteria are used to check the convergence of the calculation. One is the RMS residue of the four independent variables and the other is the mass flux error. They are defined as follows:

$$\text{Error(RMS)} = \frac{100 \times \sqrt{(u)^2 + (v)^2 + (\rho)^2 + (e)^2}}{\Delta t} \quad (65)$$

$$\text{Error}(m') = 100 \times |m'_0 - m'_{\text{EXIT}}| / m'_{\text{EXIT}} \quad (66)$$

The actual computational case studies are given in the following subsections.

3.7.1 C3X turbine cascade

The present numerical scheme was tested for a two-dimensional compressible flow over a C3X turbine cascade [2,34] by using the algebraic H-mesh as shown in Figure 3.2. The flow considered corresponds to the case 144 of the experiment conducted by Hylton et al. [34]. The geometric inlet and exit angles of the blade are 0 degrees and 72 degrees, respectively. The flow conditions at the inlet are total pressure $P_{T0} = 7.899 \text{ kPa}$, total temperature $T_{T0} = 815 \text{ K}$, and the inlet Reynolds number based on the true chord $\text{Re} = 0.63 \times 10^6$. The ratio of exit static to inlet total pressure is 0.59. The flow was assumed to be fully turbulent in the entire computation domain.

The effects of the mesh refinement study for case 144 were conducted prior to the computations. The different levels of the meshes based on the basic size 43×12 are shown in Figure 3.3. The study showed that the mass flux residual tends to approach a certain level with the increase in the mesh size. It was suggested that the mesh size of 86×24 is sufficient for a grid-independent condition. In order to assess the impact of the present scheme, the study of the influence of the CFL number was investigated. Figure 3.4 shows the typical convergent histories with different CFL numbers (1, 5, 7, 9, and 11) by using a mesh distribution of 86×24 . The research shows that when the CFL number increases, the convergent rate is improved. The results also show that when the CFL number is greater than 7, the improvement in the convergent rate is small. If the CFL = 1, the computational performance is

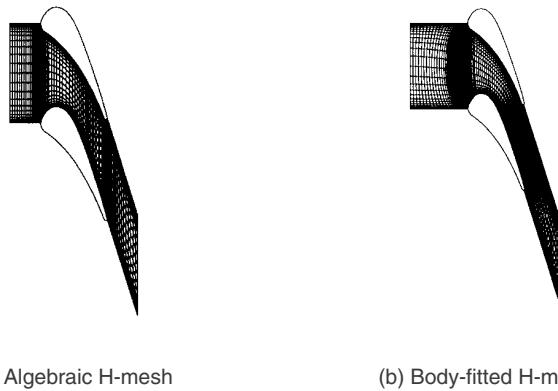


Figure 3.2. Computational meshes.

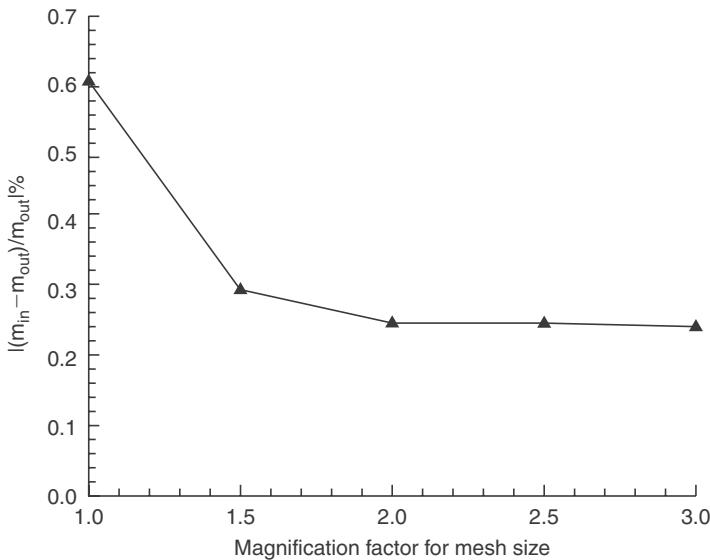


Figure 3.3. Mesh refine study.

similar to the case of explicit scheme. In the present computation, the CFL number is selected in the range between 5 to 11. It is clear that the present scheme is more effective than the explicit scheme. It is well known that refinement of the high-grid aspect ratio influences the convergence of the calculation.

Figure 3.5 shows good agreement of the present predictions for surface static pressure distribution with the experimental data measured by Hylton et al. [34] and N-S computations by Kwon [35]. Figure 3.6 presents the computed Mach number

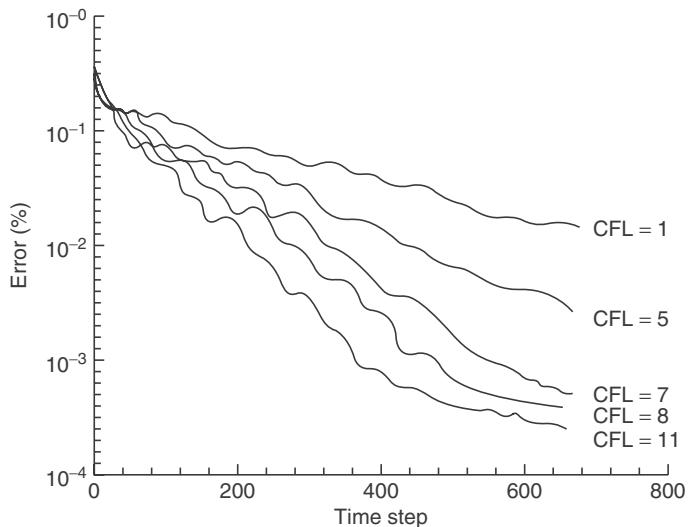


Figure 3.4. Computational convergent histories.

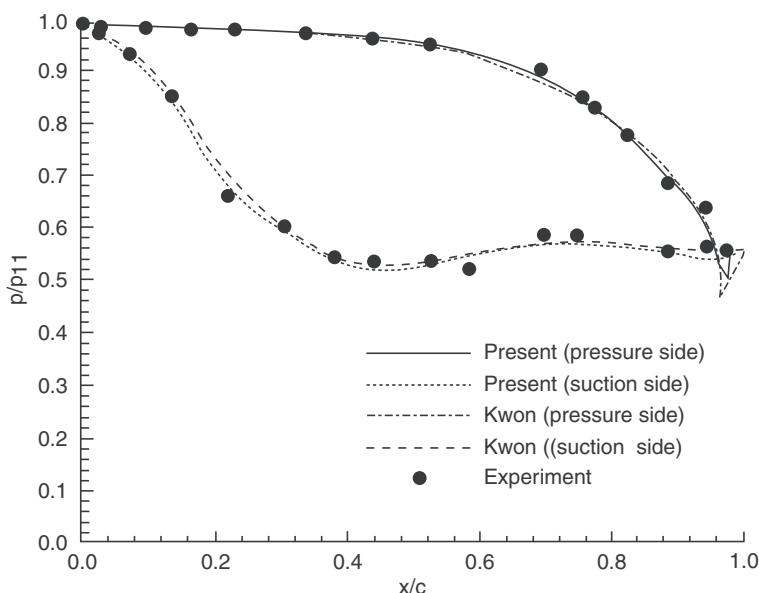


Figure 3.5. Blade surface static pressure distributions.

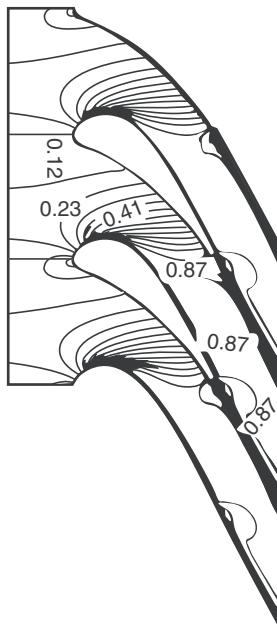


Figure 3.6. Contour of computed Mach.

contours. The results show that the flow is entirely subsonic and the computed maximum Mach number is about 0.87. The wakes associated with the airfoils are clearly seen in the figure. It is also shown that the stagnation point occurs near the nose of the blade surface. These features are consistent with those expected in a cascade flow.

Next, the heat transfer coefficient was computed and compared with the experimental data. It can be seen in Figure 3.7 that the computed heat transfer coefficient is in good agreement with the experimental data on the pressure surface except near the leading edge. However, the computed heat transfer coefficient was overpredicted in the downstream region immediate vicinity of the leading edge along the suction surface. This discrepancy may be caused by the heat transfer evaluation through the energy equation that is coupled with the momentum equations. In the experiments, the flow field starts from laminar at the upper stream and then undergoes transition to become fully turbulent. However, in the calculation, the flow was assumed to be fully turbulent resulting in the overprediction of the heat transfer coefficient near the leading edge of the suction surface.

Figure 3.8 shows the convergent history when using these two types of H-meshes. There are slight differences appearing between the two mesh models. It seems that the algebraic H-mesh has a better convergent property than the body-fitted H-mesh.

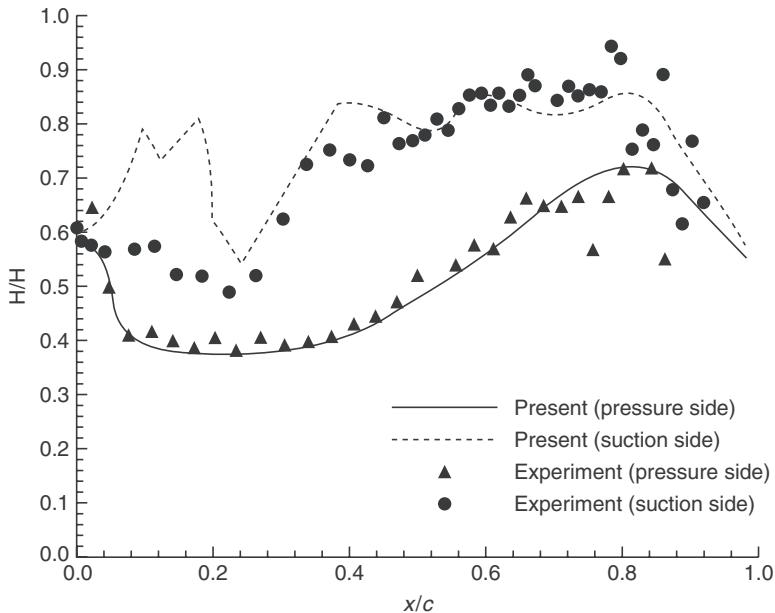


Figure 3.7. Heat transfer coefficient distribution on blade surface.

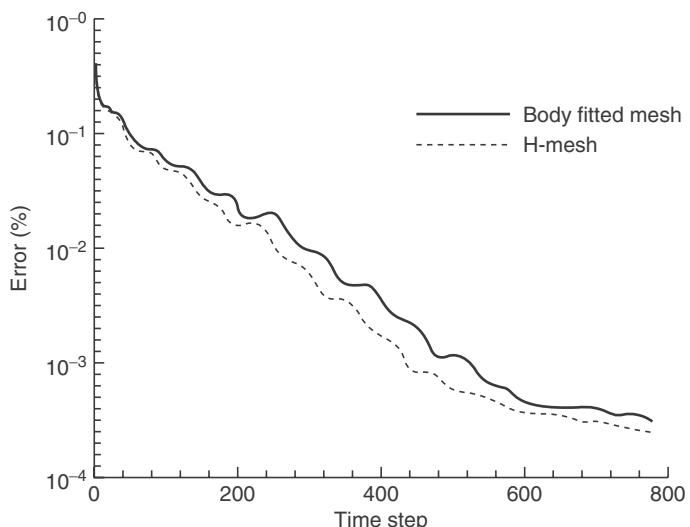


Figure 3.8. Mesh influence of the convergent histories.

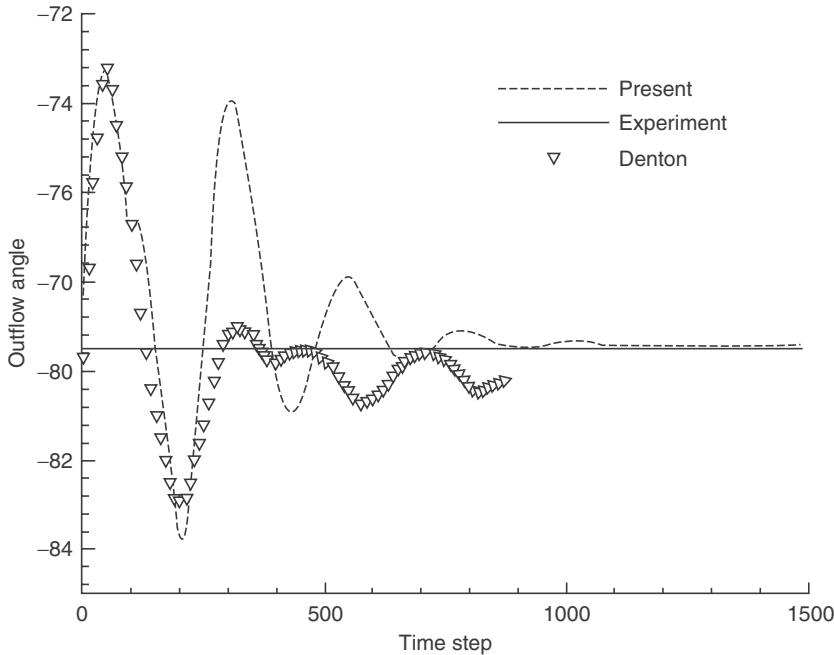


Figure 3.9. Convergent histories of the outflow angle.

3.7.2 VKI turbine cascade

Another test case of the present algorithm is the flow around the VKI turbine cascade. In the calculation a 77×21 H-type mesh, which is similar to that shown in Figure 3.2a, was used with the same surface definition as suggested by Xu and Amano as well as Kwon [2,35]. For comparison with other computational results, the mesh points were also selected to be the same as those suggested by Sieverding [38]. The flow conditions were the same as in his experiments. Figure 3.9 shows the convergence history for the average outflow angle. The present computations show the angle effectively reaching a steady value, which is in good agreement with the experiment [37]. Further noted in the figure is the convergent rate of the present scheme being slightly better than that in the method proposed by Denton [37]. The computed outflow angle with different isentropic exit Mach numbers is shown in Figure 3.10. The isentropic Mach numbers were calculated using the Bernoulli's equation with total pressure at the entrance and static pressure on the wall. The results show that the present calculation agrees well with the experiment. The calculated exit angle when the exit isentropic Mach number is 1.2 is almost the same as that of Denton. Figure 3.11 shows the computed blade surface isentropic Mach number distributions compared with both the experiment and computed results by Denton [37]. It shows that the present computation agrees with the experimental

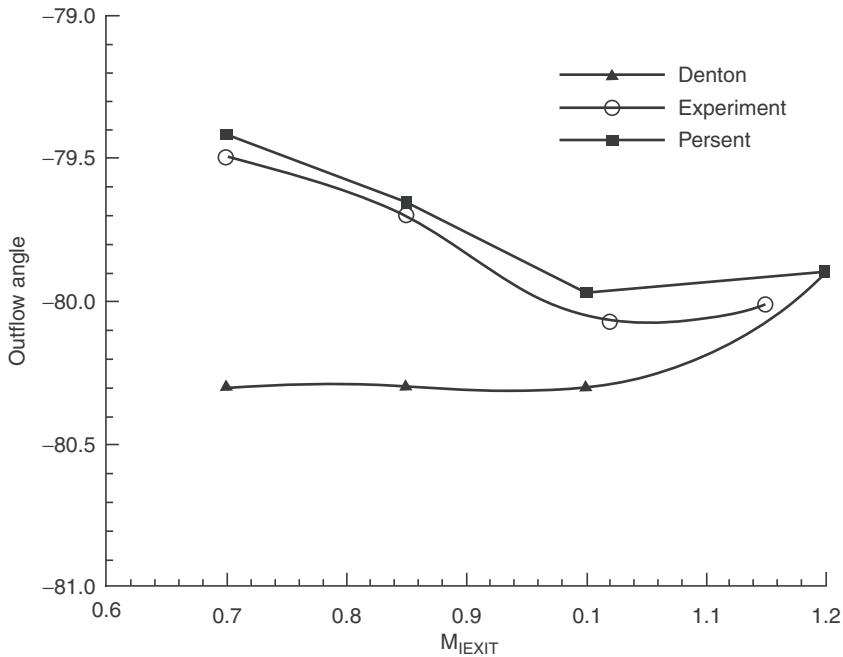


Figure 3.10. Outflow angle with Mach number.

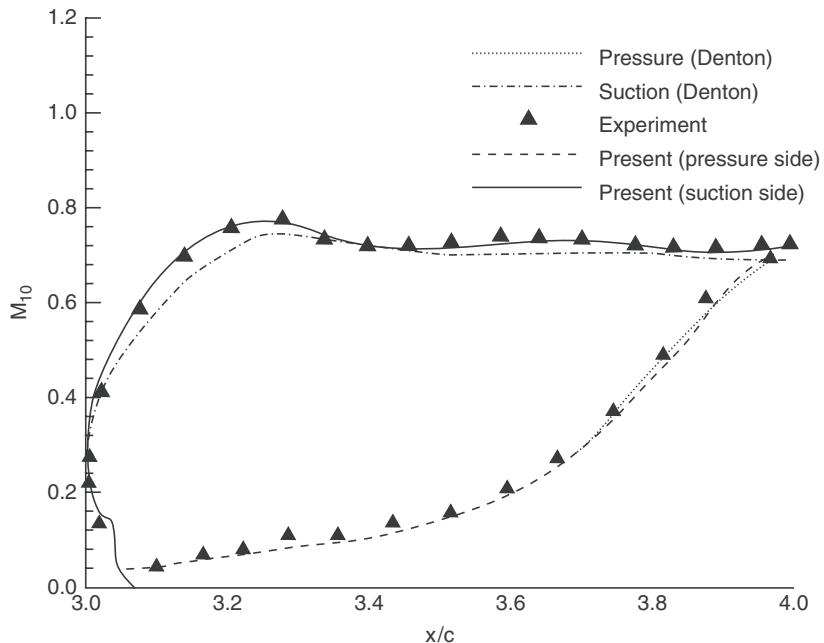


Figure 3.11. Isentropic Mach number distributions on the blade surface.

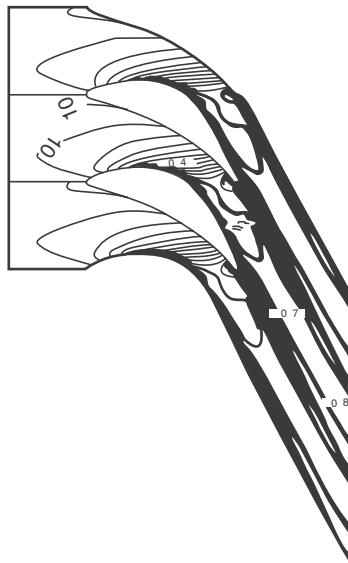


Figure 3.12. Contour of computed Mach number.

ones when compared with those by Denton. Figure 3.12 is Mach number contour between the cascade blades.

An efficient numerical scheme for computations of the flow and heat transfer in turbine cascade flow was proposed and tested. It was demonstrated that flux-vector scheme has the following advantages:

- The flux-vector scheme avoids the coefficient matrix operation and omits any iteration within the same time-step calculation. Therefore, less computational effort is needed in the calculation.
- The present algorithm possesses the positive features of the explicit and implicit algorithms. Because the present algorithm did not use approximation factor (AF) in the time-marching process, the CFL number can be chosen to be large enough to save computational time. Furthermore, since the present scheme includes an implicit part, the method improves the convergence ability especially for the large time-step computations.
- The computation shows that the present scheme has a similar convergent rate compared with the multigrid method developed by Denton [37]. The upwind scheme is used in the FVS scheme in order to improve the stability of the overall scheme.
- The flux-vector method has several merits such as good convergence and overall accuracy in predicting the flow field and heat transfer around the turbine cascades. This fact was supported by the validation test shown in good agreement with experiments.

In the present method, the dissipation terms were incorporated into the time-derivative terms. This character renders the scheme more accurate as well as avoids the eigenvalue stiffness when the calculation is performed in a low Mach number range and accelerates the convergence of the simulation. Finally, it is shown that the present scheme is more advantageous than common implicit and explicit schemes. The present scheme can be easily extended to full three-dimensional turbine flow calculations.

3.8 Numerical Method for Three-Dimensional Flows

The Navier–Stokes equations can be written in a body-fitted curvilinear coordinate system (ξ, η, ζ) as follows:

$$\frac{\partial(J^{-1}U)}{\partial t} + \frac{\partial F}{\partial \xi} + \frac{\partial G}{\partial \eta} + \frac{\partial H}{\partial \zeta} = \frac{\partial F_V}{\partial \xi} + \frac{\partial G_V}{\partial \eta} + \frac{\partial H_V}{\partial \zeta} \quad (67)$$

where the convective terms can be written as:

$$\frac{\partial F}{\partial \xi} = A \frac{\partial U}{\partial \xi} \quad (68)$$

$$\frac{\partial G}{\partial \eta} = B \frac{\partial U}{\partial \eta} \quad (69)$$

$$\frac{\partial H}{\partial \zeta} = C \frac{\partial U}{\partial \zeta} \quad (70)$$

The detailed expressions in the above equations can be found in Turkel and Vatsa [15]. Following the previous section, a dissipation model employed in this study is a nonisotropic model, in which the dissipative terms are given as a function of the spectral radii of the Jacobian matrices associated with the appropriate coordinate directions. The dissipative terms in the three directions of the coordinates have forms similar to that discussed in previous section and in Turkel and Vatsa [15]. This method can overcome the effect of eigenvalue stiffness on the convergence of an incompressible flow calculation by using the time-matching scheme. The dissipation terms were incorporated into the time-derivative terms. In this way, the dissipation terms can enlarge the eigenvalue of the time-derivative terms, which will benefit the flow solver convergence, in particular when the code is used to solve the low-speed flow problems. The dissipation term that is required to stabilize the scheme is implemented in a convenient manner by modifying the convective fluxes as follows:

$$F_{\xi+1/2,\eta,\zeta} = 0.5 * (F_{\xi,\eta,\zeta} + F_{\xi+1,\eta,\zeta}) - d_{\xi+1/2,\eta,\zeta} \quad (71)$$

$$G_{\xi,\eta+1/2,\zeta} = 0.5 * (G_{\xi,\eta,\zeta} + G_{\xi,\eta+1,\zeta}) - d_{\xi,\eta+1/2,\zeta} \quad (72)$$

$$H_{\xi,\eta,\zeta+1/2} = 0.5 * (H_{\xi,\eta,\zeta} + H_{\xi,\eta,\zeta+1}) - d_{\xi,\eta,\zeta+1/2} \quad (73)$$

The terms $d_{\xi+1/2,\eta,\zeta}$, $d_{\xi,\eta+1/2,\zeta}$, and $d_{\xi,\eta,\zeta+1/2}$ represent the dissipative terms in ξ —, η —, and ζ — directions, respectively, which can be found in Xu and Amano [1,3]. Using equations (2)–(4), the time derivative of dissipative terms can be written into a matrix form as:

$$D = [1, d_{\xi+1/2,\eta,\zeta}, /u, d_{\xi,\eta+1/2,\zeta}/v, d_{\xi,\eta,\zeta+1/2}/w, 1]^T \quad (74)$$

By substituting the dissipative terms of equation (74) into equations (71) through (73) and incorporating into equation (67), we obtain:

$$\left(I + \frac{D}{\Delta U} \right) \frac{\partial(J^{-1}U)}{\partial t} + \frac{\partial F}{\partial \xi} + \frac{\partial G}{\partial \eta} + \frac{\partial H}{\partial \zeta} = \frac{\partial F_V}{\partial \xi} + \frac{\partial G_V}{\partial \eta} + \frac{\partial H_V}{\partial \zeta} \quad (75)$$

The second-order numerical dissipation terms were set so that the variables decrease exponentially as a function of time. This technique allows a computation to damp out high-frequency errors at the initial stage of iterations and, therefore, avoids the second-order numerical dissipation terms to dissipate when the computation approaches a converged state. A finite-volume algorithm based on a Runge–Kutta time-stepping scheme [15] is used to obtain the steady-state solutions by solving the Navier–Stokes equations. The turbulence model used in this study is a Baldwin–Lomax turbulence model [32]. The meshes generated in this study are hexagonal meshes; the grid lines near the solid wall are almost normal or parallel to the solid walls. In order to simplify the programming effort, the turbulence equations were applied along grid lines rather than in the direction normal to the solid boundaries. This avoids flow calculations in all the normal directions to the solid wall and the necessary interpolations of the flow variables. During the calculations, it is difficult to keep all the y^+ values within a small level, especially close to the corner region. Therefore, the wall function was used in the calculation. The static pressure coefficient is defined as:

$$C_{ps} = \frac{P - P_{01}}{p_{01} - \bar{p}_{s2}} \quad (76)$$

The total pressure loss is defined as:

$$C_{p0} = \frac{P_{01} - P_0}{p_{01} - \bar{p}_{s2}} \quad (77)$$

where \bar{p}_{s2} represents the mass average static pressure of the blade exit plane.

3.9 Applications of Three-Dimensional Method

3.9.1 Analysis of pitch-width effects on the secondary flows of turbine blades

The inlet and outlet boundary conditions used in the computations were taken from measurement. The total pressure at inlet was equal to the test [38] for all the

calculations. At the exit plane, one of the most critical conditions was to specify the static pressure. The average exit pressure in the experiment [38] was used in the calculations. The asymptotic boundary conditions with second-order streamwise derivatives of variables were set to zero. The overall mass conservation through the blade flow passage is imposed by correcting the outlet velocity components with the calculated velocity profiles on the plane next to the outlet and inlet flow mass during iterations. The no-slip wall boundary conditions are used for all the solid walls.

Grid-converged solutions for the high-speed viscous flows over a turbine blade should be obtained with sufficiently fine meshes. However, a computation with fine meshes is very time consuming and difficult to assess the numerical accuracy of the solution. In fact Inoue and Furukawa [23] reported that it was difficult to optimize the artificial dissipation coefficients by using a common method that impaired accuracy and reliability of the schemes to predict a cascade performance. To predict a high Reynolds number turbulent flow, highly clustered grids are required toward the wall. The grid refinement study was conducted through the computational uncertainty study. The relationship between the calculated mass residual, $100 \times |m_{in} - m_{out}|/m_{out} \%$, and various grid sizes was tested to identify the point where the mass residual reaches its asymptotic value. In this study, the systematic grid refinement studies are performed. The grid refinement study was conducted by using Richardson extrapolation method [39]. The mass factor, $f = |1 - m_{in}/m_{out}|$, is used to evaluate the grid refine study. Using the fine grid calculation with a magnification factor of 3, we obtain $f_1 = 0.18 \times 10^{-4}$. Then we change the grid with the magnification factor to 2, and obtain $f_2 = 0.85 \times 10^{-4}$. In this case, the error is $\varepsilon = 100 \times (f_2 - f_1)/f_1 = 0.67\%$. For a grid increase rate, $IN = 1.5$, the fine grid value of the grid convergence index (GCI) for the present second order is $GCI = 3\varepsilon/(IN^2 - 1) = 1.61\%$. Although the confidence in the GCI as error band is not justifiable, it shows the current grid structure to be conservative. It also shows that the uncertainty in the current calculation for the mass conservation is within 1.61%. The distance of the first mesh from the solid surface was chosen so that a maximum y^+ value becomes less than 15, which is considered to be fine enough for the present calculations. The mesh has $45 \times 41 \times 115$ node points in the pitchwise, spanwise, and streamwise directions, respectively, which was identified to an optimal situation as a grid-independent state. The blade-to-blade computational mesh is shown in Figure 3.13. The mesh is formed consisting of three zones: upstream of the blade, inside the blade, and downstream of the blade sections. For comparison with the test results [38], the outlet of the calculation plane was selected to be perpendicular to the x -direction (axial direction), which corresponds to the tested exit plane.

Most CFD studies [40–42] on turbomachines were mainly based on a linear cascade of high-turning turbine blades where the secondary flow was very strong. In this study, an annular turbine blade was focused, which is similar to actual turbine stator vanes. The airfoil model investigated by Sieverding [38], where the blade tests were cited from NACA TN D-3751, is used for validation of the code. The tip and root radii are 0.355 and 0.283 m, respectively. The stagger angle is 42.5 degrees with respect to axial direction and the blade axial width length, $C_a = 0.087$ m. The blade profile is constant along the blade height and is untwisted. Three root pitch-to-width

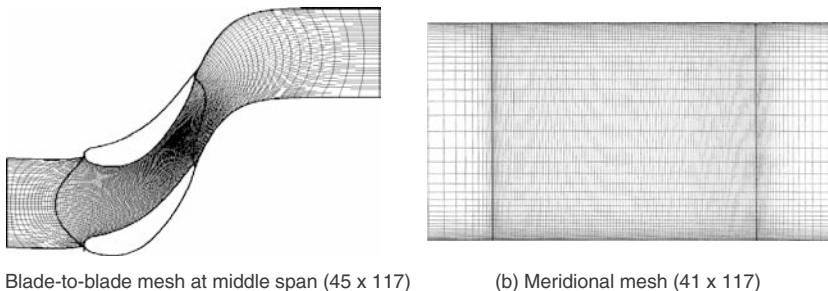


Figure 3.13. Calculation meshes.

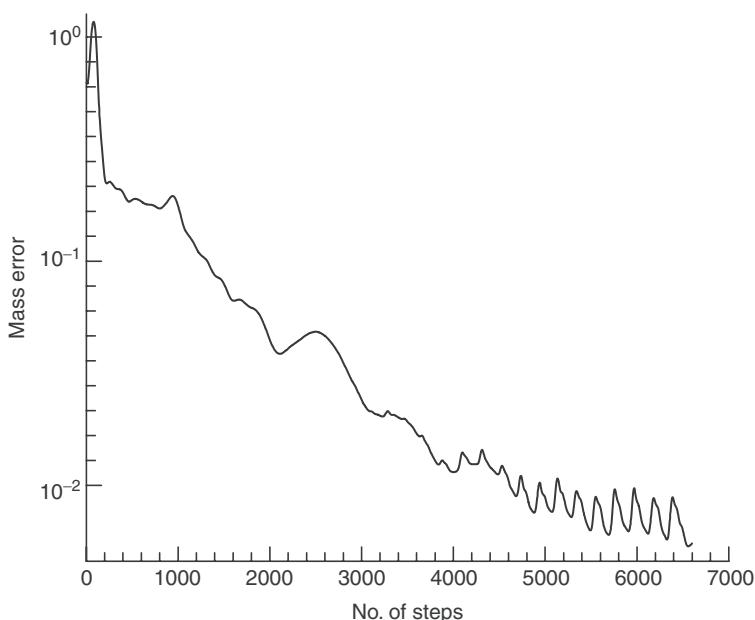


Figure 3.14. Convergence history.

ratios, $b/C_a = 0.639$, 0.973 , and 1.460 , are studied here. The root pitch-to-width ratio, b/C_a , was varied by altering the number of blades. Three blade numbers were used in the calculation: 26 , 21 , and 14 . These calculations were compared with the experimental data of the annular turbine blades [38].

The computations were assumed to be converged when the mass flow error became less than 0.1% of the inlet mass flow rate. The mesh used in this study and typical convergence history are shown in Figure 3.14. The convergence speed became slow after the mass flow error became close to 0.1% . In all the plots shown in this section, the viewpoint is set to look in the downstream direction from the

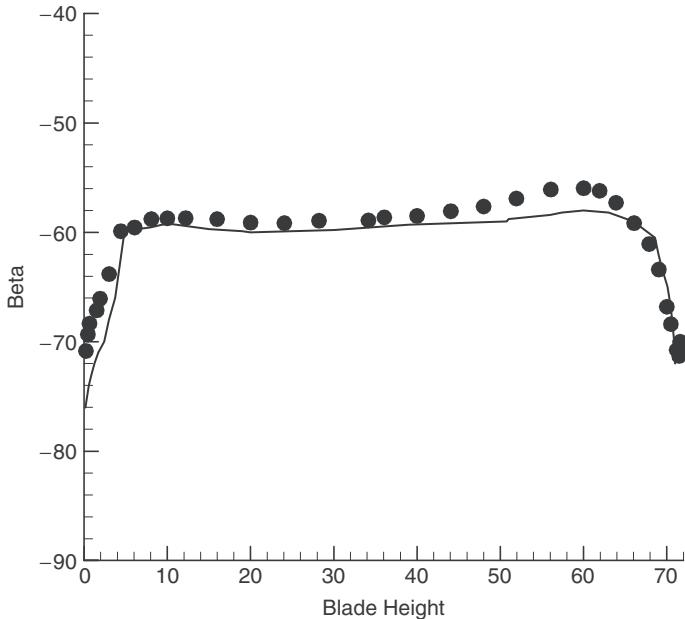


Figure 3.15. Average spanwise flow angle distribution at $x/C_a = 0.9$ (dot is experimental results).

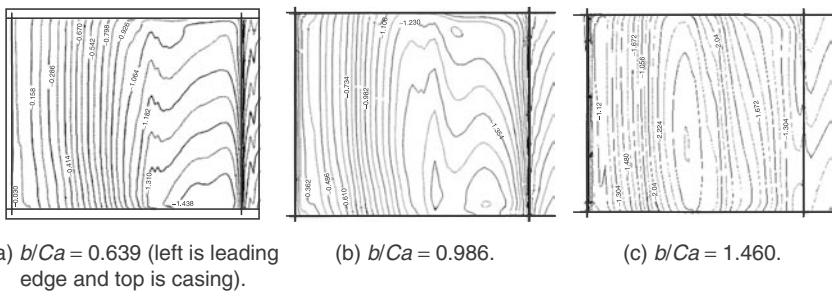


Figure 3.16. Static pressure coefficient distributions on the suction surface of the blade.

upstream location, with the pressure surface on the left-hand side and the suction surface on the right-hand side.

Figure 3.15 shows the comparison of the experimental results [38] with the present computations for the average spanwise flow angle distribution located at $x/C_a = 0.9$. The computations are in agreement with the experiment. Figure 3.16 shows the static pressure coefficient distributions near the suction surface of blades for different root pitch-to-width ratios, b/C_a . The inception and transport of the passage vortex and its impact on the surface boundary layers present different

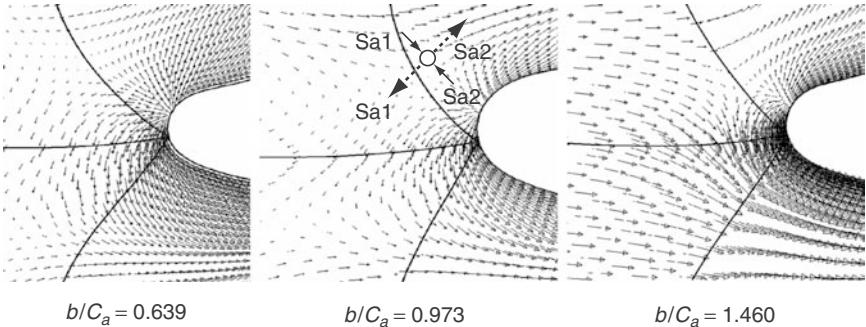


Figure 3.17. Velocity vectors at the leading edge of the blade at 0.28% height from root.

characteristics for different b/C_a . There is a region near the wall corner where the chordwise and spanwise pressure gradients are small. It is also clear from Figure 3.16 that the possible boundary layer separation lines occur for cases $b/C_a = 0.639$ and 0.986 , and they are located at the corners of the hub and trailing edge. The separation region for the case $b/C_a = 1.460$ is located at 50% of C_a position. The separation location seems to move forward as b/C_a increases. This feature confirms that the inception of the passage vortex developed near the airfoil leading edge and grew fast when the pitch b/C_a was increased. Figure 3.16 also shows that the pressure coefficient on the suction surface is lower for larger b/C_a . This is because, as b/C_a increases, the flow constraint from the blade becomes small, and therefore, the blade's local curvature plays an important role. For the case of $b/C_a = 0.639$ and 0.986 , a vortex develops faster near the blade tip and moves toward the casing because the local pitch size at the tip is much larger than that at the hub. In the case of $b/C_a = 1.460$, the casing and the hub vortex have similar characteristics because b/C_a is large enough, thus causing the blockage to be small enough so that it does not restrict the flow. The flow characteristics are dominated by local curvature for b/C_a larger than 1.46 .

The experimental data [43,44] indicated that one of the most important features in a turbine blade boundary layer is the occurrence of the separation and saddle point on the end wall near the leading edge. The predicted boundary layer flow patterns for different b/C_a near the hub leading edge at 0.28% height are shown in Figure 3.17. The calculations show that there is a saddle point for all b/C_a . Two attachment lines, Sa1 and Sa2, and two separation lines, Ss1 and Ss2, can be identified from the velocity vectors. The separation lines correspond to the two legs of the horseshoe vortices formed near the leading edge. The separation line Ss2 is pushed away from the leading edge. The first separation line Ss1 moves around the leading edge and moves against the freestream flow direction and it merges with the flow coming from the upstream direction to form a suction-side horseshoe vortex. The separation line Ss1 is also developed along the suction side of the blade, which can be seen from the static pressure coefficient distribution in Figure 3.17.

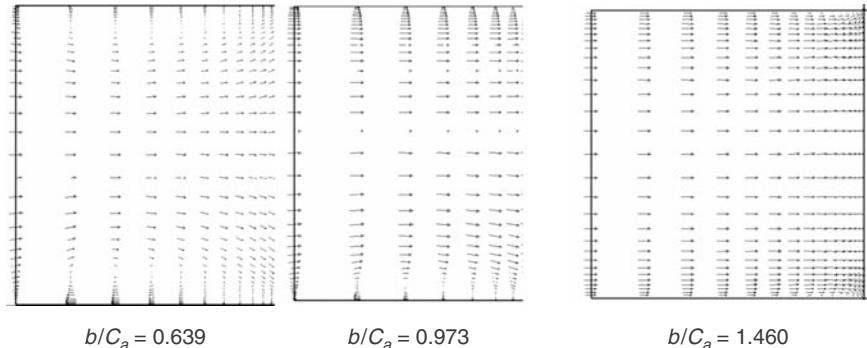


Figure 3.18. Velocity vectors in the meridional surface before the leading edge.

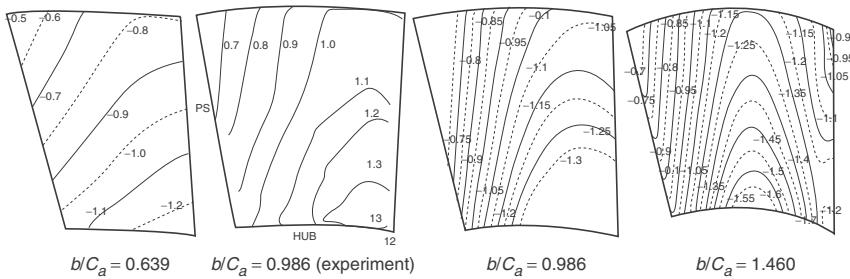
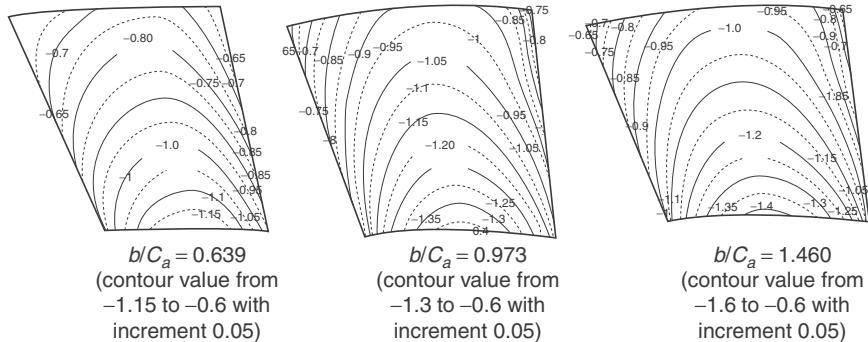
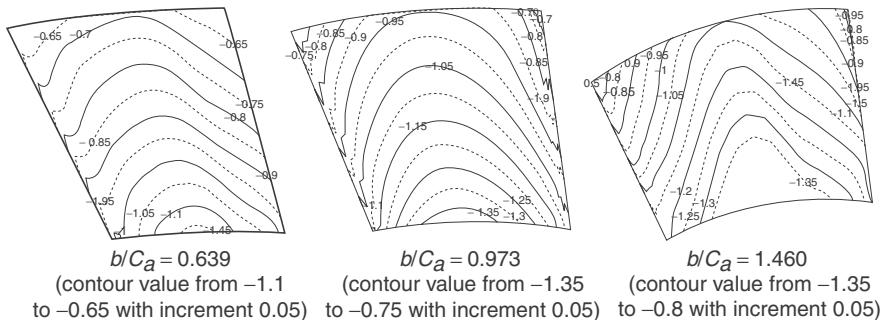


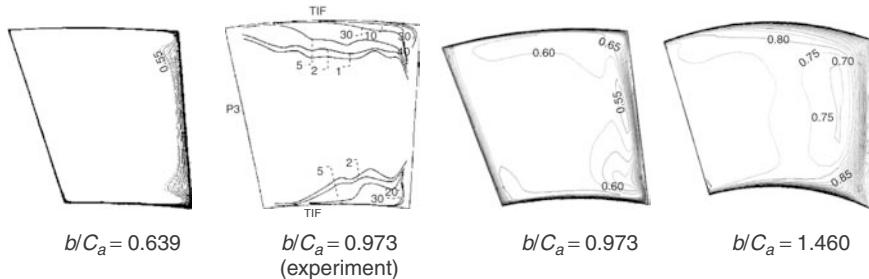
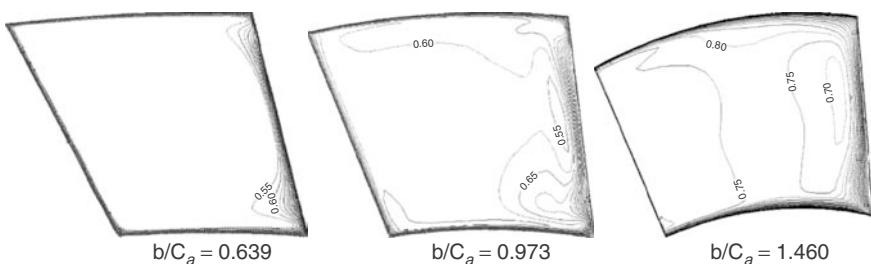
Figure 3.19. Static pressure coefficient contour at $x/C_a = 0.9$.

The locations of the saddle points are closer to the leading edge for larger pitch sizes. This is why some of the flow visualization [42] did not find a saddle point if the experiment did not extend sufficiently far upstream. The velocity vectors in the meridional plane upstream of the leading edge are shown in Figure 3.18. The vector distributions show that there is a separation region occurring upstream of the leading edge. It is shown that the region of the separation depends on the pitch-width ratio, b/C_a . A lower b/C_a generates a relatively larger separation region. This characteristic determines the location of the saddle points on the blade-to-blade plane depending on the pitch-width ratio b/C_a . The present computations show that there is a saddle point upstream of the leading edge attributed to the blockage of the blades. With increasing blockage, the secondary saddle vortices move more forward and the separation in the front of the blade becomes smaller. The location of the saddle point depends on the b/C_a as well as the local curvature of the blade.

Figures 3.19, 20, and 21 show the contours of the static pressure coefficient C_{ps} distributions at 90%, 100%, and 110% chord positions, respectively, from the leading edge for different values of b/C_a . The measurement results for $-C_{ps}$ at 90% chord position is plotted for comparison with the calculations. It should be pointed out that the measurement plane in the experiments [44] is defined to

Figure 3.20. Static pressure coefficient contour at $x/C_a = 1$.Figure 3.21. Static pressure coefficient contour at $x/C_a = 1.1$.

be perpendicular to the axial direction while the computations are along the mesh direction, which causes a slight discrepancy. Figure 3.19 shows that the computation and measurement are in good agreement with each other. The results for different b/C_a show that for smaller b/C_a , the lowest pressure becomes higher than those with larger pitch cases. The core of the secondary vortex occurs at different locations for all three cases. For the smaller b/C_a , the secondary vortex appears closer to the corner of the hub on the suction side. For larger b/C_a , the vortex core moves away from the suction-side toward the pressure side. When compared to the different axial location of C_{ps} , it is shown that the vortex core is developed and moves away from the suction side, to the middle point between the suction and pressure-side surfaces. As the fluid moves out from the blade passage, the vortices become very small. This is because the blade-to-blade pressure gradient is reduced. This vortex core is part of the leading edge vortex. Initially, the vortex occurs very close to the suction-side wall and then it develops toward the exit plane. At the same time, the horseshoe vortex at the pressure-side leg, Ss2, mixes with the secondary flow and rapidly decays. This phenomenon is in good agreement with the measurements for the pitch-width influence in the linear cascade [40].

Figure 3.22. Total pressure loss contour at $x/C_a = 0.9$.Figure 3.23. Total pressure loss contour at $x/C_a = 1.0$.

This figure also shows that the corner vortex at the tip of the blade is not very strong and cannot be clearly observed in the computations.

The total pressure losses C_{po} in Figures. 19, 20, and 21 further confirm the aforementioned conclusions that corner vortices occur on both the hub and the casing suction-side corners. The computations for the case of $b/C_a = 0.973$ are compared with the experiment [44] and are shown in Figure 3.19. The measured C_{po} values of the first line in the experiment are the same as the computations. It is shown that the computations are in fair agreement with the experiments, except at tip and hub regions where the computations show a larger vortex region than that shown by the experiment.

The influence of the losses due to the pitch-width variation can be seen by comparing Figures 3.22, 23, and 24 for different b/C_a . For the cases of $b/C_a = 0.973$ and 1.46, the high-loss regions from both tip and hub corners near the suction side are developed. These losses are caused due to the corner vortices and the suction-side pressure characteristics. The end wall boundary layer is very thin near the pressure side within the blade passage. The pressure-side boundary layer and the suction-side boundary layer meet after the flow passes the blade trailing edge, as shown in Figure 3. 20. It is also shown that the total pressure losses C_{po} increases with the increase in b/C_a . The loss contour shows that the secondary flow is stronger for larger b/C_a , and it develops faster. This is because for large b/C_a , the flow control capacity become worse than that for small b/C_a . The secondary vortex is easy to

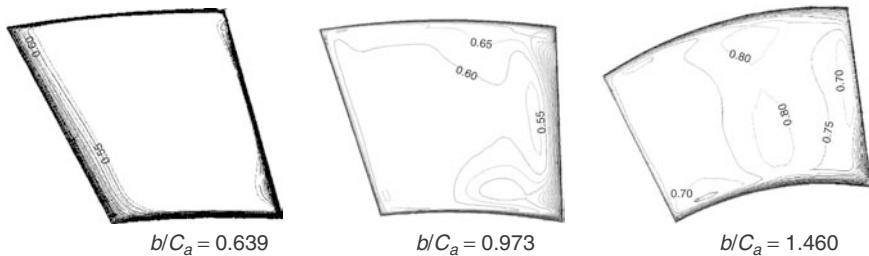


Figure 3.24. Total pressure loss contour at $x/C_a = 1.1$.

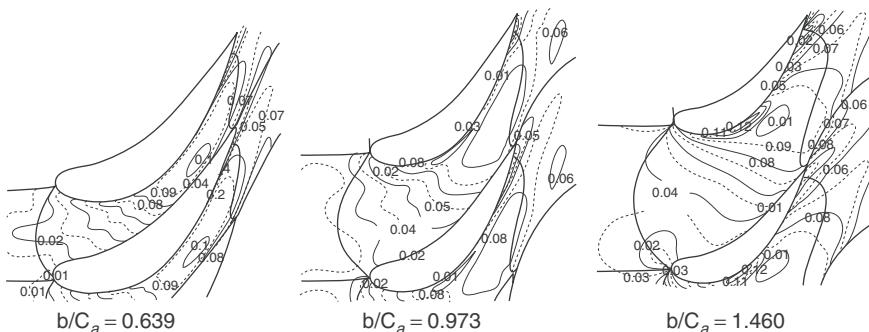


Figure 3.25. Blade-to-blade Mach number contour at 0.28% height.

develop in the blade channel. For the smallest b/C_a , the two passage vortices occur in the upper and lower parts of the blade passage closer to the suction side of the blade without strong interactions. With the increase in the b/C_a , the two vortices start interacting with each other. Owing to the occurrence of the passage vortices, and their interactions for the large b/C_a cases, the secondary flow is very strong. The strong three-dimensional effects dominate the whole passage flow for large b/C_a . However, for smaller b/C_a , it has been found that the flow configuration is different from those for larger b/C_a cases. The two vortices occur from lower and upper parts of the suction side, and they are more confined in the end wall region, and there remains a two-dimensional flow pattern for a wide blade height range.

The blade-to-blade Mach number contours for the hub boundary layer region and the pitch and tip boundary layer regions are shown in Figures 3.25, 26, and 27, respectively, where the Mach number contours present different characteristics for the boundary region and the main flow region. In the near-hub boundary region, a smaller b/C_a does not show a high Mach number region near the suction surface. It seems that the flow accelerates smoothly along the suction side for smaller b/C_a . For the cases of $b/C_a = 0.973$ and 1.460 , a small region of a high Mach number zone is observed close to the middle of the width in the suction side of the blade. It is shown that a separation zone occurs and expands with the increase in b/C_a .

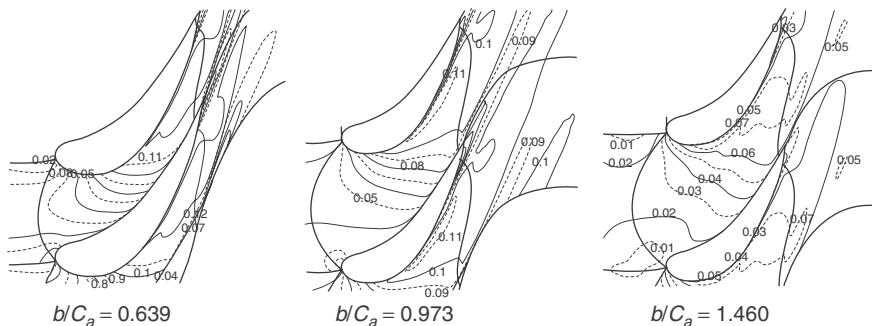


Figure 3.26. Blade-to-blade Mach number contour at 50% height.

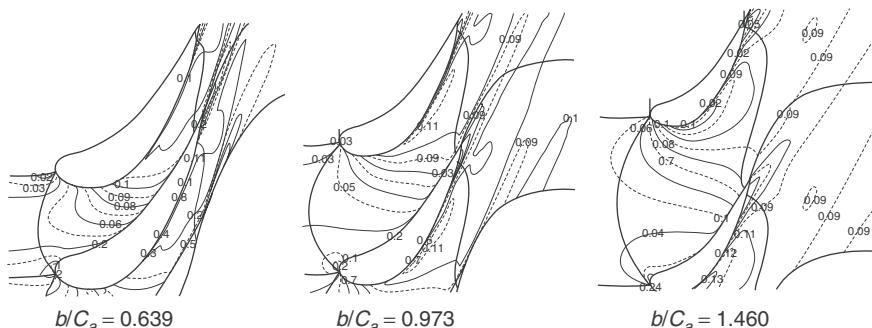


Figure 3.27. Blade-to-blade Mach number contour at 99.72% height.

The results show that the blade-to-blade Mach number gradient along the turbine axial direction increases with the increase in b/C_a .

The blade-to-blade static pressure coefficient contours near the hub, pitch, and tip are shown in Figures 3.28, 29, and 30, respectively. It is shown that, due to the strong secondary-flow effect, the minimum pressure occurs in the passage away from the blade surface. Also note that, with the increase in b/C_a , the flow accelerates faster in the leading edge region. A clear low-pressure zone occurs on the suction side of the blade for all the different pitch-width cases when a flow separation zone on the blade suction surface.

Figures 3.31, 32, and 33 show the contours of the blade-to-blade total pressure coefficient distributions in the hub boundary (0.28% height), pitch (50% height), and tip boundary (99.72% height) regions, respectively. The losses for different b/C_a , the plots start with the same level at the same blade location. The total pressure losses are much larger in hub and tip boundary regions than in the midspan for all cases. In general, small b/C_a has small losses for all flow regions. For the smaller b/C_a , a small zone of high loss extending from trailing edge can be observed only

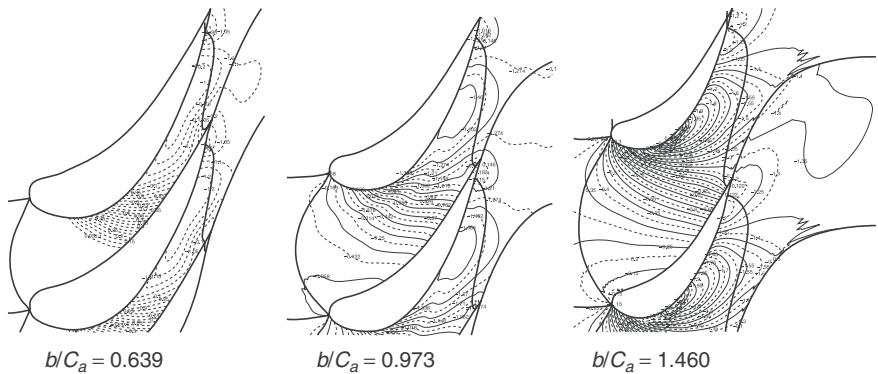


Figure 3.28. Blade-to-blade static pressure coefficient contour at 0.28% height.

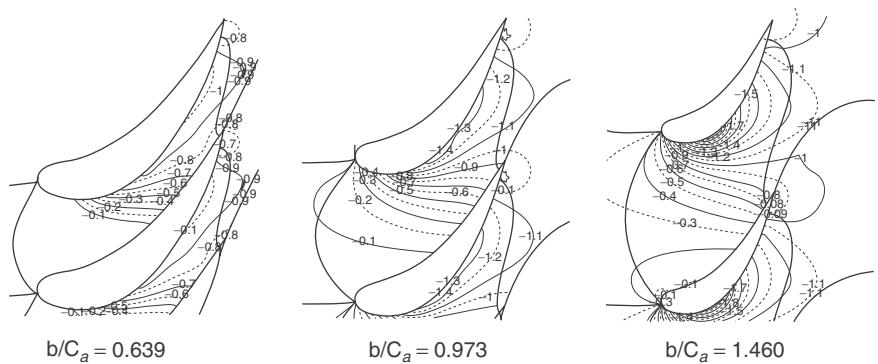


Figure 3.29. Blade-to-blade static pressure coefficient contour at 50% height.

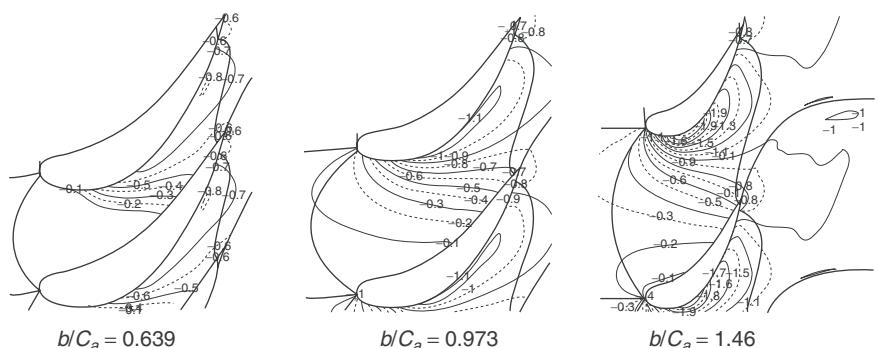


Figure 3.30. Blade-to-blade static pressure coefficient contour at 97.72 height.

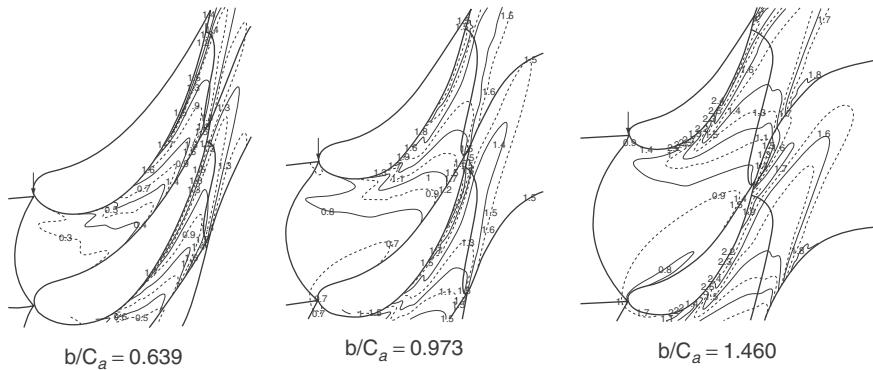


Figure 3.31. Blade-to-blade total pressure loss at 0.28% height.

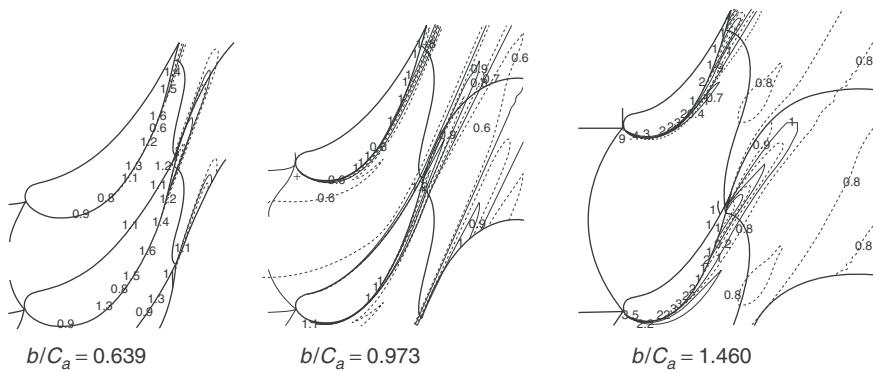


Figure 3.32. Blade-to-blade total pressure loss at 50% height.

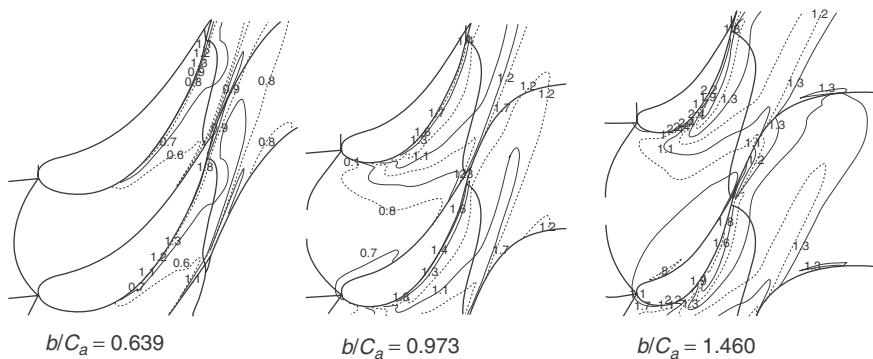


Figure 3.33. Blade-to-blade total pressure loss at 99.72% height.

on the tip boundary. For a larger b/C_a , there appears a high-loss zone starting from the trailing edge extending to the upstream of the suction side of the blade.

3.9.2 Flow around centrifuge compressors scroll tongue

Centrifugal compressors utilize radial change across the impeller to raise gas compressor. A centrifugal compressor allows more diffusion within the impeller blades to raise pressure; thus, centrifugal compressor can produce higher specific work transfer within one stage compared to an axial compressor. Although the efficiency of centrifugal compressor is lower than that of axial compressor at large volumetric flow rate, centrifugal compressor has many applications for moderate- and small-flow gas compressions. A centrifugal compressor has a shorter axial length with wider operating range than axial compressors to get the same discharge pressure. An increased efficiency of centrifugal compressors has been the traditional emphasis of the compressor design. However, an increasing operating range and part load efficiency have become more important. Many efforts have been made in research field to optimize different component of the compressor stage [45–51].

An industrial centrifugal compressor stage consists of three main components: a rotating impeller, vane or vaneless diffusers, and a scroll or collector as shown in Figure 3.34. The impeller adds work to gas by increasing its angular momentum. Gas static pressure and velocity increase after passing through the impeller. Vaned and vaneless diffusers convert kinetic energy into static pressure by decelerating the fluid. A scroll is used for collecting gas from diffuser and delivering it to application system or next stage. The scroll is the last component in the compression process in a single-stage compressor. Gas leaves impeller with logarithmic spiral shape due to the rotation of the impeller. A scroll has two functions: to collect and transport the fluid into downstream system and to raise the static pressure by converting kinetic energy into potential energy (static pressure). A scroll design is complicated due to the effect of the strong three-dimensional swirl flow. Typically, a scroll design

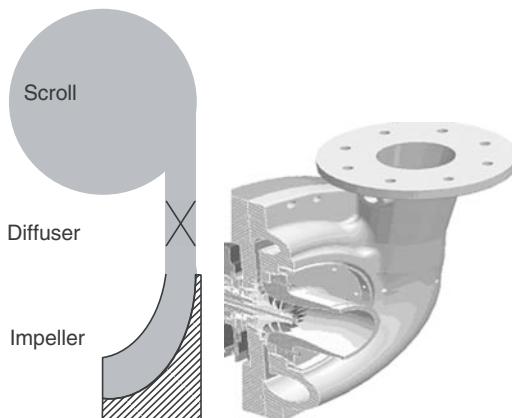


Figure 3.34. Centrifugal compressor.

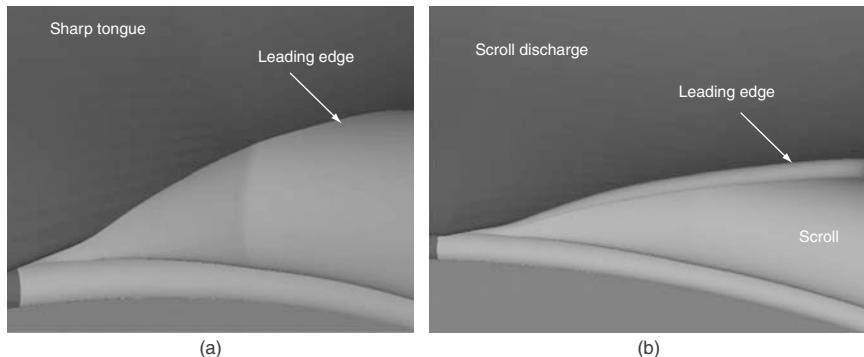


Figure 3.35. (a) Sharp tongue. (b) Scroll tongue structures.

is based on the two-dimensional analysis. Traditionally the emphasis on the scroll design is mainly on collection and less on diffusion function. Diffusion inside of scroll is more effective for the kinetic energy associated with the tangential velocity; this is accomplished by increasing cross-sectional area from scroll tongue to exit, creating a conical diffuser. But there is an efficiency loss caused by the inability of the scroll to convert portion of the kinetic energy due to the radial component of fluid velocity out of the diffuser. This effect is compounded when a vaned diffuser is used, as vanes create a more radial flow at the vaned diffuser exit. Typically, there is a tradeoff between the increased recovery of vaned diffuser and the reduced recovery of scroll; a different solidity vaned diffuser is used to meet performance and operating range requirements.

At a scroll tongue as shown in Figure 3.35, the flow encounters a dividing plane, so-called tongue, above which the flow exits the compressor and below which the flow reenters the scroll. The tongue of a scroll is located at a 360 degree point, corresponding to the joint of scroll smallest area, largest area, and exit cone. The leading edge of the scroll tongue is analogous to that of an airfoil; a sharp leading edge has a less blockage effect but is sensitive to the angle of attack as shown in Figure 3.35a, while a well-rounded leading edge is insensitive to the flow incidence but has a larger blockage effect as shown in Figure 3.35b. A sharp tongue is very sensitive to the flow incidence, and causes a flow separation at off-design flows. The manufacturing process is very difficult for the construction of a sharp tongue. A rounded tongue, however, has a larger blockage area than a sharp tongue, which may reduce efficiency at design point. With the understanding of the influence of the tongue, a compressor with higher efficiency and wider stable operating region can be designed. For obtaining a better off-design performance, the research on a scroll tongue was designed with a large cutback angle as shown in Figure 3.35b, which allows recirculation of about 25% of the stage discharge mass flow inside of the scroll.

Little research has been done to understand flows inside of a scroll, especially in the tongue area [52–54]. Recent research [55–59] found that a good scroll design could improve the compressor performance and the operating range. Most scroll

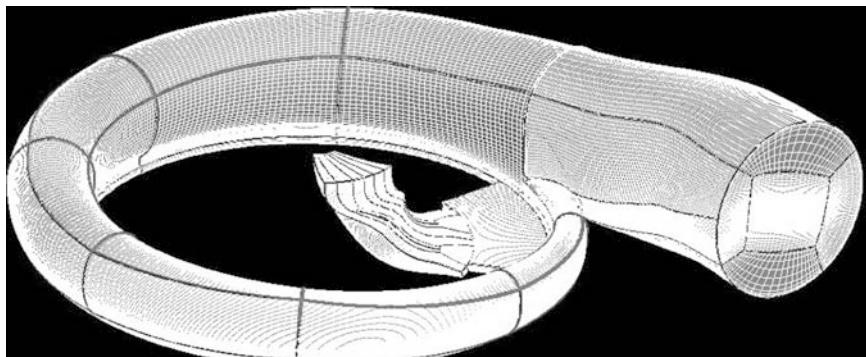


Figure 3.36. Calculation meshes.

studies have been performed on existing geometries and can be categorized into two groups: scroll flow structure investigations [52–55] and flow interaction studies between the scroll and the impeller [55–59]. Large portions of the scroll studies were performed for pumps. There is little information available for the scroll design and the scroll parametric influences on the compressor performance. Recent pump scroll studies [59] showed that the tongue shape impacts head, pressure fluctuations, and noise in a centrifugal pump. Irabu et al. [60] studied the water flow patterns around a pump tongue at small flows. The study found that flow separations occurred from both upper and lower surfaces of the tongue, and further reported the flow field influence due to different tongue geometries and focused on the detailed flow analysis besides performance test on a large cutback tongue scroll.

Turbulence modeling and turbulence model selections are critical to the accuracy of the flow field simulations. A larger number of CFD codes for a turbomachinery analysis use the zero equation turbulence model [1,2]. Most calculations showed that zero equation turbulence models gave reasonable results for flow calculations. Some researchers have demonstrated that other turbulence models can also be used to gain reasonable results [61,62]. Some comparisons for different turbulence models have been presented [63]. Different turbulence models showed different advantages. For the flow inside of the centrifugal compressor and scroll, different turbulence models also had been used for flow calculations. There are very limited publications that address the comparisons of different turbulence models.

Two types of tongues, a sharp tongue ($r/b_2 = 0$) and rounded tongue ($r/b_2 = 0.3$), were investigated. The calculation meshes are shown in Figure 3.36. The CFD studies for both tongue shapes were performed by using the zero-equation turbulence model for comparisons of the tongue shape impacts. The CFD analysis was also performed for rounded shape tongue volute by using the $k-\varepsilon$ turbulence model to investigate the model effects of the turbulence equations in the CFD simulations. In the current study, six flow rates were calculated for volutes with different tongues to provide the performance information. The computed performances for a round tongue with two different turbulence models are shown in Figure 3.37.

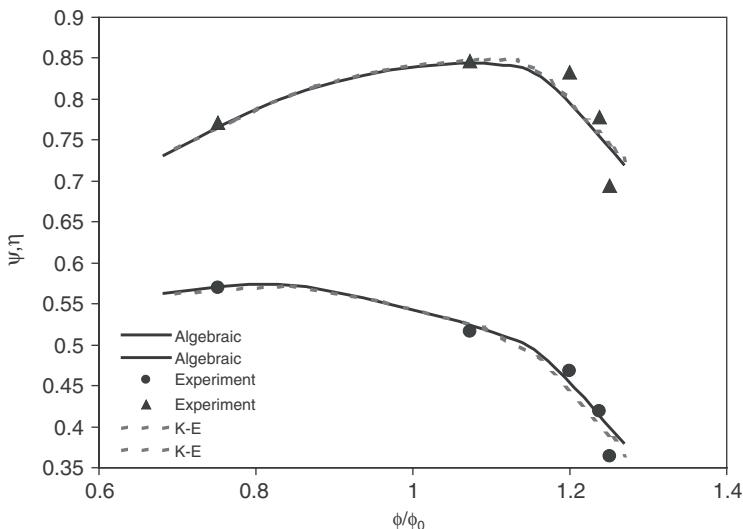


Figure 3.37. Comparison of the performance curves.

The calculations using the different turbulence models did not produce significant differences in the results of performance. The results from both two turbulence models are in good agreement with the experimental data. The calculations show that only at the flow coefficient near choke, more choke margin occurred than the experimental data. The calculations also show that the two-equation turbulence models give slightly higher efficiency at design flow point compared with the zero-equation turbulence model. Furthermore, the k - ε turbulence equations predicted a lower head coefficient at a larger flow and a lower flow compared with the zero-equation turbulence model.

However, the rounded tongue has less advantage over the sharp tongue when a compressor operates at a design flow rate. This is because when a compressor operates at around the design flow condition, the flow near the sharp tongue area has little incidence and the flow passes the sharp tongue smoothly. Therefore, the sharp tongue produces small blockage, whereas the rounded tongue produces larger blockage when the compressor operates near the design flow condition. As a result, at the design flow condition, the rounded tongue has less pressure recovery when compared with the sharp tongue case. However, when a compressor operates at a flow rate larger than the design flow rate, a scroll tongue encounters a positive incidence. Because a rounded tongue is not sensitive to the flow incidence, incidence losses for a rounded tongue are smaller than those for a sharp tongue. The rounded tongue produces better pressure recovery than the sharp tongue. When the compressor operates at a low flow rate, the advantages of the sharp tongue reduce with decrease in the flow rate. Because the tongue negative incidence increases with the decrease in compressor flow rate, the sharp tongue blockage increases faster than the rounded tongue blockage. Figure 3.40 shows the nondimensional turbulent

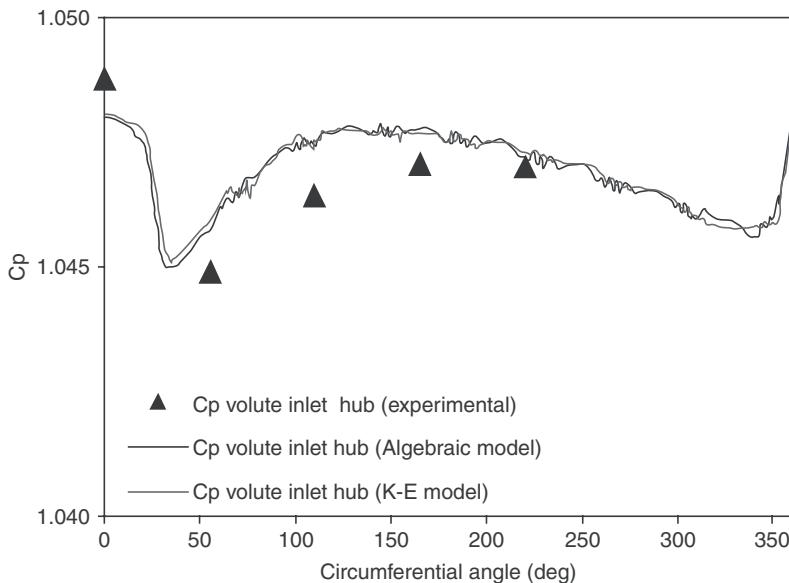


Figure 3.38. Static pressure distribution along volute inlet hub wall.

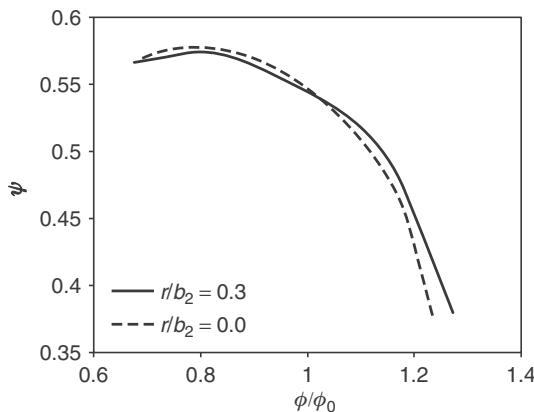


Figure 3.39. Performance characteristic.

viscosity (μ_T/μ) distributions obtained by employing the $k-\varepsilon$ turbulence equations. It can be seen that, for a lower flow rate and design flow rate cases, the turbulent viscosity is uniformly distributed excepted near the tongue area. For the lower flow rate, we can see a higher turbulence level behind the diffuser vane. Therefore, in a flow rate near surge and design, the zero-equation turbulence model can provide a reasonable prediction of flow characteristics.

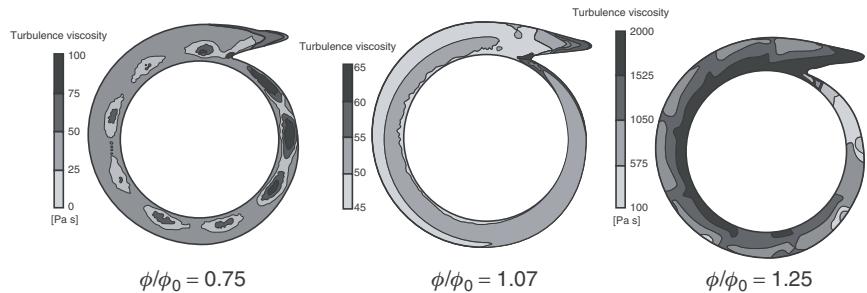


Figure 3.40. Turbulent viscosity (μ_τ/μ) contours at midplane of volute.

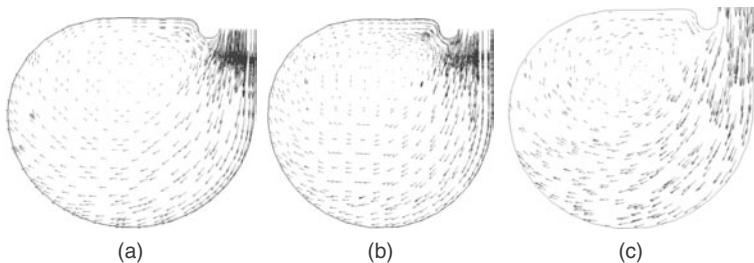


Figure 3.41. Velocity vector at $\theta = 180$ degrees ($\phi/\phi_0 = 1.07$). (a) Sharp tongue (zero-equation model) (b) Rounded tongue (zero-equation model) (c) Rounded tongue ($k-\varepsilon$ model).

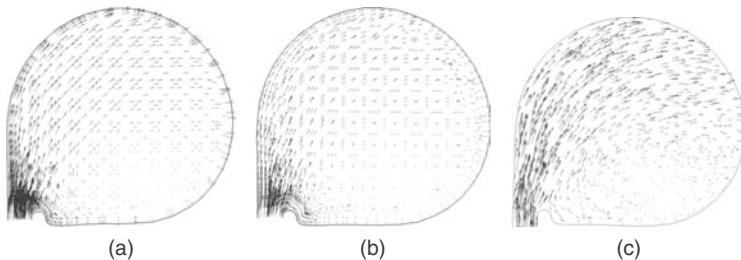


Figure 3.42. Velocity vector at scroll exit ($\theta = 360$ degrees ($\phi/\phi_0 = 1.07$)). (a) Sharp tongue (zero-equation model); (b) Rounded tongue (zero-equation model); (c) Rounded tongue ($k-\varepsilon$ model).

In the investigation of the tongue shape influence of the compressor, the sharp tongue scroll model was analyzed by using the zero-equation turbulence model. Figures 3.41 and 42 show secondary flow vectors at cross sections $\theta = 180$ and 360 degrees for a sharp tongue and a rounded tongue scrolls at a design flow condition. It can be seen that the secondary flow in the scroll sections has a single vortex structure

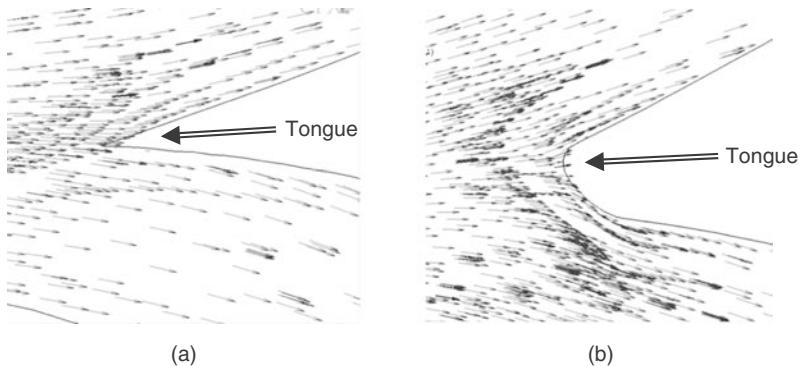


Figure 3.43. Velocity vectors near tongue intersection at midplane ($\phi/\phi_0 = 1.07$).
 (a) Sharp tongue. (b) Rounded tongue.

in both tongue geometries. It can also be seen that the tongue shape changes the secondary flow structures. The secondary flow impacted by the tongue occurs not only near the tongue area but also in other sections of the scroll. The rounded tongue creates a relatively large blockage near the tongue area. The large blockage forces the secondary flow center away from the tongue area. At both $\theta = 180$ and 360 degree locations, the secondary flow is stronger for the rounded tongue compared with the sharp tongue. The secondary flow center for sharp tongue occurs in the region closer to the tongue compared with that for the rounded tongue. As can be observed in these figures, the difference between the two turbulence models is very small.

The flow structures near the tongues are similar for both tongues at a design flow condition seen in Figure 3.43. The flow leaves the tongue leading edge smoothly at a design flow rate for both cases. No separation zone is found. It is also found that the sharp tongue has a smaller blockage than the rounded tongue (Figure 3.43). The flow near the leading edge of the rounded tongue is blocked by the tongue and then pushed away from the tongue.

Figures 3.44 and 45 show computed static pressure contours for the scroll with rounded tongue near the design flow rate at cross sections $\theta = 180$ degree and 360 degree at a design condition with different turbulence models. Figure 3.46 shows the exit plane total pressure contours at the design condition. It can be seen that the $k-\varepsilon$ turbulence model predicted lower static pressure and higher total pressure. However, the differences of the mass average pressures are less than 1%, as indication in Figure 3.46. The total pressure distributions are asymmetrical for both cases, and the symmetrical total pressure distributions are caused by secondary flows. The location of the highest total pressure range is different for the two calculations. It is attributed to the usage of the different turbulence models, which resulted in different secondary flow predictions.

Figure 3.47 shows the velocity vectors near the tongue intersection at a midplane and the secondary velocity vectors near the tongue trailing edge near the choke flow rate $\phi/\phi_0 = 1.25$. It is observed that the flows separate near the tongue for both

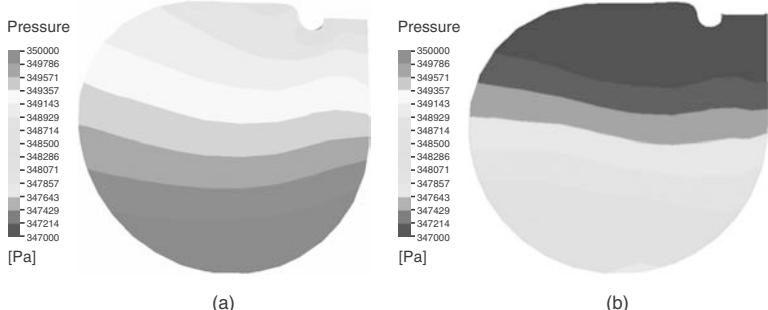


Figure 3.44. Static pressure contour at $\theta = 180$ degree ($\phi/\phi_0 = 1.07$). (a) Zero-equation model. (b) $k-\epsilon$ model.

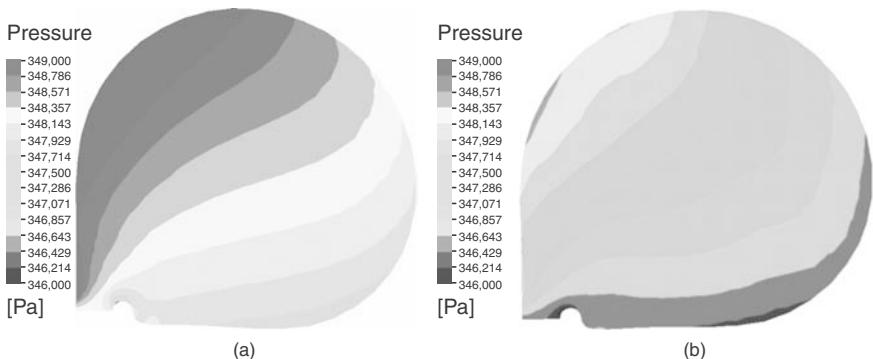


Figure 3.45. Static pressure contour at scroll exit ($\theta = 360$ degree ($\phi/\phi_0 = 1.07$)).
(a) Zero-equation model. (b) $k-\epsilon$ model.

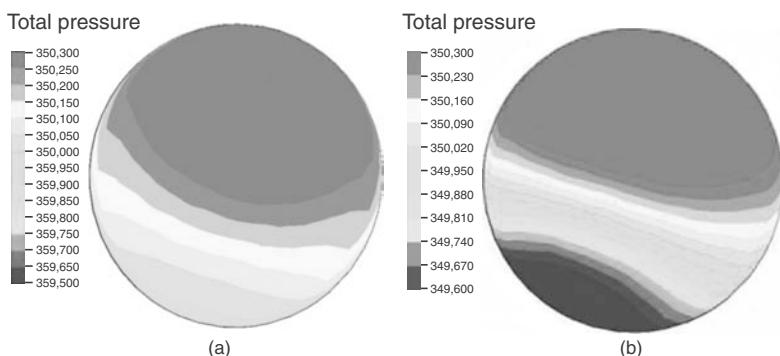


Figure 3.46. Total pressure contour at discharge of the exit cone ($\phi/\phi_0 = 1.07$).
(a) Zero-equation model. (b) $k-\epsilon$ model.

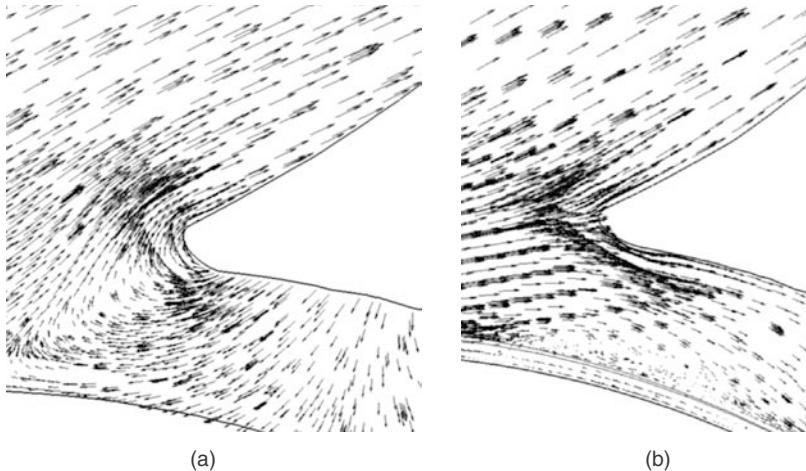


Figure 3.47. Velocity vectors near tongue intersection at midplane ($\phi/\phi_0 = 1.25$).
 (a) Zero-equation model. (b) $k-\varepsilon$ model.

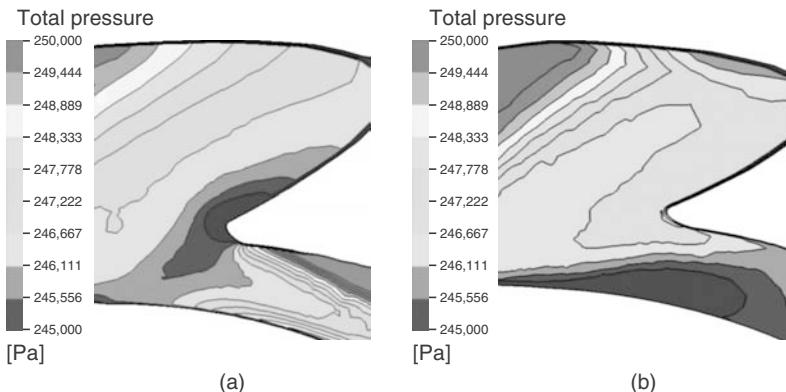


Figure 3.48. Total pressure contours near tongue intersection at midplane ($\phi/\phi_0 = 1.25$). (a) Zero-equation model. (b) $k-\varepsilon$ model.

cases. The scroll tongue encounters a large incidence because high flow rate results in large flow angles from the tangential direction. A large incidence results in flow separations from near the leading edge of the tongue. The $k-\varepsilon$ turbulence model predicts a recirculation zone near the inlet of the volute, and the zero-equation turbulence model predicts a larger recirculation zone. Total pressure contours near the tongue intersection at midplane are shown in Figure 3.48.

Total pressure distributions near the tongue are different between two different turbulence model predictions. The zero-equation model predicts the lowest total pressure zone near the tongue. However, the $k-\varepsilon$ turbulence model predicts the lowest total pressure in the region, which is located near the volute inlet tongue area.

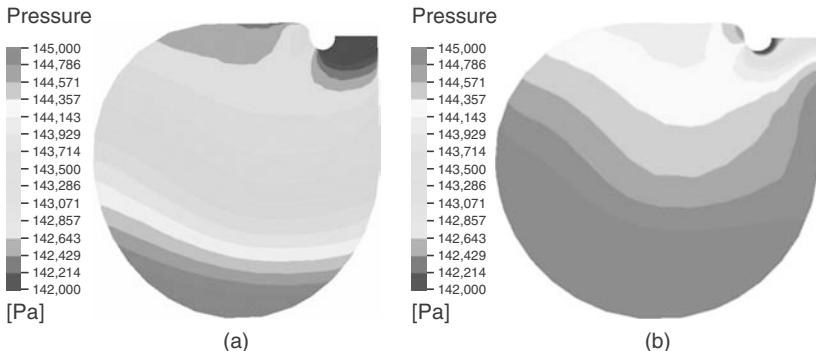


Figure 3.49. Static pressure at choke flow at $\theta = 180$ degrees ($\phi/\phi_0 = 1.25$).
(a) Zero-equation model. (b) $k-\epsilon$ model.

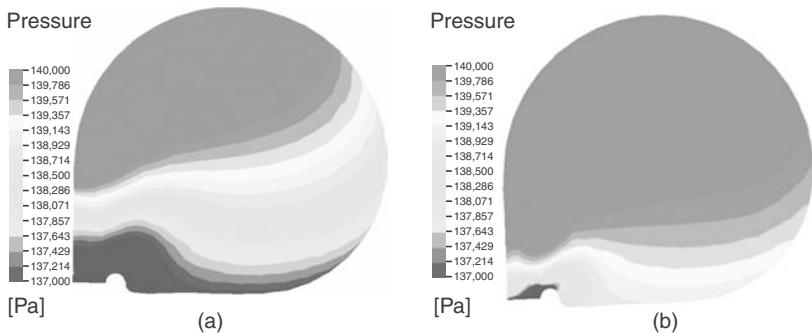


Figure 3.50. Static pressure at choke flow at $\theta = 360$ degree ($\phi/\phi_0 = 1.25$).
(a) Zero-equation model. (b) $k-\epsilon$ model.

The computed static pressure contours near the choke flow rate at cross sections $\theta = 180$ and 360 degrees are shown in Figures 3.49 and 50. It can be seen that the $k-\epsilon$ model predicted higher static pressure level than the zero-equation model. The total pressure distributions at the scroll exit are shown in Figure 3.51. The total pressure level predicted using the $k-\epsilon$ model is higher than that using the zero-equation model. The lowest total pressure zone location appears to be different for the two computational cases. This is because these two turbulence models result in different strengths of the secondary flow. The total pressure distributions predicted using the $k-\epsilon$ model showed the boundary-layer effects; that is, there is a lower total pressure zone around the discharge pipe wall. Figure 3.52 shows velocity vectors near the tongue intersection at midplane near the surge flow rate $\phi/\phi_0 = 0.75$. Flow vectors near the tongue indicate a continuous flow passing through the tongue and into the exit duct. Flow vectors also indicate that due to the large recirculation flow, the pressure side of the tongue does not show significant flow separation. The large cutback tongue with significant flow circulation inside the scroll part can delay the flow separation near scroll tongue at a small flow rate. Two types of the

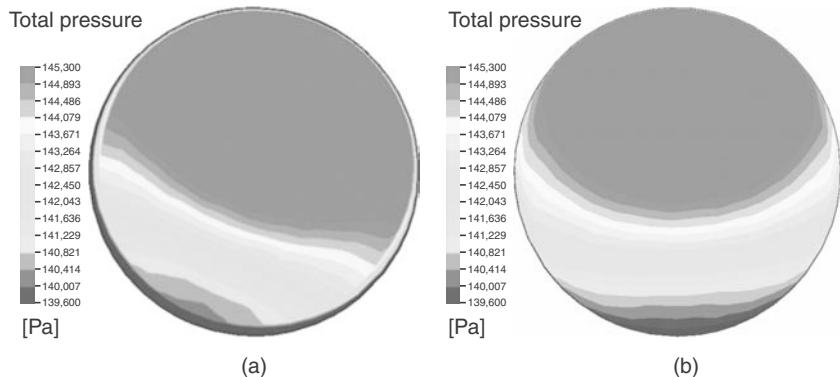


Figure 3.51. Total pressure discharge of exit cone at choke flow ($\phi/\phi_0 = 1.25$).
(a) Zero-equation model. (b) $k-\varepsilon$ model.

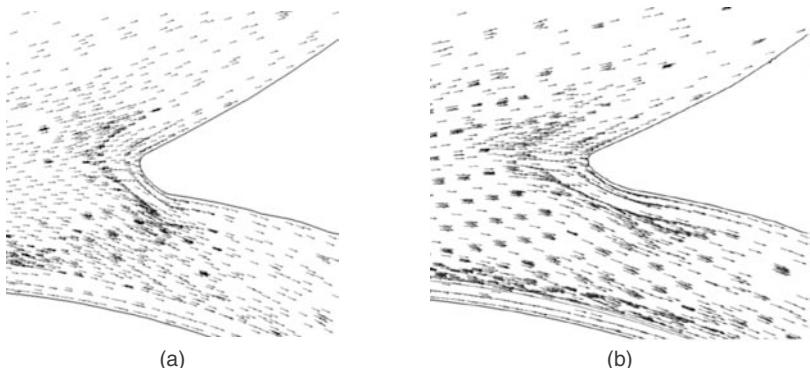


Figure 3.52. Velocity vectors near intersect diffuser and tongue near surge condition ($\phi/\phi_0 = 0.75$). (a) Zero-equation model. (b) $k-\varepsilon$ model.

turbulence model provided very similar results. Total pressure distributions near the scroll tongue at the midplane area are shown in Figure 3.53. Total pressure distributions show that the $k-\varepsilon$ model predicted a slightly higher level of the total pressure. The performance test also confirms this. Static pressure contours at cross sections $\theta = 180$ and 360 degrees are shown in Figures 3.54 and 55. It is shown that the $k-\varepsilon$ model predicted a lower level of static pressure than the zero-equation turbulence model. Total pressure distributions as shown in Figure 3.56 indicate that the $k-\varepsilon$ model predicted higher total pressure levels than the zero-equation model. However, the differences between two calculations are insignificantly small. Again, the boundary-layer effects can be seen in the $k-\varepsilon$ turbulence model calculations.

The calculations showed that, by using different turbulence models, the performance prediction results did not make significant difference. In these calculations, the impeller and the vaned diffuser as well as scroll tongue have relatively sharp

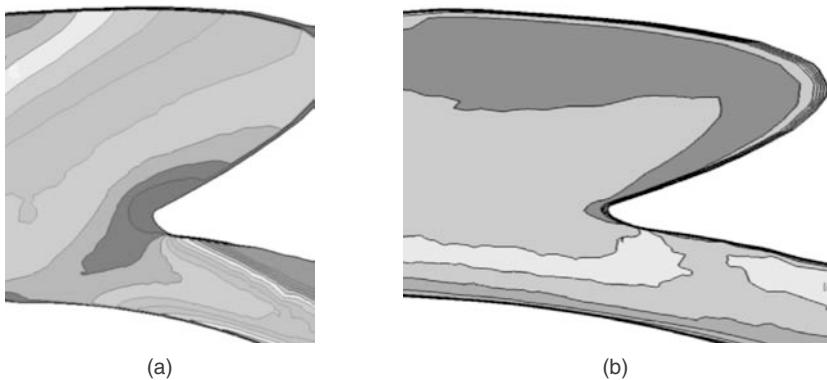


Figure 3.53. Total pressure contours near tongue intersection at midplane ($\phi/\phi_0 = 0.75$). (a) Zero-equation model. (b) k - ϵ model.

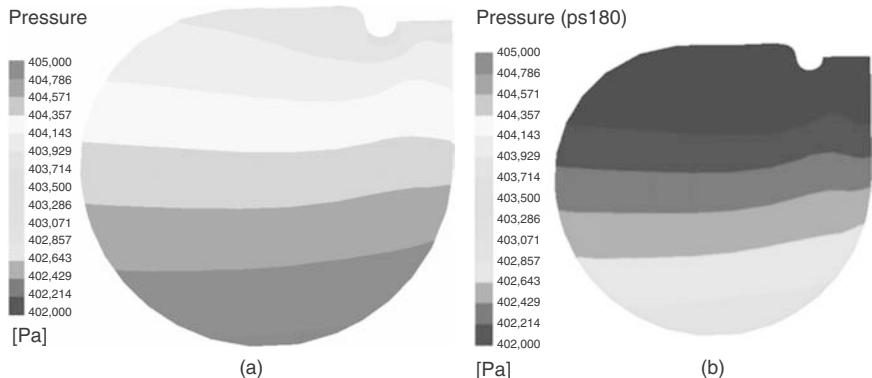


Figure 3.54. Static pressure at surge condition at $\theta = 180$ degrees ($\phi/\phi_0 = 0.75$). (a) Zero-equation model. (b) k - ϵ model.

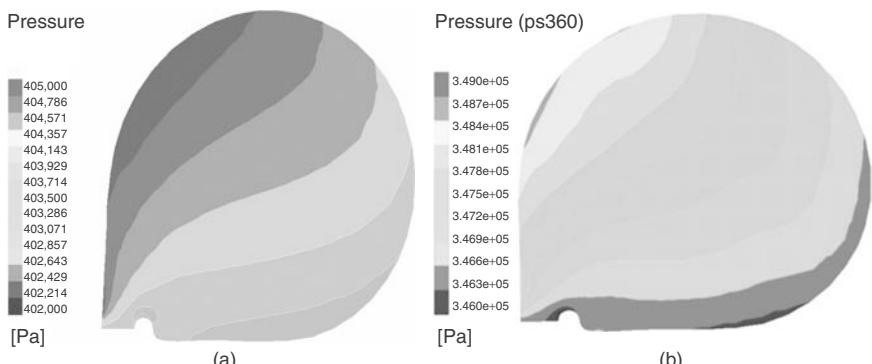


Figure 3.55. Static pressure at surge flow at $\theta = 360$ degrees ($\phi/\phi_0 = 0.75$). (a) Zero-equation model. (b) k - ϵ model.

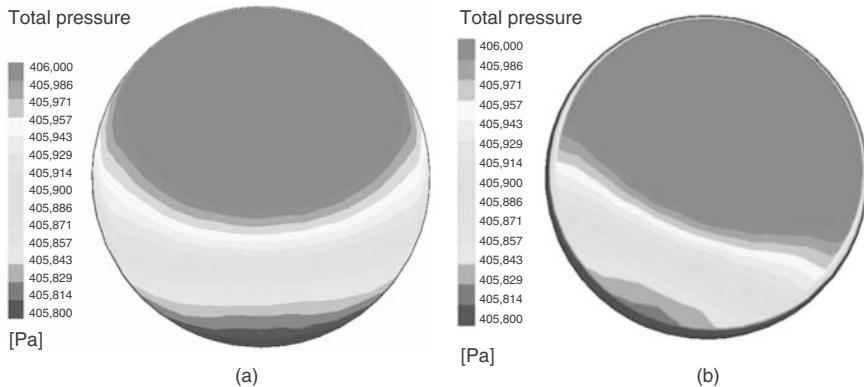


Figure 3.56. Total pressure discharge at exit cone at surge flow ($\phi/\phi_0 = 0.75$).
 (a) Zero-equation model. (b) $k-\epsilon$ model.

leading edges. The flow losses may mainly be dependent on the flow angle separations; this type of flow separation is not sensitive to the turbulence models.

3.10 CFD Applications in Turbomachine Design

With the development of the computer hardware, the CFD has been implemented in the turbomachine design process. Enormous efforts are devoted to improving the efficiency of the gas turbine components. The designs for turbine and compressor airfoils play one of the most important roles in increasing the turbine efficiency. In the airfoil design, there are two types of implementations the aerodynamic design engineer often considers. One is to design and employ custom-design blade profiles with minimum losses and controlled blade boundary-layers. The second and even more complex part is to minimize losses resulting from secondary flows near hub and casing. Recently three-dimensional blade design concepts that help to control secondary flows were proposed [1]. However, the complex flow is difficult even with fully three-dimensional Navier–Stokes flow solvers. And the validation of the N–S solvers need to take large numbers of experimental data, which is time consuming and expensive. Therefore, almost all the aerodynamic designs are based on the two-dimensional design. The inviscid analyses of the two-dimensional airfoil sections still play an important role in the design process.

It is known that for a blade row in an annulus, the stream surface between two annular walls is twisted for most cases. These twists are induced by either shed vorticity or Secondary flow arising from. Stream surface twist can arise in an irrotational flow owing to either spanwise components of velocity or spanwise blade forces. Many efforts have been taken to reduce the stream surface twist and the secondary flow losses, such as sweep, lean, bow, and twist the blades or design an asymmetric end wall. However, there is little information available in the literature for using three-dimensional design and almost no information available to show how to integrate the three-dimensional features into the design process. Most of the

turbomachinery design system investigations were still on the academic research and were based on the particular machines or blades. Moreover, most of the studies were based on the particular blade and flow situations. For example, Singh et al. [14] argued that closing the blade throat near the end walls could obtain significant efficiency improvements, and Wallis and Denton [20] also obtained an efficiency increase from almost the opposite type of blade twist near the end wall. For different machines and different designs, many different techniques should be used according to the flow field nature of the designs. It is very important to design a blade design procedure and optimize the design.

The increased use of CFD tools has been driven mainly by two factors. First, from performance standard point of view, efficiency has steadily increased. Second, the turbomachinery industry as a whole has been pushed toward reduced cost designs. The cost reduction is in terms of development, modification, production, and operating costs. The cost reduction drives a turbomachine toward high loading in order to reduce stage count, while maintaining or exceeding past performance goals. The current design of the new stage already is outside of the standard airfoil database. Most of the airfoil needs to be designed. And the development of the design tools to meet this requirement becomes critical. The application of CFD methodology to improve the turbomachinery design is becoming established within the turbomachinery community. However, still only limited number of publications suggest how to use CFD to help design and modify processes especially during the blade design. This chapter serves to present a design process that contains a novel two- and three-dimensional viscous turbulent code and optimization process.

Expensive manpower is invested in order to find configurations that are stable and efficient in the work range in the turbine and compressor designs. One of the most important methods is inverse design or called stream line design where two-dimensional blade profiles are to be found to insure the desired working range stability and efficiency. During the design, the constraints arising from aerodynamics, aeromechanical, mechanical, heat transfer, and manufacturing considerations have to be satisfied.

The design of turbine and compressor blade had made a great progress. Many advanced design methods and CFD tools have been incorporated into the design procedure. However, most of the design procedures focus only on the flow prediction, and there are few papers that describe the overall design processes and design implements. For example, Wellborn and Delaney [5] described a compressor design system in Rolls-Royce, which comprises three tools: through-flow, three-dimensional isolated blade, and three-dimensional multistage predication. The turbomachinery design is an integrated process that contains a process from meanline, through-flow, airfoil design, and analysis. This paper developed a design process that can be easily adapted by the industry.

The aerodynamic design procedures for turbomachinery airfoil used in this study is shown in Figure 3.57. A design system includes meanline analysis, through-flow analysis, airfoil section design, airfoil stackup, three-dimensional blade row, and multistage flow analysis. For obtaining the highest design efficiency, the optimizer was used to do the section optimization. The three-dimensional CFD

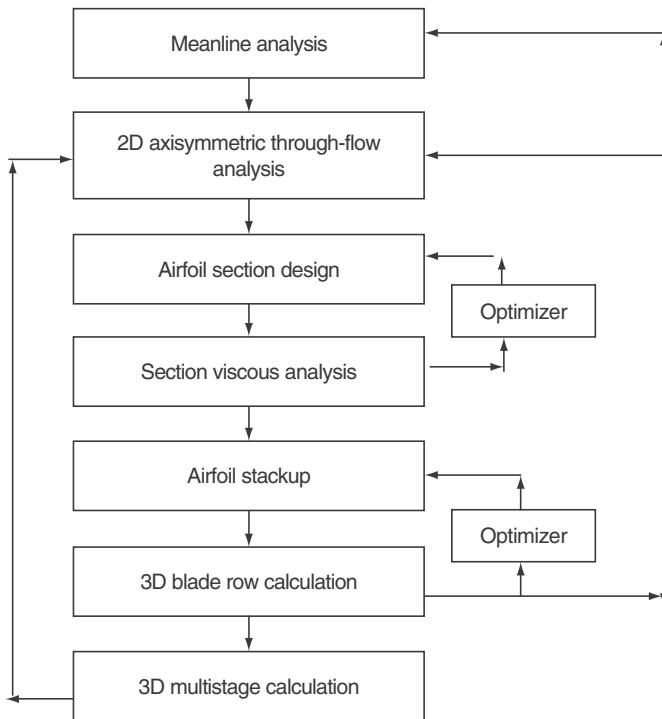


Figure 3.57. Blade design and optimization procedure.

code was used for blade stackup optimization. The optimizer may be used for three-dimensional blading although authors do not encourage the use of optimizer for three-dimensional optimization.

Meanline analysis determines the loading of the stage and annular area. It plays an important role in the turbomachinery design. The meanline design for the first-stage compressor and the last-stage turbine is critical. The enthalpy rise for compressor and drop for turbine are fixed by the meanline analysis. The overall machine character is determined by the meanline analysis. The compressor and turbine efficiency is strongly influenced by meanline design.

Through-flow analysis is one of the preliminary design modules. The axisymmetric streamline curvature calculations can be used to optimize the overall parameters of a multistage turbomachine. This module establishes the definitions of the flow path and work distributions in radial direction. The velocity diagrams at design point for different blade rows and different streamlines are determined. The optimization code can be used to do the optimization for selecting the best design parameters, for example, stage loading and stage enthalpy change.

The airfoil section design is a very important step for the aerodynamic design. All the airfoils are designed by the section design and reasonable stackup. A three-dimensional blade method was developed in this study as shown in Figure 3.57. It can be seen that CFD code for both section analysis and three-dimensional blade row analysis is very important. The development of the efficient and wide-range application of CFD codes is very important. In this study, an effective numerical method was developed for both two-dimensional and three-dimensional codes. In this chapter, a brief introduction for the numerical method developed for two-dimensional incompressible and compressible flows was provided.

3.10.1 Flow solver for section analysis

A time-marching algorithm was used in the present cascade flow computations [2,20,22,23,29]. The computation starts with a rough perturbation, which develops under certain boundary conditions. In this approach, the governing equations are replaced by a time-difference approximation with which steady or time-dependent flows of interest can be solved at each time level. The details of the governing equations and the method of solving the equations are discussed in previous sections.

The discussion has been extended for many years for the optimum type of the mesh that should be used for turbine or compressor blade flow calculation [15,16]. The more orthogonal the grid, the smaller will be the numerical errors due to the discretization. However, no one type of grid is ideal for blade-to-blade flow calculation. In this study, the H-type mesh is used. Another problem for the blade-to-blade mesh is the trailing edge mesh and the trailing edge Kutta condition [7]. The authors have noticed that the number of mesh points near the trailing edge points strongly influence the loss calculation. Here, a realistic method is proposed; that is, when the mesh on the trailing edge circle is more than 10 points, an explicit viscous Kutta condition [17] is applied on the training edge circle which allows flow leaving the blade smoothly.

The mesh refinement study was conducted prior to the calculations [2,3]. It is shown that the mesh size of 110×45 with 80 points on the blade surface in the blade-to-blade section is sufficient. The H-mesh [2,3] was used in all the computations. The computational mesh is shown in Figure 3.58. A typical convergence history of the calculation is shown in Figure 3.59. It is demonstrated that the present code has a good convergence. In this study, this code was used in the two-dimensional airfoil section analysis and optimization.

3.10.2 Optimization

Turbine and compressor designs faced the challenge of high efficiency and reliable blade design. Design techniques are typically based on the engineering experience and may involve much trial and error before an accepted design is found. The ability of the three-dimensional codes and the ability to perform a three-dimensional flow calculation are difficult to improve the design. The three-dimensional approach is a way to help designers do some parameter study and compare the numerical

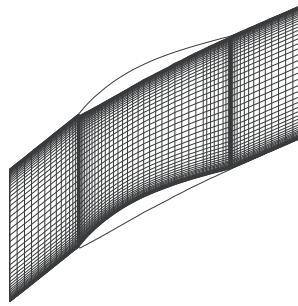


Figure 3.58. Computational mesh.

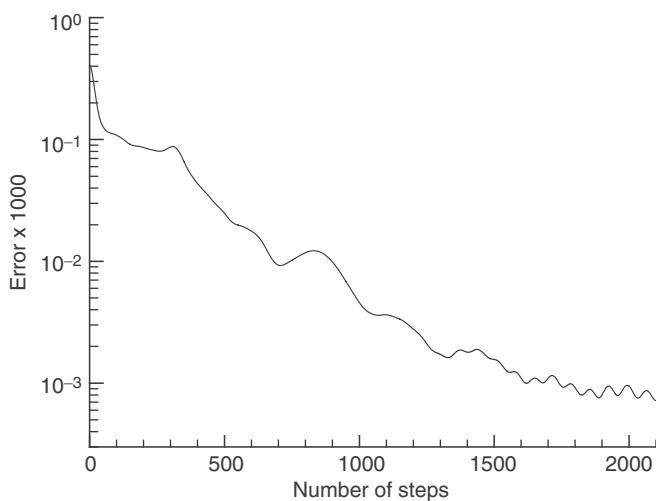


Figure 3.59. Typical convergence history.

study with testing in order to improve the design. The three-dimensional method is used for parameter studies, like lean, bow, and sweep effects. The two-dimensional analysis developed in the study is used to optimize blade section airfoil.

The blade design often starts with two-dimensional airfoil section. Some of the design parameters are obtained by through-flow optimization and vibration and heat transfer analysis, for example, the solidity and number of blades. And most of the parameters are optimized through the sectional design. For example, leading- and trailing-edge radii, stagger angle, and maximal thickness position. The final three-dimensional analyses are used to make lean, bowed, and sweep modification of the blades.

The optimization process used in this study was based on the constrained optimization method [26]. If the objective function to be minimized is $F(\mathbf{X})$ and constraint functions is $g_j(\mathbf{X})$, the optimization problem can be formulated by the

objective and constraint functions as:

$$\nabla F(\mathbf{X}) + \sum \lambda_j \nabla g_j(\mathbf{X}) = 0 \quad (78)$$

where λ_j are the Lagrange multipliers and \mathbf{X} is the vector of the objective. The finite differencing method can be used to obtain the gradients of objective and constraint functions with respect to design variables and constraints. Many commercial optimization packages are available as an optimizer to design problems [26].

The objective function of this study is adiabatic efficiency. The optimization objective is to obtain the highest efficiency under given constraints. For more convenience, the total pressure loss is used as an objective variable to judge the design. The definition of the total pressure loss in this study is:

$$\xi = \frac{(p_{\text{in}}^* - p_{\text{out}}^*)}{\Delta H} \quad (79)$$

where ΔH is the outlet dynamic head of the exit plane.

Two-dimensional section optimization

In this study, a compressor stator blade originally stacked up by using NACA 65 was redesigned and optimized. In this design, the thickness of the airfoil, chord, stagger angle, and inlet and outlet flow parameters were fixed. The designed blade was used to replace the old blade. The section design optimization was mainly achieved through adjusting the section maximum thickness location. It was found that the maximum thickness locations influence the losses and performance. In this study, five sections were selected as design section. All section designs used the similar method and procedure. As an example, the results presented here are the design information for 50% span section.

Figures 3.60 through 62 show the Mach number distributions of the design blades with different maximum thickness locations. Figure 3.63 shows the relationship between total pressure losses and location of the maximum thickness. It was shown that the Mach number distributions changed with the maximum thickness. The change in the Mach number distributions will influence the boundary-layer development and influence the losses of the section. It was found that there is an optimum location of the maximum thickness. Studies have also shown that this location changes with flow conditions and stagger of the airfoil. After optimization study, the airfoil section with highest efficiency was selected. The airfoil was stacked up using a smooth leading edge curve method. Before the airfoil was selected as the final blade, the three-dimensional analysis was performed to investigate the benefits of the three-dimensional blade features. Three-dimensional analysis showed that after section optimization the airfoil performance was improved, as shown in Figure 3.64. However, the end wall region did not show significant improvement. Some three-dimensional features were used to reduce the losses and increase the efficiency.

Three-dimensional blading

Three-dimensional CFD analysis codes were used to guide the three-dimensional final modification of the two-dimensional section stackup. Some parameters were

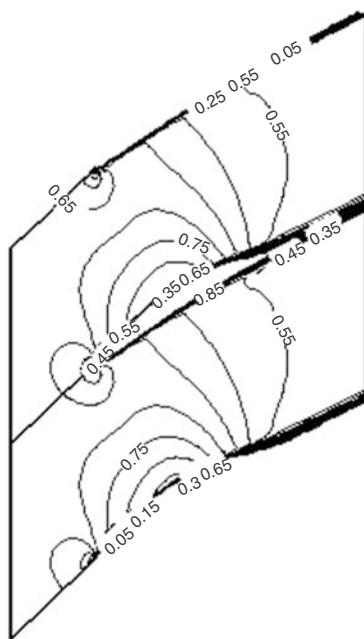


Figure 3.60. Mach number distributions for maximum thickness at 15% chord.

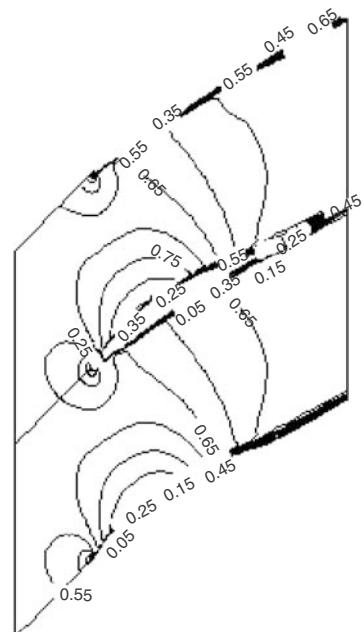


Figure 3.61. Mach number distributions at maximum thickness at 25% chord.

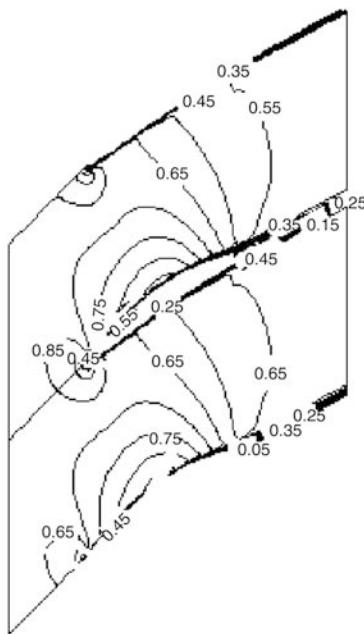


Figure 3.62. Mach number distributions at maximum thickness at 45% chord.

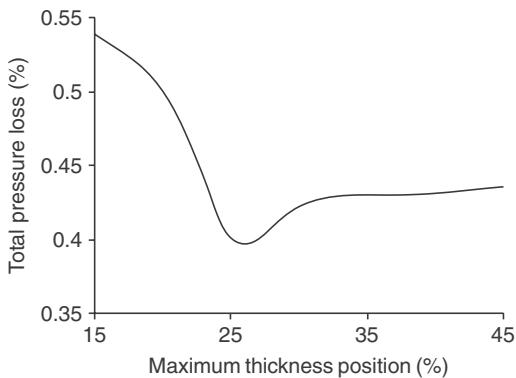


Figure 3.63. Total pressure losses Vs maximum thickness location.

selected to do the study based on design experience. In this redesign, the three-dimensional feature-bowed blade was used to reduce the secondary losses. In the three-dimensional blading, the optimizer will not be used in this study, because optimization with optimizer like the three-dimensional code is very time consuming. Although the mesh sizes, turbulence models and mixing between the blade rows strongly influence the optimization results, the process propose here provide a way to drive a better design. In the current study, three-dimensional optimization was obtained through parameter study.

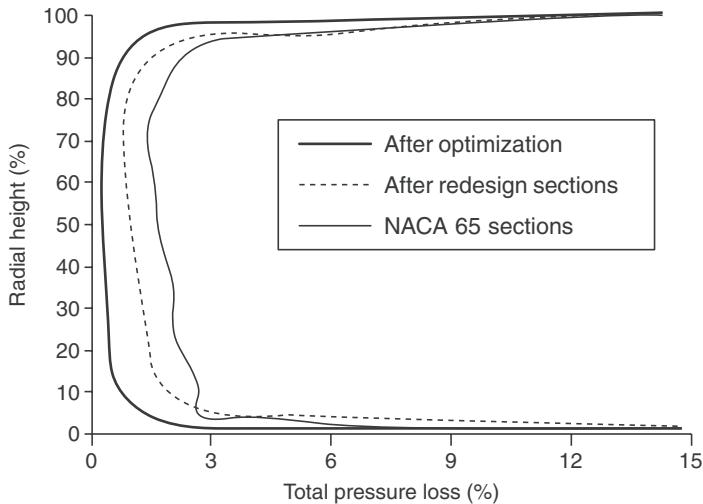


Figure 3.64. Total pressure losses along with blade span.

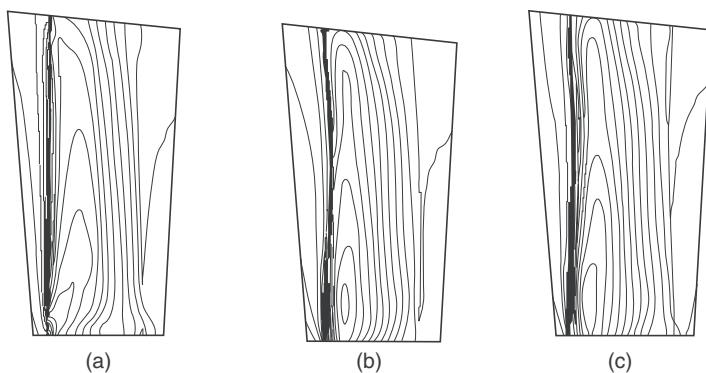


Figure 3.65. Static pressure distribution on the suction surface. (a) Original NACA section blade. (b) Before 3D optimization. (c) After 3D optimization.

In this study, only bow feature is applied. The bow location and degree of the bow were selected as study parameters. It was found that the 15 degree bow located at 30% of the span can eliminate separation and get better performance. The three-dimensional analysis showed that after two-dimensional section optimized design, the flow in most of the spanwise locations remained attached and well behaved although it had a small region of end wall separation on the section side, as shown in Figures 3.65 through 68. After the bow was introduced the flow around the airfoil on both tip and root end wall was attached. The total pressure losses were reduced, as shown in Figure 3.64. This study showed that the total section efficiency increased by about 3% as compared with the original blade. Both two-dimensional

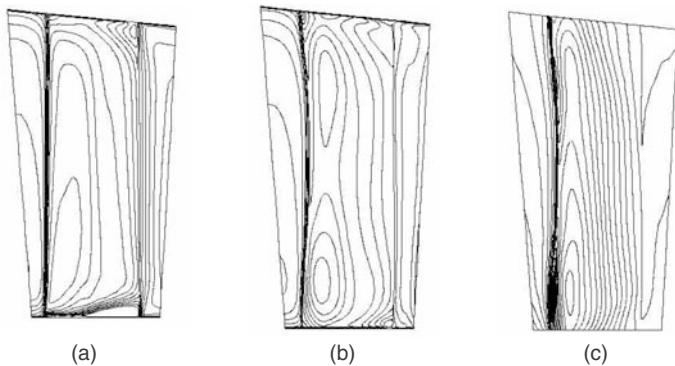


Figure 3.66. Axial velocity contour near blade suction surface. (a) Original NACA section blade. (b) Before 3D optimization. (c) After 3D optimization.

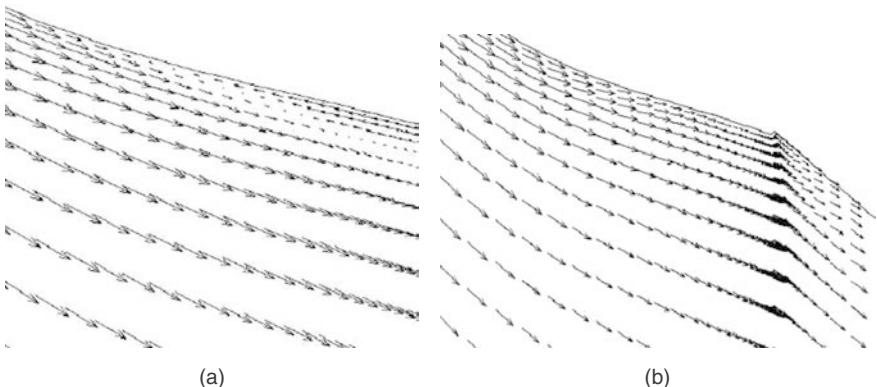


Figure 3.67. Velocity vector near the root section near the trailing edge suction surface. (a) Before 3D optimization. (b) After 3D optimization.

section design and three-dimensional optimization increased the efficiency with similar percentage. It has been shown that both two-dimensional optimization and three-dimensional study are important. However, three-dimensional blading depends more on the experience of the designer.

It is important to point out that, based on the design experience, the three-dimensional analysis results normally are not accepted as final shape. The selection of the final configuration was accepted through small adjustments based on experience to obtain high performance and stability. After three-dimensional analysis, some small adjustments were made based on the possible manufacturing uncertainties and applications. For example, the compressor stator design in this study was recambered to increase exit angle by about 1.5 degrees to increase the compressor surge margin.

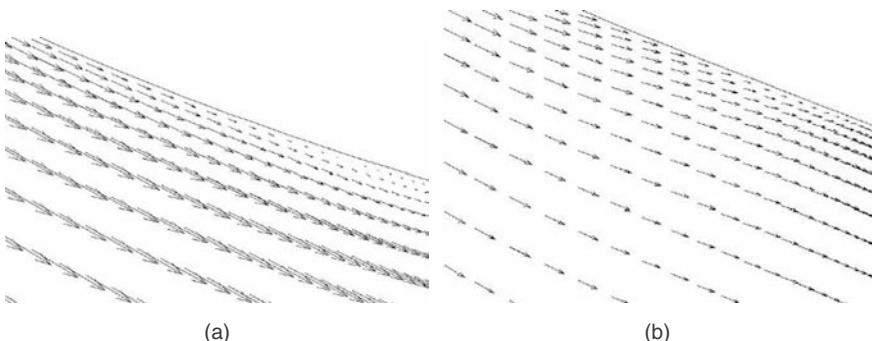


Figure 3.68. Velocity vector near the tip section near the trailing edge suction surface. (a) Before 3D optimization. (b) After 3D optimization.

The flow field around turbine and compressor blades exhibits very complex flow features. The flow field involves various types of loss phenomena, and hence high-level flow physics is required to produce reliable flow predictions. The blade design process is a very time consuming process and the optimization process is very complicated. The method developed in this study for the airfoil section and blade design and optimization is one successful process and very easy for adapting in the aerodynamic design. In this study, a two-dimensional code was used for section design instead of three-dimensional code, which provides an economic way for design and optimization. The three-dimensional blades were optimized after stacking up the two-dimensional sections. The lean, bowed, and swept features of the three-dimensional blade were created during three-dimensional optimization. It was shown that both two- and three-dimensional design and analysis were the backbones of the blade aerodynamic designs. The results show that both the airfoil shape optimization and the three-dimensional optimization can be used to improve the performance of compressor and turbine blade.

References

- [1] Xu, C., and Amano, R. S. On the Development of Turbine Blade Aerodynamic Design System, ASME and IGTI Turbo Expo, 2001-GT-0443. New Orleans, LA, USA, June 4–7, 2001.
- [2] Xu, C., and Amano, R. S. A Hybrid Numerical Procedure for Cascade Flow Analysis, *Numer. Heat Transfer, Part B*, 37(2), pp. 141–164, 2000.
- [3] Xu, C., and Amano, R. S. An Implicit Scheme for Cascade Flow and Heat Transfer Analysis, *ASME J. Turbomach.*, 122, pp. 294–300, 2000.
- [4] Japikse, D. Centrifugal Compressor Design and Performance, Concepts ETI, Inc., Wilder VT, USA, 1996.
- [5] Wellborn, S. R., and Delaney, R. A. Redesign of a 12-Stage Axial-Flow Compressor Using Multistage CFD, 2001-GT-345, ASME and IGTI Turbo Expo, New Orleans, LA, USA, June 4–7, 2001.

- [6] Lakshminarayana, B. An Assessment of Computational Fluid Dynamics Techniques in the Analysis and Design of Turbomachinery—the 1990 Freeman Scholar Lecture, *ASME J. Fluids Eng.*, 113, pp. 315–352, 1991.
- [7] Xu, C. Kutta Condition for Sharp Edge Flows, *Mech. Res. Commun.*, 25(4), pp. 425–420, 1998.
- [8] Xu, C. Surface Vorticity Modeling of Flow Around Airfoils, *Comput. Mech.*, 22(6), pp. 1998.
- [9] Xu, C., and Yeung, W. W. H. Discrete Vortex Method for Airfoil with Unsteady Separated Flow, *J. Aircraft*, 33(6), pp. 1208–1210, 1996.
- [10] Xu, C., Yeung, W. W. H., and Gou, R. W. Inviscid and Viscous Simulations of Spoiler Performance, *Aeronaut. J.*, 102(1017), pp. 399–405, 1998.
- [11] Xu, C., and Yeung, W. W. H. Unsteady Aerodynamic Characteristics of Airfoil with Moving Spoilers, *J. Aircraft*, 36(3), pp. 530–538, 1999.
- [12] Xu, C., and Yeung, W. W. H. Numerical Study of Unsteady Flow Around Airfoil with Spoiler, 65(1), 164–171, 1998.
- [13] Michos, A., Bergeles, G., and Athanassiadis, N. Aerodynamic Characteristics of NACA0012 Airfoil in Relation to Wind Generators, *Wind Eng.*, 7, pp. 1–8, 1985.
- [14] Singh, G., Walker, P. J., and Haller, B. R. Development of 3D Stage Viscous Time-Marching Method for Optimisation of Short Stage Heights, Proceedings of the European Conference on Turbomachinery, Erlangen, 1995.
- [15] Turkel, E., and Vatsa, V. N. Effect of Artificial Viscosity on Three-Dimensional Flow Solutions, *AIAA J.*, 32(1), pp. 39–45, 1994.
- [16] Lin, H., Yang, D. Y., and Chieng, C. C. Variants of Biconjugate-Gradient Method for Compressible Navier–Stokes Solver, *AIAA J.*, 33(7), pp. 1177–1184, 1995.
- [17] Weingold, H. D., Neubert, R. J., Behlke, R. F., and Potter, G. E. Bowed Stators: An Example of CFD Applied to Improve Multistage Compressor Efficiency, *J. Turbomach.*, 119, 1997.
- [18] Sasaki, T., and Breugelmans, F. Comparison of Sweep and Dihedral Effects on Compressor Cascade Performance, *ASME Paper No. 97-GT-2*, 1997.
- [19] Kawagishi, H., and Kawasaki, S. The Effect of Nozzle Lean on Turbine Efficiency, Proceedings of ASME International Conference on Joint Power Generation, October, 1991.
- [20] Wallis, A. M., and Denton, J. D. Comparison of design intent and experimental measurements in a low aspect ratio axial flow turbine with three-dimensional blading. Proceedings, ASME International Gas Turbine and Aeroengine Congress and Exhibition, Stockholm, Sweden, ASME Paper 98-GT-516 (June 1998).
- [21] Gonzalez, J., Fernandez, J., Blanco, E., and Santolaria, C. Numerical Simulation of Dynamic Effects due to Impeller-Volute Interaction in a Centrifugal Pump, *ASME J. Fluids Eng.*, 124, pp. 348–355, 2002.
- [22] Chima, R. V. Explicit Multigrid Algorithm for Quasi-Three-Dimensional Viscous Flows in Turbomachinery, *J. Propul. Power*, 3(5), pp. 397–405, 1987.
- [23] Inoue, M., and Furukawa, M. Artificial Dissipative and Upwind Schemes for Turbomachinery Blade Flow Calculations, *Numerical Methods for Flow Calculation in Turbomachines*, Von Karman Institute for Fluid Dynamics Lecture Series, 1994-06, 1994.
- [24] Li, H., Chen, S., and Martin, H. F. Two Dimensional Viscous Blade Flow Code Using Shifted Periodic Grids, *ASME Paper No. 97-GT-516*, 1997.
- [25] Vanderplaats, G. N. Numerical Optimization Techniques for Engineering Design, McGraw-Hill, New York, 1984.

- [26] Chung, J., Shim, J., and Lee, K. D. Shape Optimization of High-Speed Axial Compressor Blades Using 3D Navier–Stokes Flow Physics, *ASME and IGTI Turbo Expo*, 2001-GT-0594, New Orleans, LA, USA, 2001.
- [27] Dunham, J. Analysis of High Speed Multistage Compressor Throughflow Using Spanwise Mixing, *ASME Paper No. 92-GT-13*, 1992.
- [28] Farin, G. Curves and Surfaces for Computer Aided Geometrical Design: A Practical Guide, 3rd Edition, Academic Press, Inc., San Diego, 1993.
- [29] Arnone, A., and Swanson, R. C. A Navier–Stokes Solver for Turbomachinery Applications, *ASME J. Turbomach.*, 115(2), pp. 1556–1563, 1993.
- [30] Chima, R. V. Explicit Multigrid Algorithm for Quasi-Three-Dimensional Viscous Flows in Turbomachinery, *J. Propul. Power*, 3(5), pp. 397–405, 1987.
- [31] Chakravarthy, S. R., Anderson, D. A., and Salas, M. D. The Split Coefficient Matrix Method for Hyperbolic Systems of Gasdynamics Equations, *AIAA Paper No. 80-0268*, 1980.
- [32] Baldwin, B. S., and Lomax, H. Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows, *AIAA Paper 78-257*, 1978.
- [33] Amineni, N. K., and Engeda, A. Pressure Recovery in Low Solidity Vaned Diffusers for Centrifugal Compressors, 97-GT-472, 1997.
- [34] Hylton, L. D., Mihelc, M. S., Turner, E. R., Nealy, D. A., and York, R. E. Analytical and Experimental Evaluation of the Heat Transfer Distribution Over the Surfaces of Turbine Vanes, NACA CR-168015 DDA EDR 11209, 1983.
- [35] Kwon, O. K. Navier–Stokes Solution for Steady Two-Dimensional Transonic Cascade Flows, *ASME J. Turbomach.*, 110(2), pp. 339–346, 1988.
- [36] Younis, M. E., and Camarero, R. Finite Volume Method for Blade to Blade Flow Using a Body Fitted Mesh, *AIAA J.*, 19(11), pp. 1500–1502, 1981.
- [37] Denton, J. D. An Improved Time-Marching Method for Turbomachinery Flow Calculation, *ASME J. Eng. Gas Turbines Power*, 105, pp. 514–521, 1983.
- [38] Sieverding, C. H., ed. Description of Test Cases and Presentation of Experimental Results, Von Karman Institute for Fluid Dynamics Lecture Series, 1982-05, Rhode-Saint-Genese, Belgium, 1982.
- [39] Xu, C., and Amano, R. S. Computational Analysis of Pitch-Width Effects on the Secondary Flows of Turbine Blades, *Comput. Mech.*, 34(2), pp. 111–120, 2004.
- [40] Perdichizzi, A., and Dossena, V. Incidence Angle and Pitch-Chord Effects on Secondary Flows Downstream of a Turbine Cascade, *ASME J. Turbomach.*, 115, pp. 383–391, 1993.
- [41] Hodson, H. P., and Dominy, R. G. Three-Dimensional Flow in a Low-Pressure Turbine Cascade at Its Design Condition, *ASME J. Turbomach.*, 109, pp. 177–185, 1987.
- [42] Gregory-Smith, D. G., and Clerk, J. G. E. Secondary Flow Measurement in a Turbine Cascade with High Inlet Turbulence, *ASME Paper No. 90-GT-20*, 1990.
- [43] Kiss, T., Schetz, J. A., and Moses, H. L. Experimental and Numerical Study of Transonic Turbine Cascade Flow, *AIAA J.*, 34, pp. 104–109, 1996.
- [44] Langston, L. S., Nice, M. L., and Hooper, R. M. Three-Dimensional Flow within a Turbine Cascade Passage, *ASME J. Eng. Power*, 99, pp. 21–28, 1977.
- [45] Senoo, Y., Hayami, H., and Ueki, H. “Low-solidity tandem . . .” 1983, ASME Paper No. 83-GT-3. ASME Gas Turbine Conference, Phoenix, Arizona, USA.
- [46] Kenny, D. P. The History and Future of the Centrifugal Compressor in Aviation Gas Turbine, 1st Cliff Garrett Turbomachinery Award Lecture, Society of Automotive Engineers, SAE/SP-804/602, 1984.

- [47] Weber, C. R., and Koronowski, M. E. Meanline Performance Prediction of Volutes in Centrifugal Compressors, ASME 31st Gas Turbine Conference and Exhibit, Dusseldorf, Germany, 86-GT-216, 1987.
- [48] Dong, R., Chu, S., and Katz, J. Effect of Modification to Tongue and Impeller Geometry on Unsteady Flow, Pressure Fluctuations and Noise in a Centrifugal Pump, *ASME J. Turbomach.*, 119, pp. 506–515, 1997.
- [49] Arora, J. S. Introduction to Optimum Design, McGraw-Hill, NY, USA, 1998.
- [50] Duccio Bonaiuti, Andrea Arnone, Mirco Ermini, Analysis and Optimization of Transonic Centrifugal Compressor Impellers Using the Design of Experiments Technique, gt-2002-30619, 2002, Amsterdam, The Netherlands.
- [51] Harry, M. J. The Nature of Six Sigma Quality, Motorola University Press, Schaumburg, IL, USA, 1997.
- [52] Ayder, E. Experimental and Numerical Analysis of the Flow in Centrifugal Compressor and Pump Scrolls, *Ph.D. dissertation*, Von Karman Institute, 1993.
- [53] Ayder, E., and van den Braembussche, R. Numerical Analysis of the Three-Dimensional Swirling Flow in Centrifugal Compressor Scrolls, *ASME J. Turbomach.*, 116, pp. 462–468, 1994.
- [54] Ayder, E., van den Braembussche, R., and Brasz, J. J. Experimental and Theoretical Analysis of the Flow in a Centrifugal Compressor Scroll, *ASME J. Turbomach.*, 115, pp. 582–589, 1993.
- [55] Gu, F., Engeda, A., Cave, M., and Liberti, J. A Numerical Investigation on the Scroll/Diffuser Interaction due to the Axial Distortion at the Impeller Exit, *ASME J. Fluid Eng.*, 123, pp. 475–483, 2001.
- [56] Sideris, M. Circumferential Distortion of the Flow in Centrifugal Compressors due to Outlet Scrolls, *Ph.D. dissertation*, Von Karman Institute, 1998.
- [57] Gonzalez, J., Fernandez, J., Blanco, E., and Santolaria, C. Numerical Simulation of Dynamic Effects due to Impeller-Volute Interaction in a Centrifugal Pump, *ASME J. Fluids Eng.*, 124, pp. 348–355, 2002.
- [58] Xu, C., and Amano, R. S. Eliminating Static Pressure Distortion by a Large Cut Back Tongue Volute, Barcelona, Spain.
- [59] Chu, S., Dong, R., and Katz, J. Relationship between Unsteady Flow, Pressure Fluctuations, and Noise in a Centrifugal Pump—Part B: Effects of Blade-Tongue Interactions, *ASME J. Fluids Eng.*, 117, pp. 30–35, 1995.
- [60] Irabu, K., Tamazato, E., and Teruya, I. Velocity Measurement of Flow around a Scroll Tongue of the Vaneless Radial Diffuser, Proceedings of 3rd Asia Symposium on Visualization, Japan, 1994.
- [61] Hah, C., and Wennerstrom, A. J. Three-Dimensional Flowfields Inside a Transonic Compressor with Swept Blades, 90-GT-359, Brussels, Belgium 62. computational Fluid Dynamics Symposium on Aeropropulsion, NACA CP-10045, Apr. Cleveland, Ohio, 1990
- [62] Kunz, R. F., and Lakshminarayana, B. Computation of Supersonic and Low Subsonic Cascade Flows Using an Explicit Navier-Stokes Technique and the $k-\epsilon$ Turbulence Model, ANSA CP-10045, 1990.
- [63] Basi, F., Rebay, S., and Savini, M. “A Navier- ...”, AGARD-CP-527, “Heat Transfer and Cooling in Gas Turbines” Oct. 1992, pp. 41.1 to 41.16.

II. Finite Element Method

This page intentionally left blank

4 The finite element method: discretization and application to heat convection problems

Alessandro Mauro^{1,2}, Perumal Nithiarasu¹, Nicola Massarotti², and Fausto Arpino³

¹*Civil and Computational Engineering Centre, School of Engineering, Swansea University, Swansea, U.K.*

²*Dipartimento per le Tecnologie, Università degli Studi di Napoli “Parthenope”, Napoli, Italy.*

³*Dipartimento di Meccanica, Strutture, Ambiente e Territorio, Università di Cassino, Cassino, Italy.*

Abstract

In this chapter, we give an overview of the finite element method and its applications to heat and fluid flow problems. An introduction to weighted residual approximation and finite element method for heat and fluid flow equations are presented. The characteristic-based split (CBS) algorithm is also presented for solving the incompressible thermal flow equations. The algorithm is based on the temporal discretization along the characteristics, and the high-order stabilization terms appear naturally from this kind of discretization. The artificial compressibility (AC) and the semi-implicit (SI) versions of the CBS scheme are presented and some examples are also given to demonstrate the main features of both the schemes.

Keywords: Characteristic-based split, Finite elements, Porous media

4.1 Governing Equations

4.1.1 Non-dimensional form of fluid flow equations

In many heat transfer applications, it is often easy to generate data by non-dimensionalizing the equations, using appropriate non-dimensional scales. To obtain a set of non-dimensional equations, let us consider five different cases of convection heat transfer. We start with the mixed convection problem followed by the natural and forced convection problems. The turbulent flow and convective flow through porous media are also presented. For each case, we discuss one set of non-dimensional scales.

Mixed convection

Mixed convection involves features from both forced and natural flow conditions that are two limiting cases of mixed convection flow. The buoyancy effects become comparable to the forced flow effects at small and moderate Reynolds numbers. Since the flow is partly forced, a reference velocity value is normally known (example: velocity at the inlet of a channel). However, in mixed convection problems, the buoyancy term needs to be added to the appropriate component of the momentum equation.

In incompressible mixed convection flow problems, the following non-dimensional scales are normally employed:

$$\begin{aligned} x_i^* &= \frac{x_i}{L}, \quad u_i^* = \frac{u_i}{u_{\text{ref}}}, \quad t^* = \frac{t u_{\text{ref}}}{L^2}, \\ p^* &= \frac{p}{\rho u_{\text{ref}}^2}, \quad T^* = \frac{T - T_{\text{ref}}}{T_w - T_{\text{ref}}} \end{aligned} \quad (1)$$

where * indicates a non-dimensional quantity, L is a characteristic dimension, the subscript ref indicates a constant reference value and T_w is a constant reference temperature, for example, wall temperature. The density ρ and viscosity μ of the fluid are assumed to be constant everywhere and equal to the inlet value.

Using the above scales, the non-dimensional form of the incompressible mixed convection equations, in indicial notation, are written as:

Continuity equation

$$\frac{\partial u_i^*}{\partial x_i^*} = 0 \quad (2)$$

Momentum equation

$$\frac{\partial u_i^*}{\partial t^*} + u_j^* \frac{\partial u_i^*}{\partial x_j^*} = -\frac{\partial p^*}{\partial x_i^*} + \frac{1}{\text{Re}} \frac{\partial}{\partial x_i} \left(\frac{\partial u_i^*}{\partial x_j} \right) + \frac{\text{Gr}}{\text{Re}^2} T^* \gamma_i \quad (3)$$

where γ_i is a unit vector in the gravity direction.

Energy equation

$$\frac{\partial T^*}{\partial t^*} + u_i^* \frac{\partial T^*}{\partial x_i^*} = \frac{1}{\text{Re} \text{Pr}} \frac{\partial}{\partial x_i} \left(\frac{\partial T^*}{\partial x_i} \right) \quad (4)$$

where Re is the Reynolds number defined as:

$$\text{Re} = \frac{u_{\text{ref}} L}{\nu} \quad (5)$$

Pr is the Prandtl number given as:

$$\text{Pr} = \frac{\nu}{\alpha} \quad (6)$$

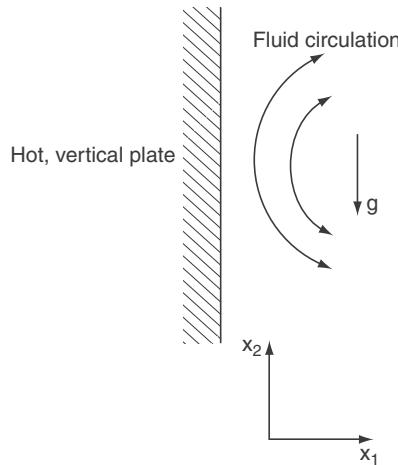


Figure 4.1. Natural convective flow near a hot vertical plate.

and Gr is the Grashof number given as:

$$Gr = \frac{g\beta\Delta TL^3}{\nu^2} \quad (7)$$

where $\nu = \mu/\rho$ is the kinematic viscosity. Appropriate changes will be necessary if an appreciable variation in these quantities occurs in a flow field.

Note that sometimes, a non-dimensional parameter referred to as the *Richardson number* (Gr/Re^2) is also used in the literature.

Natural convection

Natural convection is a limiting case of the more general case of mixed convection. Natural convection is generated only by the density difference induced by the temperature differences within a fluid system (Figure 4.1). Because of the small density variations present in these types of flows, a general incompressible flow approximation is adopted. In most buoyancy-driven convection problems, flow is generated either by a temperature variation or by a concentration variation in the fluid system, which leads to local density differences.

In practice, the following non-dimensional scales are adopted for natural convection in the absence of a real reference velocity value:

$$\begin{aligned} x_i^* &= \frac{x_i}{L}, \quad u_i^* = \frac{u_i L}{\alpha}, \quad t^* = \frac{t \alpha}{L^2}, \\ p^* &= \frac{p L^2}{\rho \alpha^2}, \quad T^* = \frac{T - T_{ref}}{T_w - T_{ref}} \end{aligned} \quad (8)$$

Upon introducing the above non-dimensional scales into the governing equations, and replacing Re with $1/\text{Pr}$ in the non-dimensional mixed convection equations of the previous subsection, we obtain the non-dimensional equations for natural convection flows.

Often, another non-dimensional number called the *Rayleigh number* is used in the calculations. This is given as:

$$\text{Ra} = \text{Gr Pr} = \frac{g\beta\Delta TL^3}{\nu\alpha} \quad (9)$$

Forced convection

Forced convection is another limiting case of mixed convection. In forced convection flow problems, the non-dimensional scales are the same as the mixed convection problem. The non-dimensional form of the forced convection governing equations is obtained from that of mixed convection by neglecting the buoyancy term on the right hand side of the momentum equation.

Another non-dimensional number, which is often employed in forced convection heat transfer calculations is the Peclet number and is given as $\text{Pe} = \text{RePr} = u_{\text{ref}} L/\alpha$.

Some examples

The following results are obtained solving the above equations for forced, natural and mixed convective flows by using an artificial compressibility (AC) version and a semi-implicit (SI) version of the characteristic-based split (CBS) algorithm and finite element discretization technique that will be explained later.

The first problem considered is the backward-facing step, that is a test case commonly used to validate numerical solutions of the incompressible Navier–Stokes equations. In fact, several data are available in literature for this problem (Denham and Patrik [1], Aung [2], Ichinose et al. [3], de Sampaio et al. [4], Massarotti et al. [5]). The computational domain and the boundary conditions used in this work are presented in Figure 4.2a. In particular, the velocity profile at the inlet is measured experimental profile (Denham and Patrik [1]). The measurement section in the experiments, as in the present calculations, was placed 1.3 times the step height upstream of the step. All the horizontal walls and the step have been imposed with no-slip velocity boundary conditions.

For non-isothermal cases, in addition to the above boundary conditions (Aung [2]), a non-dimensional temperature equal to one is imposed along the bottom wall, while the temperature of the inlet flow is assumed to be equal to zero. All other walls are assumed to be adiabatic. Figure 4.2 shows some details of the computational grids used for this problem.

Figure 4.3 shows the horizontal velocity distribution along different vertical sections of the channel, obtained by using the AC and the SI versions of the CBS algorithm.

The temperature profiles at two different vertical cross sections downstream of the step are presented in Figure 4.4 for the case of a fully developed inlet profile with

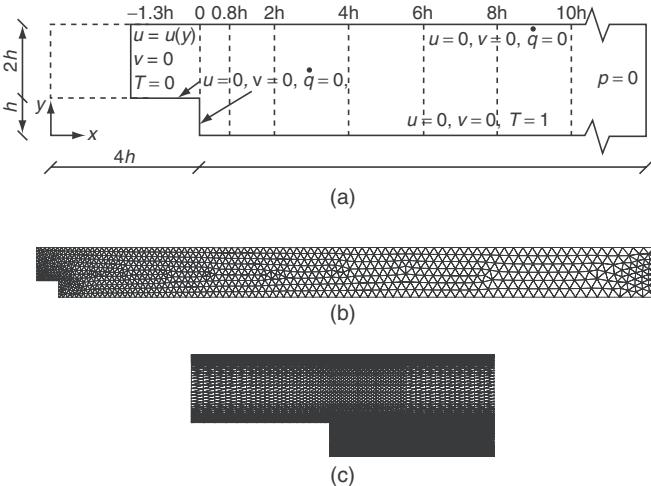


Figure 4.2. Backward-facing step: (a) computational domain and boundary conditions; (b) unstructured mesh (955 nodes and 1,704 elements); (c) detail of the structured mesh near the step (4,183 nodes, 8,092 elements).

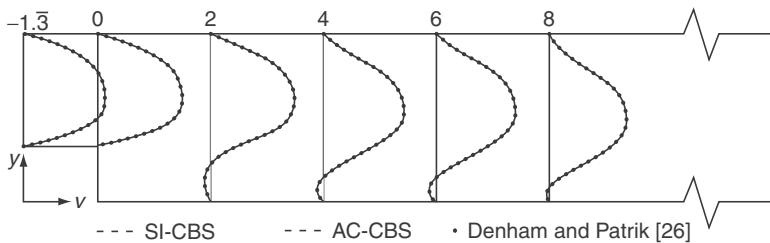


Figure 4.3. Backward-facing step. Comparison of the horizontal velocity profiles, $Re = 229$.

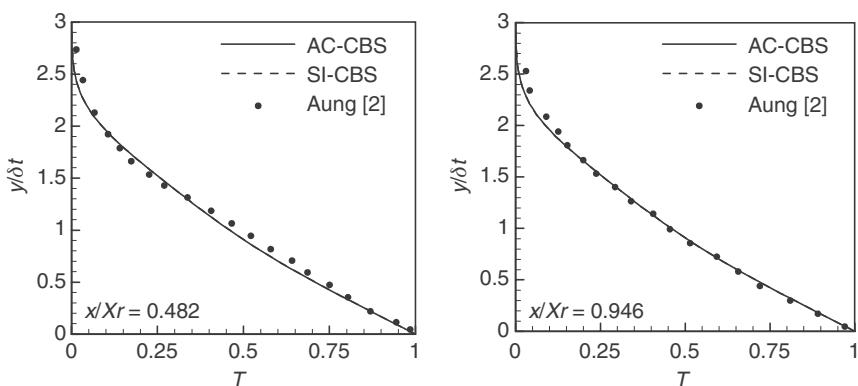


Figure 4.4. Backward-facing step. Temperature profiles at two different sections downstream of the step ($Re = 233$).

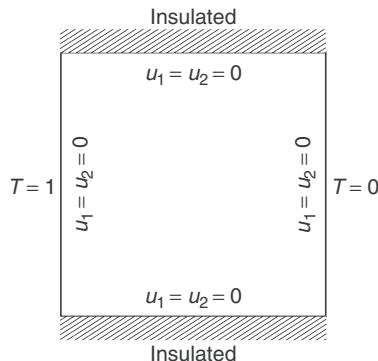


Figure 4.5. Natural convection in a square cavity. Computational domain and boundary conditions.

$Re = 233$. The profiles are practically the same for both the schemes and they show a very good agreement with the experimental data available from the literature for this problem (Aung [2]). It should be observed that the position along the vertical section is related to the thickness of the thermal boundary layer in the channel.

Another benchmark problem that is commonly employed for code validation is a well known natural convection problem in which the fluid-filled square cavity with the vertical walls subjected to different temperatures is used. The two horizontal walls are assumed to be adiabatic as shown in Figure 4.5. No-slip velocity conditions are applied on all walls. The flow is driven by the buoyancy forces acting in the vertical direction, caused by the temperature differences induced in the fluid. Both structured and unstructured meshes are used (Figure 4.6).

Figures 4.7–4.9 show the temperature contours and streamlines obtained with the AC-CBS algorithm at different Rayleigh numbers.

The transient flow over a circular cylinder is a popular test case for validating temporal discretization of numerical schemes. The inlet flow is uniform and the cylinder is placed at a centreline between two slip walls. The distance from the inlet to the centre of the cylinder is $4D$, where D is the diameter of the cylinder. Total length of the domain is $16D$. No slip condition is applied on the cylinder surface. The problem has been solved both on two and three dimensions. Figure 4.10 shows the mesh used for two-dimension case. As seen, the mesh close to the cylinder is very fine in order to capture the boundary layer and separation. Figure 4.11 shows the contours of the horizontal and vertical velocity components at a real non-dimensional time of 98.02. The time history of the vertical velocity component at an exit central point is plotted with respect to time and is shown in Figure 4.12.

In order to show that the method works also on three dimensions, a coarse mesh with 17,382 nodes and 69,948 tetrahedral elements was used to recompute the solution. This mesh was generated using the PSUE software available within the

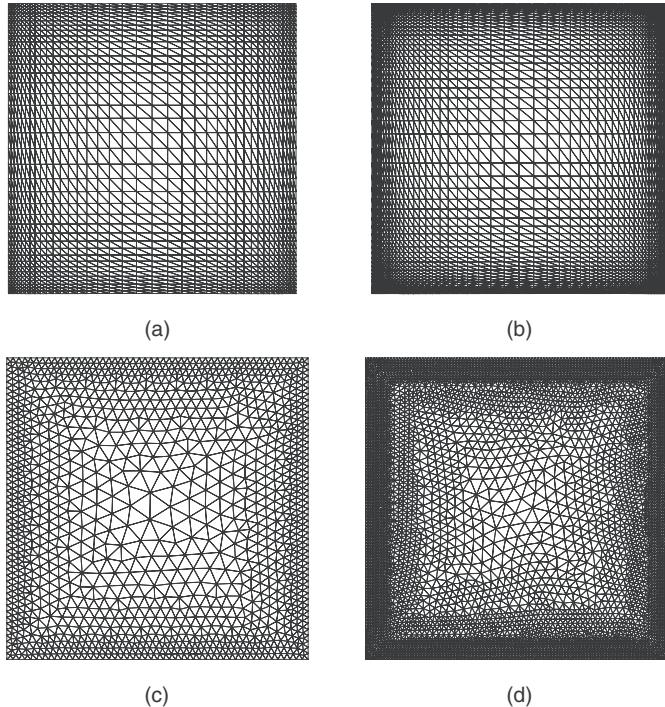


Figure 4.6. Natural convection in a square cavity. (a) Mesh 1: 1,251 nodes and 2,888 elements; (b) Mesh 2: 2,601 nodes and 5,000 elements; (c) Mesh 3: 1,338 nodes and 2,474 elements; (d) Mesh 4: 5,515 nodes and 10,596 elements.

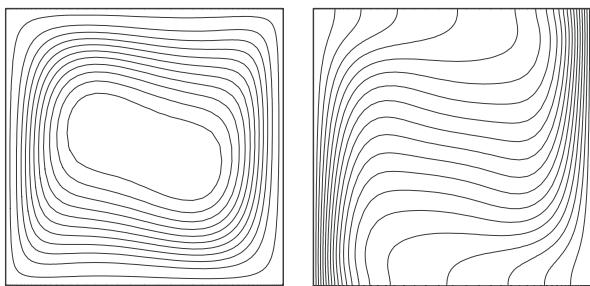


Figure 4.7. Natural convection in a square cavity. Streamlines and isotherms at $\text{Ra} = 10^5$.

School of Engineering, Swansea University (Morgan et al. [6]). The finite element mesh and the contours of velocity components are shown in Figures 4.13 and 4.14, respectively, at a real non-dimensional time of 83.33. As seen, the solution obtained is very similar to the one shown in Figure 4.11 for a two-dimensional case.

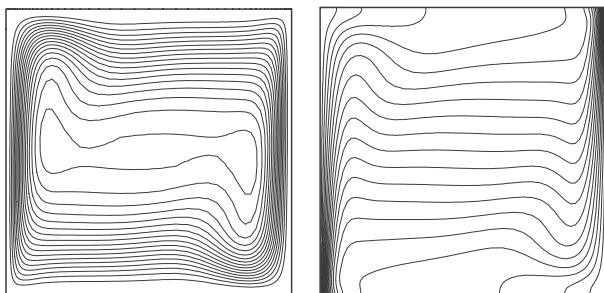


Figure 4.8. Natural convection in a square cavity. Streamlines and isotherms at $\text{Ra} = 10^6$.

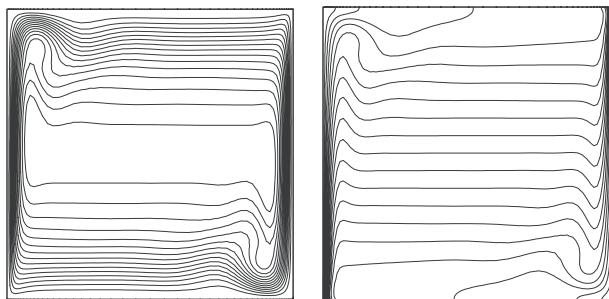


Figure 4.9. Natural convection in a square cavity. Streamlines and isotherms at $\text{Ra} = 10^7$.

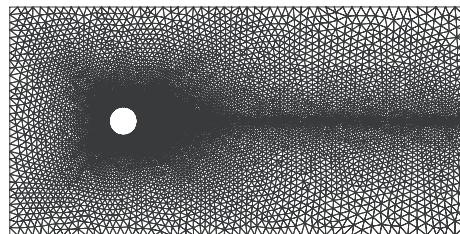


Figure 4.10. Flow past a circular cylinder. Finite element mesh, 7,129 nodes and 13,990 elements.

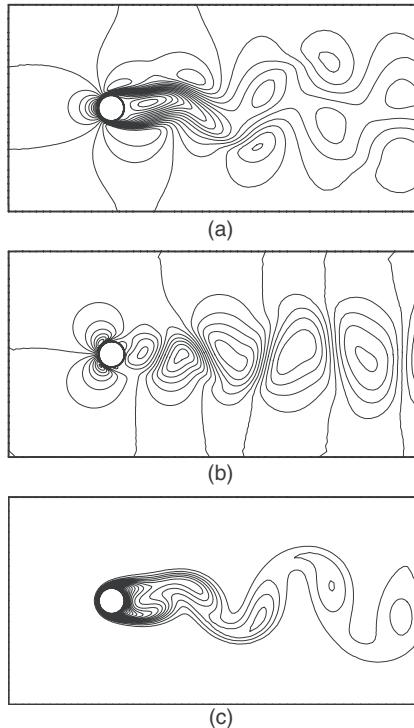


Figure 4.11. Flow past a circular cylinder, $Re = 100$. (a) Horizontal velocity contours; (b) vertical velocity contours; (c) temperature contours.

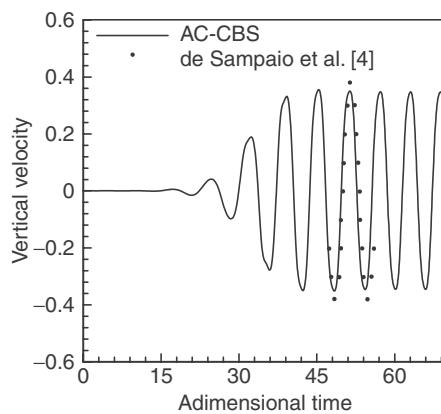


Figure 4.12. Flow past a circular cylinder. The vertical velocity distribution at an exit point of the domain with respect to real time. Comparison with de Sampaio et al. [4].

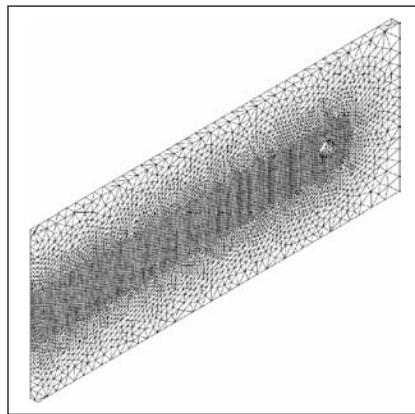


Figure 4.13. Three-dimensional flow past a circular cylinder. Finite element mesh: 17,382 nodes, 69,948 elements.

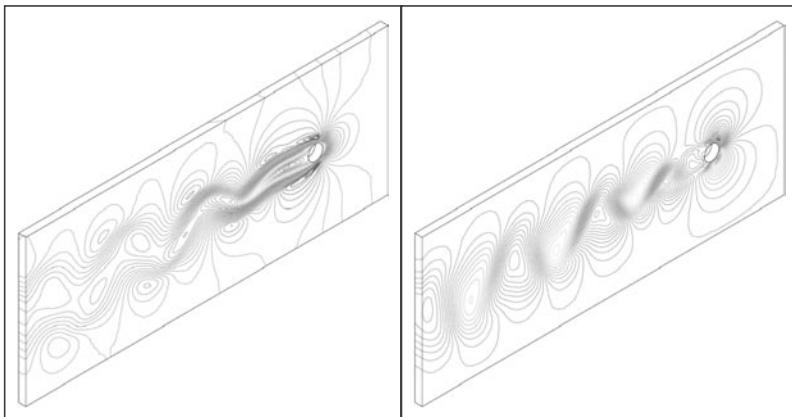


Figure 4.14. Three-dimensional flow past a circular cylinder, $Re = 100$; u_1 velocity contours (left); u_3 velocity contours (right).

4.1.2 Non-dimensional form of turbulent flow equations

For turbulent flow computations, Reynolds averaged Navier–Stokes equations of motion are written in conservation form as follows:

Mean continuity equation

$$\frac{1}{\beta^2} \frac{\partial \bar{p}}{\partial t} + \frac{\partial (\bar{\rho} \bar{u}_i)}{\partial x_i} = 0 \quad (10)$$

Mean momentum equation

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j} (\bar{u}_j \bar{u}_i) = - \frac{\partial \bar{p}}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} + \frac{\partial \tau_{ij}^R}{\partial x_j} \quad (11)$$

where β is an AC parameter, which will be well explained in Section 2.6, \bar{u}_i are the mean velocity components, p is the pressure, ρ is the density and τ_{ij} is the laminar shear stress tensor given as:

$$\tau_{ij} = \frac{1}{Re} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} - \frac{2}{3} \frac{\partial \bar{u}_k}{\partial x_k} \delta_{ij} \right) \quad (12)$$

The Reynolds stress tensor, τ_{ij}^R , is introduced by Boussinesq assumption as:

$$\tau_{ij}^R = \frac{\nu_T}{Re} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} - \frac{2}{3} \frac{\partial \bar{u}_k}{\partial x_k} \delta_{ij} \right) - \frac{2}{3} k \delta_{ij} \quad (13)$$

In the above equations, ν is the kinematic viscosity of the fluid, ν_T is the turbulent eddy viscosity and δ_{ij} is the Kroneker delta. The following non-dimensional scales are used to derive the above equations:

$$\begin{aligned} x_i^* &= \frac{x_i}{L}, \bar{u}_i^* = \frac{\bar{u}_i}{u_{ref}}, t^* = \frac{t u_{ref}}{L^2}, \bar{p}^* = \frac{\bar{p}}{\rho u_{ref}^2}, \\ k^* &= \frac{k}{u_{ref}^2}, \varepsilon^* = \frac{\varepsilon L}{u_{ref}^3}, v_T^* = \frac{\nu_T}{u_{ref}}, \hat{v}^* = \frac{\bar{v}}{u_{ref}} \end{aligned} \quad (14)$$

where L is a characteristic dimension and the subscript ref indicates a reference value. The turbulent flow solution is obtained by solving equations (10) and (11) with appropriate boundary conditions and a turbulence model, as the Spalart–Allmaras model (Spalart and Allmaras [7], Nithiarasu and Liu [8], Nithiarasu et al. [9]). The Spalart–Allmaras (SA) model was first introduced for aerospace applications and currently being adopted for incompressible flow calculations. The SA model is another one-equation model, which employs a single scalar equation and several constants to model turbulence. The scalar equation is:

$$\frac{\partial \hat{v}}{\partial t} + \frac{\partial(\bar{u}_j \hat{v})}{\partial x_j} = c_{b1} \hat{S} \hat{v} + \frac{1}{Re \sigma} \left[\frac{\partial}{\partial x_i} \left\{ (1 + \hat{v}) \frac{\partial \hat{v}}{\partial x_i} \right\} + c_{b2} \left(\frac{\partial \hat{v}}{\partial x_i} \right)^2 \right] - \frac{c_{w1} f_w}{Re} \left[\frac{\hat{v}}{y} \right]^2 \quad (15)$$

where

$$\hat{S} = S + \frac{1}{Re} (\hat{v}/k^2 y^2) f_{v2} \quad (16)$$

and

$$f_{v2} = 1 - X/(1 + X f_{v1}) \quad (17)$$

In equation (16) S is the magnitude of vorticity. The eddy viscosity is calculated as:

$$\nu_T = \hat{v} f_{v1} \quad (18)$$

where

$$f_{v1} = X^3 / (X^3 + c_{v1}^3) \quad (19)$$

and

$$X = \hat{v}/v \quad (20)$$

The parameter f_w is given as:

$$f_w = g \left[\frac{1 + c_{w3}^6}{g^6 + c_{w3}^6} \right]^{1/6} \quad (21)$$

where

$$g = r + c_{w2}(r^6 - r) \quad (22)$$

and

$$r = \frac{1}{\text{Re}} \frac{\hat{v}}{\hat{S} k^2 y^2} \quad (23)$$

The constants are $c_{b1} = 0.1355$, $\sigma = 2/3$, $c_{b2} = 0.622$, $k = 0.41$, $c_{w1} = c_{b1}/k^2 + (1 + c_{b2})/\sigma$, $c_{w2} = 0.3$, $c_{w3} = 2$ and $c_{v1} = 7.1$. From the above equations it is clear that the turbulent kinetic energy is not calculated here. Thus, the last term of equation (13) is dropped when SA model is employed.

Some examples

The following results are obtained by solving the above equations using an AC CBS algorithm and finite element discretization technique that will be described later in this chapter.

A standard test case commonly employed for turbulent incompressible flow models at moderate Reynolds number is the recirculating flow past a backward-facing step. The definition of the problem is shown in Figure 4.15. The characteristic dimension of the problem is the step height. All other dimensions are defined with respect to the characteristic dimension. The step is located at a distance of four times the step height from the step. The inlet channel height is two times the step height. The total length of the channel is 40 times the step height. The inlet velocity profile is obtained from the experimental data reported by Denham et al. [10]. No slip conditions apply on the solid walls. A fixed value of 0.05 for the turbulent scalar variable of the SA model at inlet was prescribed. The scalar variable was assumed to be zero on the walls. Both structured and unstructured meshes were employed in the calculation (Figures 4.16 and 4.17).

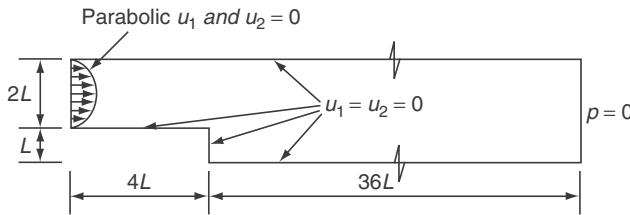


Figure 4.15. Turbulent flow past a two-dimensional backward-facing step. Problem definition.

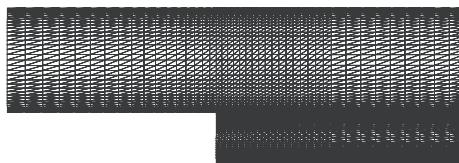


Figure 4.16. Detail of the structured mesh (8,092 elements and 4,183 nodes) near the step.

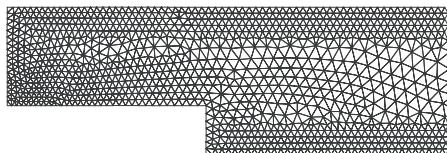


Figure 4.17. Detail of the unstructured mesh (8,662 elements and 4,656 nodes) near the step.

Figure 4.18 shows the comparison of velocity profiles against the experimental data of Denham et al. [10]. The SA model predicts accurately the recirculation region.

The other turbulence problem considered is a complex three-dimensional model of flow through an upper human airway (Nithiarasu and Liu [8], Nithiarasu et al. [9]). The geometry used is a reconstruction of an upper human airway employed in the spray dynamics studies (Gemci et al. [11]). It is apparent from the available studies on particle movement in the upper human airways that this problem is important (Li et al. [9, 10], Martonen et al. [11]). To understand the mechanism behind many upper human airway-related problems including “sleep apnoea” and vocal cord-related problems. Most of the reported studies on upper airway fluid dynamics use either structured or semi-structured meshes in the calculations. Here, the results obtained with an unstructured mesh are presented. The surface mesh of the grid used in the present study is shown in Figure 4.19. This mesh contains just under a million tetrahedral elements. The mesh is generated using the PSUE code (Morgan et al. [6]). The Reynolds number of the flow is defined based on the

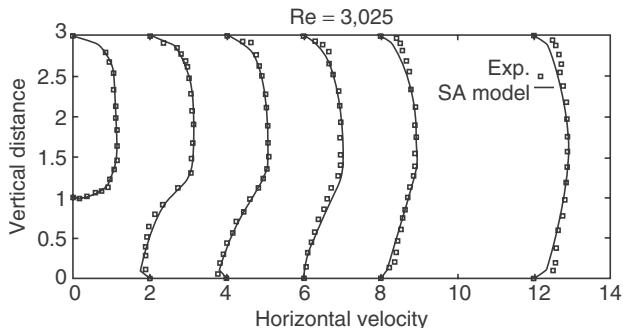


Figure 4.18. Incompressible turbulent flow past a backward-facing step. Velocity profiles at various downstream sections at $\text{Re} = 3,025$.

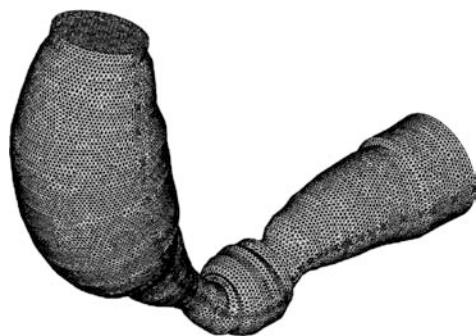


Figure 4.19. Incompressible turbulent flow through a model upper human airway. Surface mesh.

diameter of the narrow portion close to the epiglottis. The total non-dimensional length of the domain in the horizontal direction is 29.68, and in the vertical direction it is 23.05. The diameter at the inlet of the geometry (at the top) is 4.91.

A uniform velocity is assumed at the inlet of the geometry in the negative direction perpendicular to the inlet surface in the downward direction. No slip conditions are assumed on the solid walls. The turbulent scalar variable value at the inlet is fixed at 0.05 and assumed to be equal to zero on the walls.

Figure 4.20 shows the contours of velocity components and pressure within a section along the axis in the middle of the geometry. It is apparent that the majority of the activities are taking place near the narrow portion close to the epiglottis of the upper human airway. The flow is accelerated as it passes through the narrow portion of the airway. As seen, the pressure contours are clustered close to the narrow portion representing a very high gradient region. Figure 4.21 shows the velocity vectors within the section. The velocity vectors close to the narrow portion are also shown in this figure. As seen, the recirculation region is clearly predicted by the AC-CBS method.

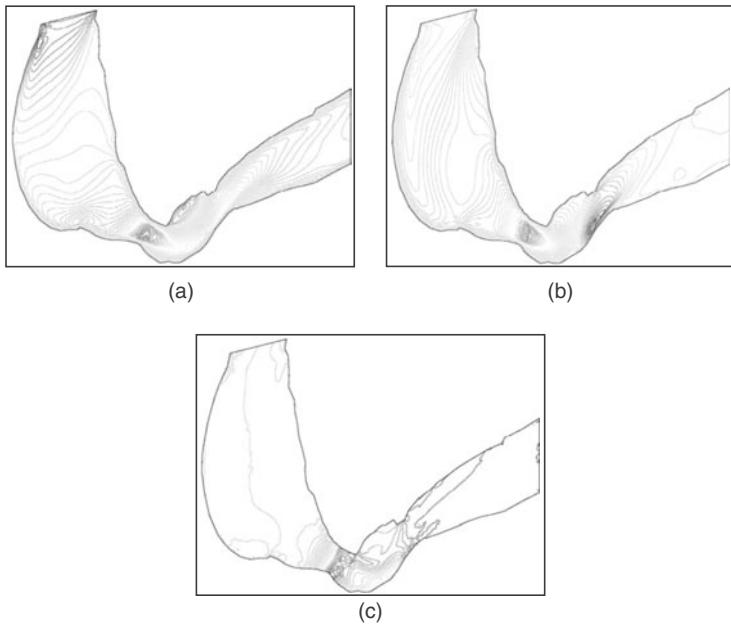


Figure 4.20. Incompressible turbulent flow through a model upper human airway.
 (a) u_1 contours, (b) u_3 contours, (c) pressure contours.

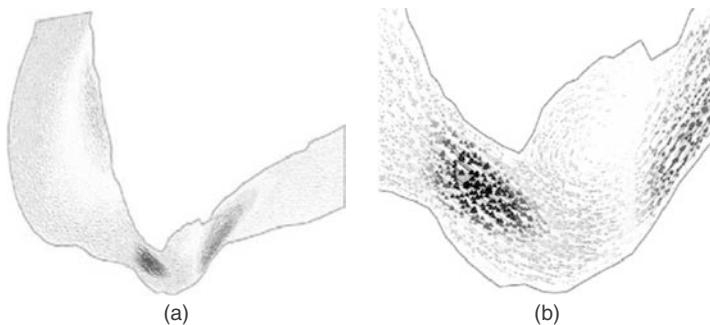


Figure 4.21. Incompressible turbulent flow through a model upper human airway.
 (a) Velocity vectors and (b) velocity vectors near the narrow portion.

4.1.3 Porous media flow: the generalized model equations

The general form of the equations for a porous medium can be derived by averaging the Navier–Stokes equations over a representative elementary volume (REV; Figure 4.22), using the well-known volume averaging procedure (Whitaker [15], Vafai and Tien [16], Hsu and Cheng [17]).

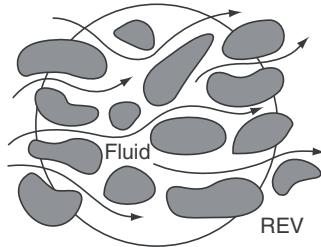


Figure 4.22. Representative elementary volume.

The non-dimensional form of the generalized model for the description of flow through a fluid-saturated porous medium can be written as:

Continuity equation

$$\frac{\partial u_i^*}{\partial x_i^*} = 0 \quad (24)$$

Momentum equation

$$\begin{aligned} \frac{1}{\varepsilon} \frac{\partial u_i^*}{\partial t^*} + \frac{1}{\varepsilon^2} u_j^* \frac{\partial u_i^*}{\partial x_j^*} &= - \frac{\partial p_f^*}{\partial x_i^*} + \frac{J}{\varepsilon \text{Re}} \frac{\partial}{\partial x_j^*} \left(\frac{\partial u_i^*}{\partial x_j^*} \right) \\ &\quad - \frac{1}{\text{Re Da}} u_i^* - \frac{F}{\sqrt{\text{Da}}} |\mathbf{u}^*| u_i^* + \frac{\text{Ra}}{\text{Pr Re}^2} T^* \gamma_i \end{aligned} \quad (25)$$

Energy equation

$$\sigma \frac{\partial T^*}{\partial t^*} + u_i^* \frac{\partial T^*}{\partial x_i^*} = \frac{\lambda^*}{\text{Pr Re}} \frac{\partial}{\partial x_i^*} \left(\frac{\partial T^*}{\partial x_i^*} \right) \quad (26)$$

where Da is the Darcy number and F is the Forchheimer coefficient. The buoyancy effects are incorporated by invoking the Boussinesq approximation:

$$g(\rho_f - \rho_{\text{ref}}) = \rho_{\text{ref}} g \beta (T_{\text{ref}} - T) \quad (27)$$

where the subscript f refers to the fluid that saturates the porous medium. The scales and the parameters used to derive the above non-dimensional equations for mixed convection through a saturated porous medium are the same as those shown at the beginning of the chapter. The new parameters introduced here are:

$$J = \frac{\mu_{\text{eff}}}{\mu_f}, \sigma = \frac{\varepsilon(\rho c_p)_f + (1 - \varepsilon)(\rho c_p)_s}{(\rho c_p)_f}, \lambda^* = \frac{\lambda_{\text{eff}}}{\lambda_f}, \text{Da} = \frac{\kappa}{L^2} \quad (28)$$

where J is the ratio between the effective and the fluid viscosity, σ is the heat capacity ratio, λ^* is the ratio between the effective and the fluid thermal conductivity, Da is

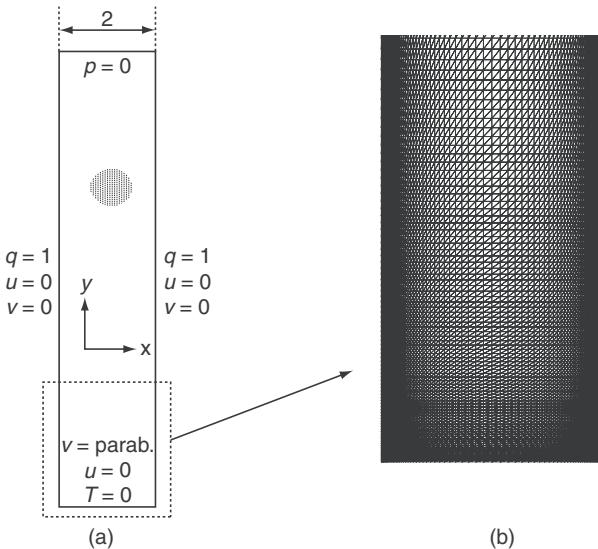


Figure 4.23. Mixed convection in porous vertical channel. (a) Computational domain and boundary conditions; (b) detail of the structured computational grid near the entrance (8,601 nodes and 16,800 elements).

the Darcy number, κ and ε are the permeability and porosity of the medium, respectively, the subscripts p and eff refer to the porous medium and to the effective values, respectively. As mentioned, equations (24)–(26) are derived for mixed convection problems, therefore this set of PDEs describes both natural and forced convection. When forced convection dominates the problem ($\text{Ra}/\text{PrRe}^2 < 1$), the buoyancy term on the right hand side of the momentum conservation can be neglected.

The generalized model equations introduced above reduce to the Navier–Stokes equations when the solid matrix in the porous medium disappears, that is when $\varepsilon \rightarrow 1$ and $\text{Da} \rightarrow \infty$, while, as the porosity $\varepsilon \rightarrow 0$ and $\text{Da} \rightarrow 0$, the equations represent a solid. Therefore, the procedure can be used to describe interface problems in which a saturated porous medium interacts with a single-phase fluid. In these cases, it is possible to use a single domain approach, by changing the property of the medium accordingly.

Some examples

The following results are obtained by solving the above equations for mixed convection flows by using an AC version of the CBS algorithm and finite element discretization technique that will be explained later.

The example showed here is the fully developed mixed convection in a region filled with a fluid-saturated porous medium, confined between two vertical walls. The computational domain and the boundary conditions are shown in Figure 4.23a.

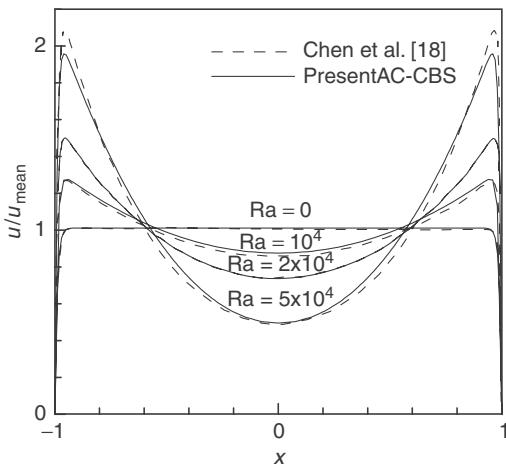


Figure 4.24. Mixed convection in porous vertical channel. Non-dimensional vertical velocity at different Ra for aiding flow, at $\text{Da} = 10^{-4}$.

The flow enters the domain from the bottom at a non-dimensional temperature of $T = 0$ with a fully developed parabolic velocity profile. The uniform heat flux condition is symmetrically imposed on both walls. Figure 4.23b shows the details of the computational grid near the entrance. The mesh employed has 8,601 nodes and 16,800 elements and it is refined near the walls and at the inlet region.

Figure 4.24 shows the dimensionless vertical velocity profile at a Darcy number equal to 10^{-4} and four different Rayleigh numbers. The present results are compared with the numerical results of Chen et al. [18]. For pure forced convection ($\text{Ra} = 0$), the velocity profile is flat in the central part of the domain and sharply goes to zero at the walls of the channel. In the case of aiding flow, if the Rayleigh number increases, the buoyancy forces result in an increase in the fluid velocity near the walls. As the flow rate entering the channel is the same for any Rayleigh numbers considered, the increase in fluid velocity in the near-wall regions will result in a decreased velocity at the centre region of the channel. When Rayleigh number is increased to 5×10^4 , the maximum velocity in the near-wall region is significantly higher than the fluid velocity at the centre of the channel.

4.2 The Finite Element Method

We provide here an introduction to weighted residual approximation and finite element method for self-adjoint equations and for convection-diffusion equations.

4.2.1 Strong and weak forms

The Laplace equation is a very convenient example for the start of numerical approximations. We shall generalize slightly and discuss in some detail the quasi-harmonic

(Poisson) equation:

$$-\frac{\partial}{\partial x_i} \left(k \frac{\partial \phi}{\partial x_i} \right) + Q = 0 \quad (29)$$

where k and Q are specified functions. These equations together with appropriate boundary conditions define the problem uniquely. The boundary conditions can be of Dirichlet type:

$$\phi = \bar{\phi}, \quad \text{on } \Gamma_\phi \quad (30)$$

or that of Neumann type:

$$q_n = -k \frac{\partial \phi}{\partial n} = \bar{q}_n, \quad \text{on } \Gamma_q \quad (31)$$

where a bar denotes a specified quantity. Equations (29)–(31) are known as the *strong form* of the problem.

We note that direct use of equation (29) requires computation of second derivatives to solve a problem using approximate techniques. This requirement may be weakened by considering an integral expression for equation (29) written as:

$$\int_{\Omega} v \left[-\frac{\partial}{\partial x_i} \left(k \frac{\partial \phi}{\partial x_i} \right) + Q \right] d\Omega = 0 \quad (32)$$

in which v is an arbitrary function.

If we assume equation (29) is not zero at some point x_i in Ω then we can also let v be a positive parameter times the same value resulting in a positive result for the integral equation (32). Since this violates the equality we conclude that equation (29) must be zero for every x_i in Ω , hence proving its equality with equation (32).

We may integrate by parts the second derivative terms in equation (32) to obtain:

$$\int_{\Omega} \frac{\partial v}{\partial x_i} \left(k \frac{\partial \phi}{\partial x_i} \right) d\Omega + \int_{\Omega} v Q d\Omega - \int_{\Gamma} v n_i \left(k \frac{\partial \phi}{\partial x_i} \right) d\Gamma = 0 \quad (33)$$

We now split the boundary into two parts, Γ_ϕ and Γ_q , with $\Gamma = \Gamma_\phi + \Gamma_q$, and use equation (31) in equation (33) to give:

$$\int_{\Omega} \frac{\partial v}{\partial x_i} \left(k \frac{\partial \phi}{\partial x_i} \right) d\Omega + \int_{\Omega} v Q d\Omega - \int_{\Gamma_q} v \bar{q}_n d\Gamma = 0 \quad (34)$$

which is valid only if v vanishes on Γ_ϕ . Hence we must impose equation (30) for equivalence.

Equation (34) is known as the *weak form* of the problem since only first derivatives are necessary in constructing a solution. Such forms are the basis for obtaining the finite element solutions.

4.2.2 Weighted residual approximation

In a weighted residual scheme, an approximation to the independent variable ϕ is written as a sum of known *trial functions* (basis functions) $N_a(x_i)$ and unknown parameters $\tilde{\phi}^a$. Thus we can always write:

$$\begin{aligned}\phi \approx \hat{\phi} &= N_1(x_i)\tilde{\phi}^1 + N_2(x_i)\tilde{\phi}^2 + \dots \\ &= \sum_{a=1}^n N_a(x_i)\tilde{\phi}^a = \mathbf{N}(x_i)\tilde{\boldsymbol{\varphi}}\end{aligned}\quad (35)$$

where

$$\mathbf{N} = [N_1, N_2, \dots, N_n] \quad (36)$$

and

$$\tilde{\boldsymbol{\varphi}} = [\tilde{\phi}^1, \tilde{\phi}^2, \dots, \tilde{\phi}^n]^T \quad (37)$$

In a similar way we can express the arbitrary variable v as:

$$\begin{aligned}v \approx \hat{v} &= W_1(x_i)\tilde{v}^1 + W_2(x_i)\tilde{v}^2 + \dots \\ &= \sum_{a=1}^n W_a(x_i)\tilde{v}^a = \mathbf{W}(x_i)\tilde{\mathbf{v}}\end{aligned}\quad (38)$$

in which W_a are *test functions* and \tilde{v}^a arbitrary parameters. Using this form of approximation will convert equation (34) to a set of algebraic equations.

In the finite element method and indeed in all other numerical procedures for which a computer-based solution can be used, the test and trial functions will generally be defined in a local manner. It is convenient to consider each of the tests and basis functions to be defined in partitions Ω_e of the total domain Ω . This division is denoted by:

$$\Omega \approx \Omega_h = \bigcup \Omega_e \quad (39)$$

and in a finite element method Ω_e are known as *elements*. The very simplest uses lines in one dimension, triangles in two dimensions and tetrahedra in three dimensions in which the basis functions are usually linear polynomials in each element and the unknown parameters are *nodal values* of ϕ . In Figure 4.25 we show a typical set of such linear functions defined in two dimensions.

In a weighted residual procedure, we first insert the approximate function $\hat{\phi}$ into the governing differential equation creating a residual, $R(x_i)$, which of course should be zero at the exact solution. In the present case for the quasi-harmonic equation we obtain:

$$R = -\frac{\partial}{\partial x_i} \left(k \sum_a \frac{\partial N_a}{\partial x_i} \tilde{\phi}^a \right) + Q \quad (40)$$

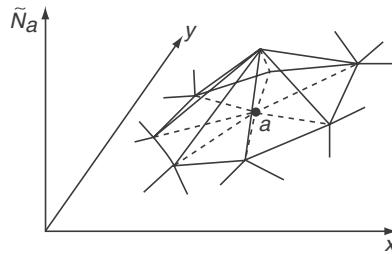


Figure 4.25. Basis function in linear polynomials for a patch of triangular elements.

and we now seek the best values of the parameter set $\tilde{\phi}^a$, which ensures that:

$$\int_{\Omega} W_b R d\Omega = 0, \quad b = 1, 2, \dots, n \quad (41)$$

Note that this is the term multiplying the arbitrary parameter \tilde{v}^b . As noted previously, integration by parts is used to avoid higher-order derivatives (i.e. those greater than or equal to two) and therefore reduce the constraints on choosing the basis functions to permit integration over individual elements using equation (39). In the present case, for instance, the weighted residual after integration by parts and introducing the natural boundary condition becomes:

$$\int_{\Omega} \frac{\partial W_b}{\partial x_i} \left(k \sum_a \frac{\partial N_a}{\partial x_i} \tilde{\phi} \right) d\Omega + \int_{\Omega} W_b Q d\Omega + \int_{\Gamma_q} W_b \bar{q}_n d\Gamma = 0 \quad (42)$$

4.2.3 The Galerkin, finite element, method

In the Galerkin method we simply take $W_b = N_b$, which gives the assembled system of equations:

$$\sum_{a=1}^n K_{ba} \tilde{\phi}^a + f_b = 0, \quad b = 1, 2, \dots, n - r \quad (43)$$

where r is the number of nodes appearing in the approximation to the Dirichlet boundary condition (i.e., equation (30)) and K_{ba} is assembled from element contributions K_{ba}^e with:

$$K_{ba}^e = \int_{\Omega_e} \frac{\partial N_b}{\partial x_i} k \frac{\partial N_a}{\partial x_i} d\Omega \quad (44)$$

Similarly, f_b is computed from the element as:

$$f_b^e = \int_{\Omega_e} N_b Q d\Omega + \int_{\Gamma_{eq}} N_b \bar{q}_n d\Gamma \quad (45)$$

To impose the Dirichlet boundary condition we replace $\tilde{\phi}^a$ by $\bar{\phi}_a$ for the r boundary nodes.

It is evident in this example that the Galerkin method results in a symmetric set of algebraic equations (e.g. $K_{ba} = K_{ab}$). However, this only happens if the differential equations are *self-adjoint*. Indeed the existence of symmetry provides a test for self-adjointness and also for existence of a variational principle whose stationarity is sought.

It is necessary to remark here that if we were considering a pure convection equation:

$$u_i \frac{\partial \phi}{\partial x_i} + Q = 0 \quad (46)$$

symmetry would not exist and such equations can often become unstable if the Galerkin method is used.

4.2.4 Characteristic Galerkin scheme for convection–diffusion equation

Unlike a simple conduction equation (as the Laplace equation), a numerical solution for the convection equation has to deal with the convection part of the governing equation in addition to diffusion. For most conduction equations, the finite element solution is straightforward. However, if a Galerkin type approximation was used in the solution of convection equations, the results will be marked with spurious oscillations in space if certain parameters exceed a critical value (element Peclet number). This problem is not unique to finite elements as all other spatial discretization techniques have the same difficulties. A very well-known method used in finite elements approximation to reduce these oscillations is the *Characteristic Galerkin (CG) scheme* (Lewis et al. [19], Zienkiewicz et al. [20]). Here, we follow the Characteristic Galerkin (CG) approach to deal with spatial oscillations due to the discretization of the convection transport terms.

In order to demonstrate the CG method, let us consider the simple convection–diffusion equation in one dimension, namely:

$$\frac{\partial \phi}{\partial t} + u_1 \frac{\partial \phi}{\partial x_1} - \frac{\partial}{\partial x_1} \left(k \frac{\partial \phi}{\partial x_1} \right) = 0 \quad (47)$$

Let us consider a characteristic of the flow as shown in Figure 4.26 in the time–space domain. The incremental time period covered by the flow is Δt from the n th time level to the $n + 1$ th time level and the incremental distance covered during this

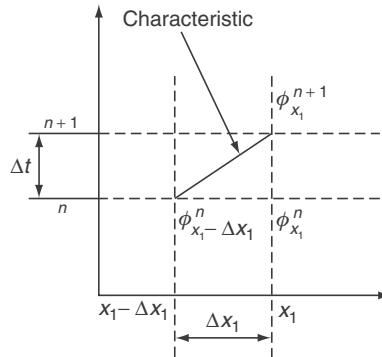


Figure 4.26. Characteristic in a space–time domain.

time period is Δx_1 , that is, from $(x_1 - \Delta x_1)$ to x_1 . If a moving coordinate is assumed along the path of the characteristic wave with a speed of u_1 , the convection terms of equation (47) disappear (as in a Lagrangian fluid dynamics approach). Although this approach eliminates the convection term responsible for spatial oscillation when discretized in space, the complication of a moving coordinate system x'_1 is introduced, that is, equation (47) becomes:

$$\frac{\partial \phi}{\partial t}(x'_1, t) - \frac{\partial}{\partial x'_1} \left(k \frac{\partial \phi}{\partial x'_1} \right) = 0 \quad (48)$$

The semi-discrete form of the above equation can be written as:

$$\frac{\phi^{n+1}_{|x_1} - \phi^n_{|x_1 - \Delta x_1}}{\Delta t} - \frac{\partial}{\partial x'_1} \left(k \frac{\partial \phi}{\partial x'_1} \right)^n_{|x_1 - \Delta x_1} = 0 \quad (49)$$

Note that the diffusion term is treated explicitly. It is possible to solve the above equation by adapting a moving coordinate strategy. However, a simple spatial Taylor series expansion in space avoids such a moving coordinate approach. With reference to Figure 4.26, we can write using a Taylor series expansion:

$$\phi^n_{|x_1 - \Delta x_1} = \phi^n_{|x_1} - \frac{\partial \phi^n}{\partial x_1} \frac{\Delta x_1}{1!} + \frac{\partial^2 \phi^n}{\partial x_1^2} \frac{\Delta x_1^2}{2!} - \dots \quad (50)$$

Similarly, the diffusion term is expanded as:

$$\frac{\partial}{\partial x'_1} \left(k \frac{\partial \phi}{\partial x'_1} \right)^n_{|x_1 - \Delta x_1} = \frac{\partial}{\partial x_1} \left(k \frac{\partial \phi}{\partial x_1} \right)^n_{|x_1} - \frac{\partial}{\partial x_1} \left[\frac{\partial}{\partial x_1} \left(k \frac{\partial \phi}{\partial x_1} \right)^n \right] \Delta x \quad (51)$$

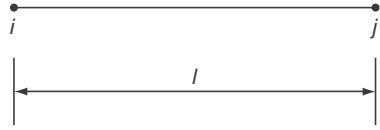


Figure 4.27. One-dimensional linear element.

On substituting equations (50) and (51) into equation (49), we obtain (higher-order terms being neglected) the following expression:

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = -\frac{\Delta x}{\Delta t} \frac{\partial \phi}{\partial x_1}^n + \frac{\Delta x^2}{2\Delta t} \frac{\partial^2 \phi}{\partial x_1^2}^n + \frac{\partial}{\partial x_1} \left(k \frac{\partial \phi}{\partial x_1} \right)^n \quad (52)$$

In this case, all the terms are evaluated at position x_1 , and not at two positions as in equation (49). If the flow velocity is u_1 , we can write $\Delta x = u_1 \Delta t$. Substituting into equation (52), we obtain the semi-discrete form as:

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = -u_1 \frac{\partial \phi}{\partial x_1}^n + u_1^2 \frac{\Delta t}{2} \frac{\partial^2 \phi}{\partial x_1^2}^n + \frac{\partial}{\partial x_1} \left(k \frac{\partial \phi}{\partial x_1} \right)^n \quad (53)$$

By carrying out a Taylor series expansion (Figure 4.26), the convection term reappears in the equation along with an additional second-order term. This second-order term acts as a smoothing operator that reduces the oscillations arising from the spatial discretization of the convection terms. The equation is now ready for spatial approximation.

The following linear spatial approximation of the scalar variable ϕ in space is used to approximate equation (53):

$$\phi = N_i \phi_i + N_j \phi_j = [\mathbf{N}] \{\boldsymbol{\phi}\} \quad (54)$$

where $[\mathbf{N}]$ are the shape functions and subscripts i and j indicate the nodes of a linear element as shown in Figure 4.27.

On employing the Galerkin weighting to equation (53), we obtain:

$$\begin{aligned} & \int_{\Omega} [\mathbf{N}]^T \frac{\phi^{n+1} - \phi^n}{\Delta t} d\Omega + \int_{\Omega} [\mathbf{N}]^T u_1 \frac{\partial \phi}{\partial x_1}^n d\Omega \\ & - \frac{\Delta t}{2} \int_{\Omega} [\mathbf{N}]^T u_1^2 \frac{\partial^2 \phi}{\partial x_1^2}^n d\Omega - \int_{\Omega} [\mathbf{N}]^T \frac{\partial}{\partial x_1} \left(k \frac{\partial \phi}{\partial x_1} \right)^n = 0 \end{aligned} \quad (55)$$

The above equation is equal to zero only if all the element contributions are assembled. For a domain with only one element, we can substitute:

$$[\mathbf{N}]^T = \begin{bmatrix} N_i \\ N_j \end{bmatrix} \quad (56)$$

On substituting a linear spatial approximation for the variable ϕ , over elements as typified in Figure 4.27, into equation (55), we get:

$$\begin{aligned} \int_{\Omega} [\mathbf{N}]^T [\mathbf{N}] \frac{\{\phi^{n+1} - \phi^n\}}{\Delta t} d\Omega &= -u_1 \int_{\Omega} [\mathbf{N}]^T \frac{\partial}{\partial x_1} ([\mathbf{N}]\{\phi\})^n d\Omega \\ &+ \frac{\Delta t}{2} u_1^2 \int_{\Omega} [\mathbf{N}]^T \frac{\partial^2}{\partial x_1^2} ([\mathbf{N}]\{\phi\})^n d\Omega + \int_{\Omega} [\mathbf{N}]^T k \frac{\partial^2}{\partial x_1^2} ([\mathbf{N}]\{\phi\})^n d\Omega \end{aligned} \quad (57)$$

Before utilizing the linear integration formulae, we apply Green's lemma to the second-order terms of equation (57), we obtain:

$$\begin{aligned} \int_{\Omega} [\mathbf{N}]^T [\mathbf{N}] \frac{\{\phi^{n+1} - \phi^n\}}{\Delta t} d\Omega &= -u_1 \int_{\Omega} [\mathbf{N}]^T \frac{\partial [\mathbf{N}]}{\partial x_1} \{\phi\}^n d\Omega \\ &- \frac{\Delta t}{2} u_1^2 \int_{\Omega} \frac{\partial [\mathbf{N}]^T}{\partial x_1} \frac{\partial [\mathbf{N}]}{\partial x_1} \{\phi\}^n d\Omega + \frac{\Delta t}{2} u_1^2 \int_{\Gamma} [\mathbf{N}]^T \frac{\partial [\mathbf{N}]}{\partial x_1} \{\phi\}^n n_1 d\Gamma \\ &- \int_{\Omega} \frac{\partial [\mathbf{N}]^T}{\partial x_1} k \frac{\partial [\mathbf{N}]}{\partial x_1} \{\phi\}^n d\Omega + \int_{\Gamma} [\mathbf{N}]^T k \frac{\partial [\mathbf{N}]}{\partial x_1} \{\phi\}^n n_1 d\Gamma \end{aligned} \quad (58)$$

where n_1 and n_2 are the direction cosines of the outward normal \mathbf{n} , Ω is the domain and Γ is the domain boundary. The first-order convection term can be integrated either directly or via Green's lemma. Here, the convection term is integrated directly without applying Green's lemma. However, integration of the first derivatives by parts is useful for problems in which the traction is prescribed. Using the integration formulae (Lewis et al. [19]), it is possible to derive the element matrices for all the terms in equation (58). The term on the left-hand side for a single element is:

$$\int_{\Omega} [\mathbf{N}]^T [\mathbf{N}] \frac{\{\phi^{n+1} - \phi^n\}}{\Delta t} d\Omega = \frac{l}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{Bmatrix} \frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} \\ \frac{\phi_j^{n+1} - \phi_j^n}{\Delta t} \end{Bmatrix} = [\mathbf{M}_e] \frac{\Delta \{\phi\}}{\Delta t} \quad (59)$$

where $[\mathbf{M}_e]$ is the mass matrix for a single element. The above mass matrix for a single element will have to be utilized in an assembly procedure for a fluid domain containing many elements.

In a similar fashion, all other terms can be integrated; for example, the convection term is given by:

$$u_1 \int_{\Omega} [\mathbf{N}]^T \frac{\partial [\mathbf{N}]}{\partial x_1} \{\phi\}^n d\Omega = \frac{u_1}{2} \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} \phi_i \\ \phi_j \end{Bmatrix}^n = [\mathbf{C}_e] \{\phi\}^n \quad (60)$$

where $[\mathbf{C}_e]$ is the elemental convection matrix. The values of the derivatives of the shape functions are substituted in order to derive the above matrix.

The diffusion term within the domain is integrated as:

$$\int_{\Omega} \frac{\partial[\mathbf{N}]^T}{\partial x_1} k \frac{\partial[\mathbf{N}]}{\partial x_1} \{\phi\}^n d\Omega = \frac{k}{l} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} \phi_i \\ \phi_j \end{Bmatrix}^n = [\mathbf{K}_e] \{\varphi\}^n \quad (61)$$

where $[\mathbf{K}_e]$ is the elemental diffusion matrix. The characteristic Galerkin term within the domain is integrated as:

$$\frac{\Delta t}{2} u_1^2 \int_{\Omega} \frac{\partial[\mathbf{N}]^T}{\partial x_1} \frac{\partial[\mathbf{N}]}{\partial x_1} \{\phi\}^n d\Omega = u_1^2 \frac{\Delta t}{2} \frac{1}{l} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} \phi_i \\ \phi_j \end{Bmatrix}^n = [\mathbf{K}_{se}] \{\varphi\}^n \quad (62)$$

where $[\mathbf{K}_{se}]$ is the elemental stabilization matrix.

The boundary term from the diffusion operator is integrated by assuming that i is a boundary node, as follows:

$$\int_{\Gamma} [\mathbf{N}]^T k \frac{\partial[\mathbf{N}]}{\partial x_1} \{\phi\}^n n_1 d\Gamma = k \begin{Bmatrix} -\frac{\phi_i}{l} + \frac{\phi_j}{l} \\ 0 \end{Bmatrix}^n n_1 = [\mathbf{f}_e] \quad (63)$$

where $\{\mathbf{f}_e\}$ is the forcing vector due to the diffusion term.

The boundary integral from the characteristic Galerkin term is integrated, again by assuming that i is a boundary node, as:

$$\frac{\Delta t}{2} u_1^2 \int_{\Gamma} [\mathbf{N}]^T \frac{\partial[\mathbf{N}]}{\partial x_1} \{\phi\}^n n_1 d\Gamma = \frac{\Delta t}{2} u_1^2 \begin{Bmatrix} -\frac{\phi_i}{l} + \frac{\phi_j}{l} \\ 0 \end{Bmatrix}^n n_1 = [\mathbf{f}_{se}] \quad (64)$$

where $\{\mathbf{f}_{se}\}$ is the forcing vector due to the stabilization term.

For a one-dimensional domain with more than one element, all the matrices and vectors need to be assembled in order to obtain the global matrices. Once assembled, the discretized one-dimensional equation becomes:

$$[\mathbf{M}] \frac{\Delta\{\varphi\}}{\Delta t} = -[\mathbf{C}] \{\phi\}^n - [\mathbf{K}] \{\phi\}^n - [\mathbf{K}_s] \{\phi\}^n + \{\mathbf{f}\}^n + \{\mathbf{f}_s\}^n \quad (65)$$

Let us now consider a simple one-dimensional convection problem, as given in Figure 4.28, to demonstrate the effect of a discretization with and without the CG scheme.

The scalar variable value at the inlet is $\varphi = 0$, and at the exit its value is 1. This scalar variable is transported in the direction of the velocity as shown in Figure 4.28. Note that the convection velocity u_1 is constant. The element Peclet number for this problem is defined as:

$$Pe = \frac{u_1 h}{2k} \quad (66)$$

where h is the element size in the flow direction, which, in one dimension is the local element length. Figure 4.29 shows the comparison between a solution with

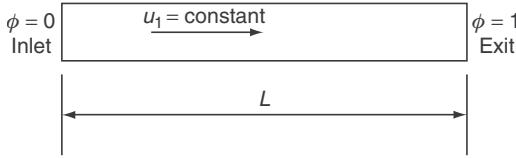
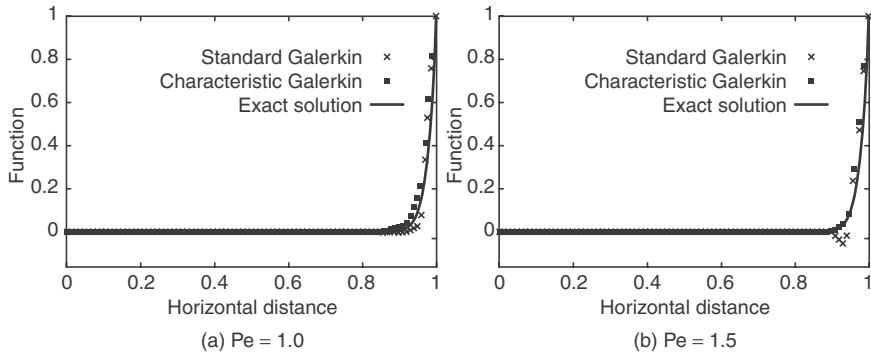


Figure 4.28. One-dimensional convection–diffusion problems.

Figure 4.29. Spatial variation of a function, ϕ , in one-dimensional space for different element Peclet numbers.

the CG discretization scheme and one without it. Only two Peclet numbers are shown in these diagrams to demonstrate the spatial oscillations without the CG discretization. As seen, both discretizations give no spatial oscillations at a Pe value of unity. However, at a Pe value of 1.5, the CG discretization is accurate and stable, while the discretization without the CG term becomes oscillatory. The exact solution to this problem is given as follows (Brooks and Hughes [21]):

$$\phi = \frac{1 - e^{\frac{u_1 x_1}{k}}}{1 - e^{\frac{u_1 L}{k}}} \quad (67)$$

In this equation, L is the total length of the domain and x_1 is the local length of the domain.

The extension of the characteristic Galerkin scheme to a multi-dimensional scalar convection–diffusion equation is straightforward and follows the previous procedure as discussed for a one-dimensional case.

4.2.5 Stability conditions

The stability conditions for a given time discretization may be derived using a Von Neumann or Fourier analysis for either the convection equation or the convection–diffusion equations. However, for more complicated equations such

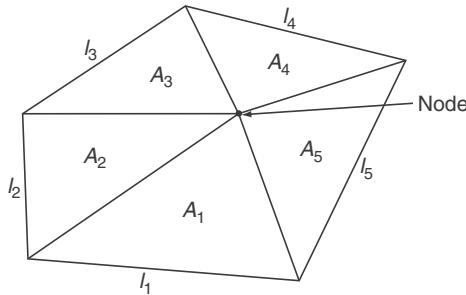


Figure 4.30. Two-dimensional linear triangular element.

as the Navier–Stokes equations, the derivation of the stability limit is not straightforward. A detailed discussion on stability criteria is not given here and readers are asked to refer to the relevant text books and papers for details (Hirsch [22], Zienkiewicz and Codina [23]). A stability analysis will give some idea about the time-step restrictions of any numerical scheme. In general, for fluid dynamics problems, the time-step magnitude is controlled by two wave speeds. The first one is due to the convection velocity and the second due to the real diffusion introduced by the equations. In the case of a convection–diffusion equation, the convection velocity is $\sqrt{u_i u_i}$, which is, in two dimensional problems, $\sqrt{u_1^2 + u_2^2} = |\mathbf{u}|$. The diffusion velocity is $2k/h$ where h is the local element size. The time-step restrictions are calculated as the ratio of the local element size and the local wave speed. It is therefore correct to write that the time step is calculated as:

$$\Delta t = \min(\Delta t_c, \Delta t_d) \quad (68)$$

where Δt_c and Δt_d are the convection and diffusion time-step limits, respectively, which are:

$$\Delta t_c = \frac{h}{|\mathbf{u}|}, \quad \Delta t_d = \frac{h^2}{2k} \quad (69)$$

Often, it may be necessary to multiply the time-step Δt by a safety factor due to different methods of element size calculations. A simple procedure to calculate the element size in two dimensions is:

$$h = \min\left(\frac{2\text{Area}_i}{l_i}\right), \quad i = 1, \text{ number of elements connected to the node} \quad (70)$$

where Area_i are the area of the elements connected to the node and l_i are the length of the opposite sides as shown in Figure 4.30. For the node shown in this figure, the local element size is calculated as:

$$h = \min(A_1/l_1, A_2/l_2, A_3/l_3, A_4/l_4, A_5/l_5) \quad (71)$$

In three dimensions, the term 2Area_i is replaced by 3Volume_i and l_i is replaced by the area opposite the node in question.

4.2.6 Characteristic-based split scheme

It is essential to understand the characteristic Galerkin procedure, discussed in Section 2.4 for the convection–diffusion equation, in order to apply the concept to solve the real convection equations. Unlike the convection–diffusion equation, the momentum equation, which is part of a set of heat convection equations, is a vector equation. A direct extension of the CG scheme to solve the momentum equation is difficult. In order to apply the CG approach to the momentum equations, we have to introduce two steps. In the first step, the pressure term from the momentum equation will be dropped and an intermediate velocity field will be calculated. In the second step, the intermediate velocities will be corrected. This two-step procedure for the treatment of the momentum equations has two advantages. The first advantage is that without the pressure terms, each component of the momentum equation is similar to that of a convection–diffusion equation and the CG procedure can be readily applied. The second advantage is that removing the pressure term from the momentum equations enhances the pressure stability and allows the use of arbitrary interpolation functions for both velocity and pressure. In other words, the well-known Babuska–Brezzi condition is satisfied (Babuska [24], Brezzi and Fortin [25], Chung [26]). Owing to the split introduced in the equations, the method is referred to as the CBS scheme.

The CG procedure may be applied to the individual momentum components without removing the pressure term, provided the pressure term is treated as a source term. However, such a procedure will lose the advantages mentioned in the previous paragraph. For more mathematical details, please refer to Zienkiewicz et al. [20], Zienkiewicz and Codina [23], Nithiarasu [27] and Zienkiewicz et al. [28].

In order to apply the CG procedure, we can refer to the general case of governing generalized porous medium flow and heat transfer equations in non-dimensional form and indicial notation, for mixed convection, that have been presented in Section 1.3 (see equations (24)–(26)).

From the governing equations, it is obvious that the application of the CG scheme is not straightforward. However, by implementing the following procedure, it is possible to obtain a solution to the convection heat transfer porous medium equations. The solution of the free fluid flow equations is obtained by applying the same procedure.

Temporal discretization

For the sake of simplicity, the asterisks are omitted and the Darcy and Forchheimer terms in equation (25) are grouped as a “porous” term to obtain:

$$P = \left(\frac{1}{\text{ReDa}} + \frac{F}{\sqrt{\text{Da}}} |\mathbf{u}| \right) \quad (72)$$

Temporal discretization along the characteristics of continuity, momentum and energy conservation equations results in the following set of equations:

$$\frac{\partial u_i^{n+\vartheta_1}}{\partial x_i} = 0 \quad (73)$$

$$\begin{aligned} u_i^{n+1}(1 + \Delta t \varepsilon P \vartheta_3) - u_i^n(1 - \Delta t \varepsilon P(1 - \vartheta_3)) \\ = -\varepsilon \Delta t \left(\vartheta_2 \frac{\partial p_f^{n+1}}{\partial x_i} + (1 - \vartheta_2) \frac{\partial p_f^n}{\partial x_i} \right) - \left[\frac{\Delta t}{\varepsilon} u_j \frac{\partial u_i}{\partial x_j} \right]^n + \left[\Delta t \frac{J}{\text{Re}} \frac{\partial^2 u_i}{\partial x_i^2} \right]^n \\ + \varepsilon \frac{\Delta t^2}{2} \left[\frac{1}{\varepsilon^2} u_k \frac{\partial}{\partial x_k} \left(u_j \frac{\partial u_i}{\partial x_j} \right) + u_k \frac{\partial}{\partial x_k} \frac{\partial p_f}{\partial x_i} \right]^n + \varepsilon \Delta t \frac{\text{Ra}}{\text{Pr Re}^2} T^n \gamma_i \end{aligned} \quad (74)$$

$$\frac{T^{n+1} - T^n}{\Delta t} = - \left[u_i \frac{\partial T}{\partial x_i} \right]^n + \left[\frac{1}{\sigma \text{Re} \text{Pr}} \frac{\partial^2 T}{\partial x_i^2} \right]^n + \frac{\Delta t}{2\sigma} \left[u_k \frac{\partial}{\partial x_k} \left(u_i \frac{\partial T}{\partial x_i} \right) \right]^n \quad (75)$$

In equations (74) and (75), additional stabilization terms appear naturally from the discretization along the characteristics (Zienkiewicz et al. [20]), while the characteristic parameters ϑ_1 , ϑ_2 and ϑ_3 ($0.5 \leq \vartheta_1 \leq 1$ and $0 \leq \vartheta_i \leq 1$, with $i = 2, 3$) are defined according to the following:

$$\begin{aligned} \frac{\partial u_i^{n+\vartheta_1}}{\partial x_i} &= \vartheta_1 \frac{\partial u_i^{n+1}}{\partial x_i} + (1 - \vartheta_1) \frac{\partial u_i^n}{\partial x_i}, \\ \frac{\partial p_f^{n+\vartheta_2}}{\partial x_i} &= \vartheta_2 \frac{\partial p_f^{n+1}}{\partial x_i} + (1 - \vartheta_2) \frac{\partial p_f^n}{\partial x_i}, \\ [Pu_i]^{n+\vartheta_3} &= \vartheta_3 [Pu_i]^{n+1} + (1 - \vartheta_3) [Pu_i]^n. \end{aligned} \quad (76)$$

Different versions of the CBS scheme can be obtained depending on the value of the above parameters. In particular an SI and an AC version of the CBS scheme can be obtained by varying the parameter ϑ_2 . For ϑ_2 between 0.5 and 1, the SI-CBS is obtained, while for ϑ_2 equal to 0, the AC-CBS scheme is derived. Moreover, for ϑ_3 between 0.5 and 1, an implicit treatment for the porous term is obtained, while for ϑ_3 equal to 0, an explicit one is derived.

The splitting in the CBS scheme consists of solving the above equations in a number of subsequent steps. In the first step, the pressure term is removed from equation (74) and the intermediate velocity components \tilde{u}_i are obtained from:

Step 1: Intermediate velocity calculation

$$\begin{aligned} \tilde{u}_i(1 + \Delta t \varepsilon P \vartheta_3) - u_i^n(1 + \Delta t \varepsilon P(\vartheta_3 - 1)) &= - \left[\frac{\Delta t}{\varepsilon} u_j \frac{\partial u_i}{\partial x_j} \right]^n + \left[\Delta t \frac{J}{\text{Re}} \frac{\partial^2 u_i}{\partial x_i^2} \right]^n \\ &+ \varepsilon \frac{\Delta t^2}{2} \left[\frac{1}{\varepsilon^2} u_k \frac{\partial}{\partial x_k} \left(u_j \frac{\partial u_i}{\partial x_j} \right) \right]^n + \varepsilon \Delta t \frac{\text{Ra}}{\text{Pr Re}^2} T^n \gamma_i \end{aligned} \quad (77)$$

Removing the pressure term from the momentum equation, the pressure stability is enhanced and the use of arbitrary interpolation functions for both velocity and pressure is allowed. In other words, the well-known Babuska–Brezzi condition is satisfied (Babuska [24], Brezzi and Fortin [25], Chung [26]). The correct velocities can be determined, once the pressure field is known, using the equation:

Step 3: Velocity correction

$$u_i^{n+1} - \tilde{u}_i = -\frac{1}{\left(\frac{1}{\varepsilon \Delta t} + P \vartheta_3\right)} \left[\vartheta_2 \frac{\partial p_f^{n+1}}{\partial x_i} + (1 - \vartheta_2) \frac{\partial p_{\text{art}}^{it}}{\partial x_i} + \left(\frac{\Delta t}{2} u_k \frac{\partial}{\partial x_k} \frac{\partial p_f}{\partial x_i} \right)^n \right] \quad (78)$$

The solution of equation (78) is the third step of the algorithm.

The second step consists in the pressure calculation through the continuity equation. This second step is where the SI and AC versions of the scheme differ. In particular, this second step is obtained in the SI version of the scheme by deriving equation (78) with respect to x_i and imposing equation (73), obtaining the following Poisson type of equation:

Step 2 SI-CBS: Pressure calculation

$$0 = -\Delta t \frac{1}{\varepsilon} \left(\vartheta_1 \frac{\partial \tilde{u}_i}{\partial x_i} + (1 - \vartheta_1) \frac{\partial u_i^n}{\partial x_i} \right) + \Delta t^2 \frac{\partial^2 p_f^{n+1}}{\partial x_i^2} \quad (79)$$

Therefore, in this case, the incompressibility constraint is satisfied at each iteration, and the pressure, evaluated through equation (79), represents the actual pressure. However, the solution of equation (79) needs a matrix inversion. Furthermore, the SI version of the scheme uses a global time step, which is the minimum value of the time step limit over the entire domain.

Alternatively, the AC scheme can be derived when the left hand side of equation (79) is obtained from a mass conservation equation retaining the transient density term:

$$\frac{\partial \rho_f}{\partial t} + \frac{1}{\varepsilon} \frac{\partial (\rho_f u_i)}{\partial x_i} = 0 \quad (80)$$

In general, it is possible to relate the density time variation to the pressure time variation, through the speed of sound, as follows:

$$\frac{\partial \rho_f}{\partial t} = \frac{1}{c^2} \frac{\partial p_f}{\partial t} \quad (81)$$

where the real compressibility parameter, c (compressible wave speed), approaches infinity for many incompressible flow problems and the solution scheme becomes stiff and imposes severe time step restrictions. However, this parameter can be replaced locally by an appropriate artificial value, β , of finite value, employing the

AC method, that was first introduced by Chorin [29] and then further developed [30–42]:

$$\frac{\partial \rho_f}{\partial t} = \frac{1}{\beta^2} \frac{\partial p_{\text{art}}}{\partial t} \Rightarrow \frac{1}{\beta^2} \frac{p_{\text{art}}^{it+1} - p_{\text{art}}^{it}}{\Delta t} = -\frac{\rho_f}{\varepsilon} \frac{\partial u_i^{it+\vartheta_1}}{\partial x_i} \quad (82)$$

In equation (82), the pressure is an artificial pressure, because the incompressibility constraint is not achieved at each time step, but only when steady-state convergence is reached. The superscripts $it + 1$ and it are referred to the iterative procedure and do not refer to real time levels. The second step of the AC scheme is carried out by imposing the equation (82) as:

Step 2 AC-CBS: Pressure calculation

$$\frac{1}{\beta^2} (p_{\text{art}}^{it+1} - p_{\text{art}}^{it}) = -\Delta t \frac{1}{\varepsilon} \left(\vartheta_1 \frac{\partial \tilde{u}_i}{\partial x_i} + (1 - \vartheta_1) \frac{\partial u_i^{it}}{\partial x_i} \right) + \Delta t^2 \frac{\partial^2 p_{\text{art}}^{it}}{\partial x_i^2} \quad (83)$$

The local value of β is calculated through the following procedure:

$$\beta = \max(0.5, u_{\text{conv}}, u_{\text{diff}}, u_{\text{ther}}) \quad (84)$$

The local convective, diffusive and thermal velocities can be calculated through the following non-dimensional relations:

$$u_{\text{conv}} = \sqrt{u_i u_i}, u_{\text{diff}} = \frac{2}{h \text{Re}}, u_{\text{ther}} = \frac{2}{h \text{Re} \text{Pr}}. \quad (85)$$

The diffusive time step limitation for the AC method may be written as $h^2/\text{Re}/2$, while the convective time step limitation may be written as:

$$\Delta t_{\text{conv}} = \frac{h}{|\mathbf{u}| + \beta} \quad (86)$$

The above relation includes the viscous effect via the artificial parameter.

Depending on the problem of interest, it is possible to consider other equations coupled to the above set, such as species concentration for multi-component flows.

The fourth step, for non-isothermal problems, is represented by the energy conservation equation (75) that allows calculating the temperature at every iteration for problems with a coupling between momentum and energy conservation equations occurs.

Spatial discretization and solution methodology

The spatial discretization of the governing equations is obtained through Galerkin finite element procedure and triangular elements. Within an element, each variable is calculated through linear approximation on the basis of nodal values, according to the following equation:

$$\phi = \sum_{n=1}^3 N_n \bar{\phi}_n \quad (87)$$

where N_n is the shape function at node n and $\bar{\phi}_n$ is the value of the generic variable ϕ at node n . Applying the standard Galerkin procedure to the set of equations presented in the previous section, the following four steps, expressed in a matrix form, are obtained:

Step 1: Intermediate velocity calculation

$$\begin{aligned} \mathbf{M}\tilde{u}_i = & (1 - \Delta t \varepsilon P) \mathbf{M}u_i^n - \frac{\Delta t}{\varepsilon} [\mathbf{C}u_i]^n - \Delta t \frac{J}{\text{Re}} [\mathbf{K}_d u_i]^n - \frac{1}{2} \frac{\Delta t^2}{\varepsilon} [\mathbf{K}_u u_i]^n \\ & + \varepsilon \Delta t \left[\frac{\text{Ra}}{\text{Pr} \text{Re}^2} \mathbf{M}T \gamma_i \right]^n + \Delta t \frac{J}{\text{Re}} [\mathbf{f}_d]^n + \frac{1}{2} \frac{\Delta t^2}{\varepsilon} [\mathbf{f}_u]^n \end{aligned} \quad (88)$$

Step 2 SI-CBS: Pressure calculation

$$[\mathbf{K}_p p_f]^{n+1} = -\frac{1}{\varepsilon \Delta t} [\mathbf{D}\tilde{u}_i] \quad (89)$$

Step 2 AC-CBS: Pressure calculation

$$\frac{1}{\beta^2} \mathbf{M}(p_{\text{art}}^{it+1} - p_{\text{art}}^{it}) = -\frac{\Delta t}{\varepsilon} [\mathbf{D}\tilde{u}_i] - \Delta t^2 [\mathbf{K}_p p_{\text{art}}]^{it} \quad (90)$$

Step 3: Velocity correction

$$\mathbf{M}(u_i^{n+1} - \tilde{u}_i) = -\varepsilon \Delta t \left[\vartheta_2 \mathbf{D}p_f^{n+1} + (1 - \vartheta_2) \mathbf{D}p_{\text{art}}^{it} - \frac{\Delta t}{2} \mathbf{K}_u p_f^n \right] \quad (91)$$

Step 4: Temperature calculation

$$\begin{aligned} \mathbf{M}T^{n+1} = & \mathbf{M}T^n - \Delta t [\mathbf{C}T]^n - \frac{\Delta t \lambda}{\text{Re} \text{Pr} R_k} [\mathbf{K}_d T]^n \\ & - \frac{\Delta t^2}{2R_k} [\mathbf{K}_u T]^n + \frac{\Delta t \lambda}{\text{Re} \text{Pr} R_k} [\mathbf{f}_t]^n + \frac{\Delta t^2}{2R_k} [\mathbf{f}_{ut}]^n \end{aligned} \quad (92)$$

where \mathbf{M} is the mass matrix, \mathbf{C} is the convection matrix, \mathbf{K}_d is the diffusion matrix, \mathbf{K}_u is the stabilization matrix obtained from higher-order terms, \mathbf{f}_d and \mathbf{f}_u are the boundary vectors from the momentum equation, \mathbf{K}_p is the stiffness matrix, \mathbf{D} is the gradient matrix, \mathbf{f}_t and \mathbf{f}_{ut} are the boundary vectors from the temperature equation. Details of all the terms presented in equations (88)–(92) are given in Lewis et al. [19] and Zienkiewicz et al. [20].

In the present procedure, the mass matrix is lumped using a standard row-summing approach, inverted and stored in an array during the pre-processing. Therefore, in the AC formulation, the calculation of the artificial pressure at $it + 1$ iteration, through equation (90), does not need a matrix inversion and the result is a matrix inversion-free procedure. On the other hand, in the second step of the SI solution procedure, the stiffness matrix needs to be inverted at each iteration.

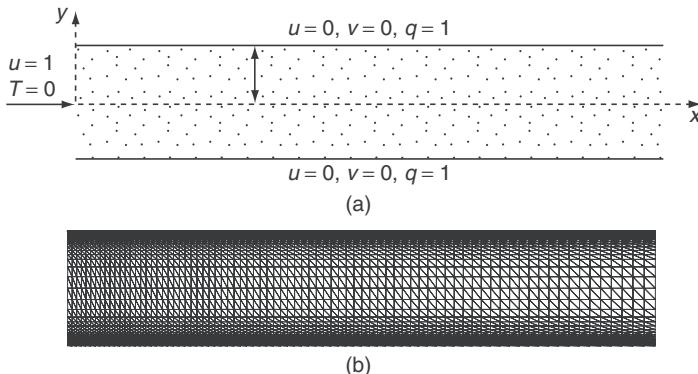


Figure 4.31. Forced convection in a porous channel. (a) Computational domain and boundary condition; (b) structured computational grid (3,321 nodes, 6,400 elements).

Some examples

Both the schemes presented above can be used to solve many engineering problems, such as mixed, natural or forced convection both in fluid-saturated porous media and in partly porous domains, or simply in free fluid flow cases.

As mentioned before, changing the properties such as porosity and thus permeability, it is possible to handle porous medium-free fluid interface problems as a single problem with different properties. The following limits are used for the porous medium part and for the free fluid part:

$$\begin{aligned} \varepsilon < 1 & \Rightarrow \text{porous medium} & \varepsilon = 1 \\ \text{Da} = \text{finite} & \Rightarrow \text{free fluid} & \text{Da} \rightarrow \infty \end{aligned} \quad (93)$$

A suitable set of matching conditions, to connect the porous medium and the free fluid domains, is needed (Massarotti et al. [43]).

The first example is the forced convection heat transfer inside a uniform porous channel with constant wall heat flux. The computational domain, together with the boundary conditions employed, is sketched in Figure 4.31a, while Figure 4.31b shows the computational grid employed. The flow enters the domain from the left with a constant velocity and at a non-dimensional temperature $T = 0$. The channel walls are heated by a constant heat flux. The numerical results have been obtained using a non-uniform grid with 6,400 elements and 3,321 nodes.

Figure 4.32 shows the non-dimensional velocity and temperature profiles, for different Darcy numbers, varying from 10^{-4} to 1.0 at a section, where the flow and the thermal field are hydrodynamically and thermally developed. The results have been compared with the analytical solutions presented by Nield et al. [44] and Lauriat and Vafai [45]. The dimensionless temperature is evaluated using:

$$T_{\text{mix}} = \frac{1}{u_{\text{mean}} S_x} \int_{S_x} u_i T_i dS \quad (94)$$

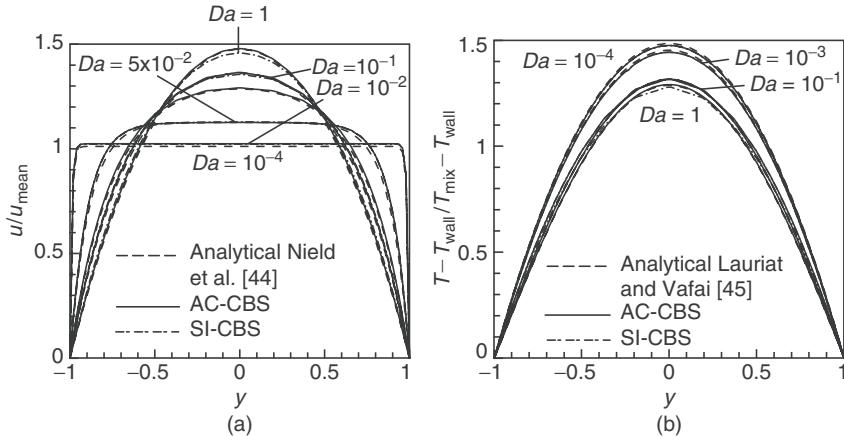


Figure 4.32. Forced convection in a porous channel: (a) non-dimensional velocity as function of transverse distance; (b) non-dimensional temperature as function of transverse distance.

where u_{mean} is the mean velocity in the section considered S_x .

For a small Darcy number (10^{-4}), the velocity profile is nearly independent of the transverse distance and slip flow occurs at the walls (Figure 4.32). An increase in Darcy number leads to a non-linear distribution of the velocity. When the Darcy number is equal to unity, the velocity profile approaches a profile similar to that of free fluid flow.

The non-dimensional temperature increases as the Darcy number decreases, as shown in Figure 4.32. A decrease in the Darcy number corresponds to a decrease in the fluid velocity in the middle of the channel and therefore the conduction heat transfer is dominating over the effect of convection.

Figure 4.33 shows the convergence histories to steady state for the AC-CBS and SI-CBS schemes obtained by using the following L_2 norm for velocities:

$$\text{Velocity residual} = \sqrt{\sum_{i=1}^{\text{nodes}} \left(\frac{|\mathbf{u}_i|^{n+1} - |\mathbf{u}_i|^n}{\Delta t_i} \right)^2} \Bigg/ \sqrt{\sum_{i=1}^{\text{nodes}} \left(\frac{|\mathbf{u}_i|^{n+1}}{\Delta t_i} \right)^2} \quad (95)$$

This figure shows that the SI version of the scheme converges faster as the Darcy number decreases. Instead, the AC version of the scheme has an opposite behaviour. Table 4.1 reports the CPU time needed by both the procedures to reach steady state solution, on a machine with 4 Gb of RAM and 2.4 GHz CPU speed. These results confirm the behaviour shown in Figure 4.33. In particular, Table 4.1 shows that the AC-CBS scheme converges faster than the SI-CBS scheme when $Da \geq 10^{-2}$. It was noticed that the SI-CBS scheme needs more time than the AC-CBS scheme per time iteration. In correspondence of smaller Darcy numbers ($Da \leq 10^{-3}$), the AC-CBS scheme takes more time to reach the steady state. This is due to the time step

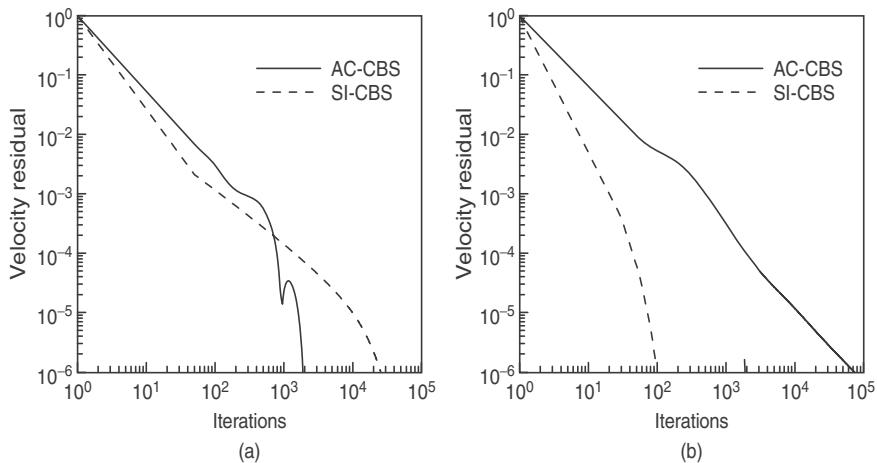


Figure 4.33. Convergence histories for SI and AC scheme. (a) $Da = 1$; (b) $Da = 10^{-4}$.

Table 4.1. CPU time (s) for forced convection in a porous channel at different Darcy numbers

Da	1	10^{-1}	5×10^{-2}	10^{-2}	10^{-3}	10^{-4}
AC-CBS	35	87	137	491	4,341	6,400
SI-CBS	14,730	10,250	8,055	3,357	1,107	250

calculation procedure used. Essentially, the AC scheme becomes slower in reaching the steady state when diffusion is overriding the effect of convection ($Da \leq 10^{-3}$) and the local time step approaches the global time step, losing the advantage of using a higher time step in the convective zones.

The second example is the natural convection in a cavity heated uniformly from the bottom side. The computational domain, together with the boundary conditions employed, is sketched in Figure 4.34a, while Figure 4.34b shows the computational grid employed, composed of 7,200 triangular elements and 3,721 nodes.

Figure 4.35 shows the dimensionless temperature contours for a Prandtl number equal to 0.71, and for two different Rayleigh numbers (10^6 and 7×10^4) and two different Darcy numbers (10^{-3} and 10^{-4}). The results obtained have been compared with the numerical solution presented by Basak et al. [46].

In general, the fluid circulation is strongly dependent on the Darcy number. In fact, when smaller Rayleigh and Darcy numbers are considered, the flow is very weak and the temperature distribution is similar to that of stationary fluid (Figure 4.35a). As the Darcy number increases, the role of convection becomes

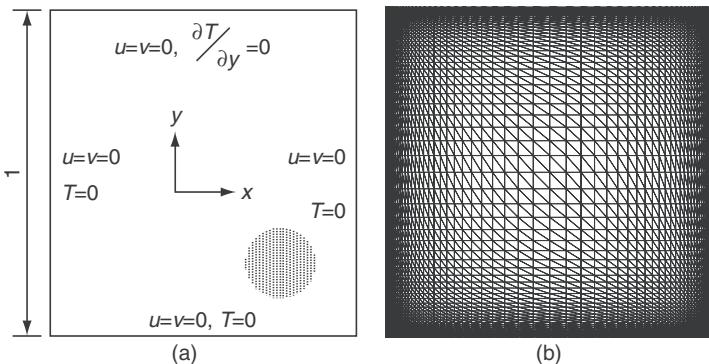


Figure 4.34. Natural convection in a porous cavity heated from the bottom side: (a) computational domain and boundary conditions; (b) structured computational grid (3,721 nodes, 7,200 elements).

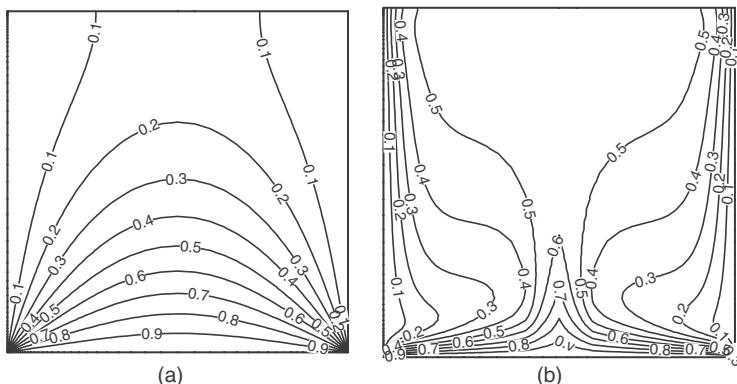


Figure 4.35. Temperature contours for natural convection in a porous cavity heated from the bottom side. (a) $\text{Ra} = 7 \times 10^4$ and $\text{Da} = 10^{-4}$; (b) $\text{Ra} = 10^6$ and $\text{Da} = 10^{-3}$.

more significant and the fluid rises up strongly from the middle portion of the bottom wall, as depicted in Figure 4.35b.

Figure 4.36 shows the steady-state convergence histories for the AC-CBS and the SI-CBS schemes. This figure shows a different behaviour from that of the case of forced convection flow problem. When natural convection occurs, the AC-CBS scheme converges faster than the SI-CBS scheme for any Rayleigh or Darcy number, both in terms of number of iterations (Figure 4.36) and CPU time to reach the steady state (Table 4.2).

The SI-CBS algorithm takes more time to reach the steady state because of the coupling between momentum and energy conservation equations present in natural convection problems. The coupling is due to the presence of a generation term on

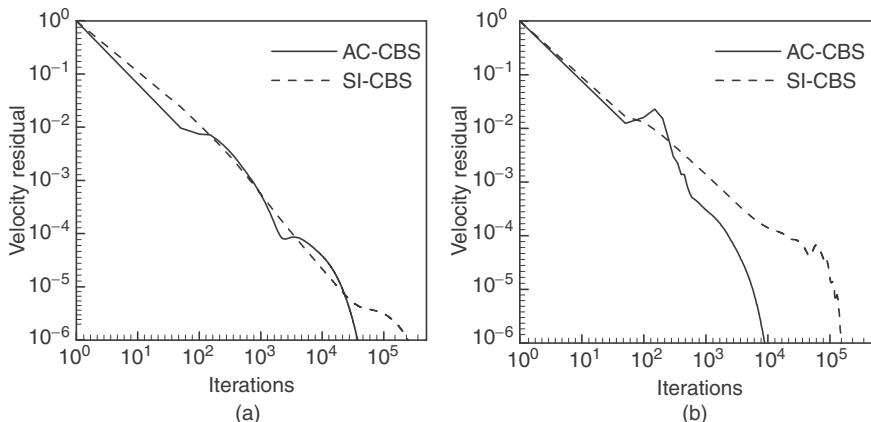


Figure 4.36. Convergence histories for SI and AC. (a) $\text{Ra} = 7 \times 10^4$ and $\text{Da} = 10^{-4}$; (b) $\text{Ra} = 10^6$ and $\text{Da} = 10^{-3}$.

Table 4.2. CPU time (s) for natural convection in a porous cavity heated from the bottom at different Rayleigh and Darcy numbers

Ra	10^6		7×10^4	
	10^{-3}	10^{-4}	10^{-3}	10^{-4}
AC-CBS	180	930	290	700
SI-CBS	63,270	48,230	53,400	47,560

the right hand side of the momentum conservation equation in gravity direction that must be evaluated at each time step. This term can be calculated only after the resolution of the energy equation (step 4 of the CBS algorithm), due to the fact that the nodal temperature values are unknown. This coupled system becomes stiff and causes a restriction on the global time step value used in the SI procedure and thus the solution to the simultaneous equations at the second step of the algorithm needs more time.

Moreover, the SI scheme experiences difficulties in reaching the steady state when very refined meshes are used (Massarotti et al. [47]). On the other hand, the AC-CBS scheme does not show any problem in reaching the convergence when natural convection flow occurs. The reason for this is the robust local time stepping procedure used by the AC-CBS scheme.

The last example is the developing mixed convection in a region partially filled with a fluid-saturated porous medium, confined between two vertical hot walls. In particular, because of the geometrical symmetry, half domain has been studied. Figure 4.37 shows the computational domain and the boundary conditions employed and the details of the computational grid near the entrance. The mesh employed has

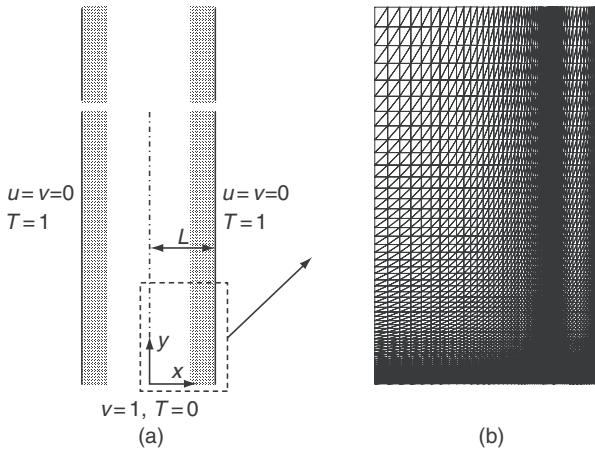


Figure 4.37. Mixed convection in a vertical channel partially filled with a porous medium. (a) Computational domain and boundary conditions; (b) detail of the structured computational grid near the entrance (8,591 nodes and 16,800 elements).

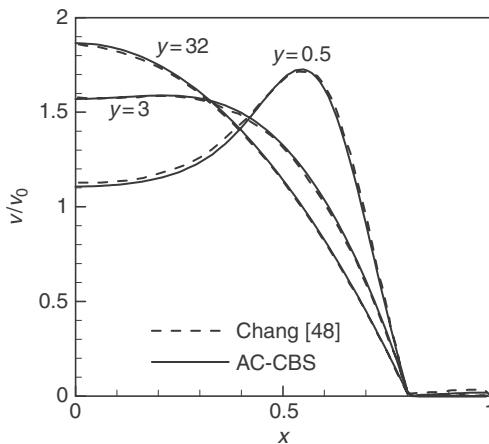


Figure 4.38. Mixed convection in a vertical parallel plate channel partially filled with a porous medium. Non-dimensional vertical velocity at different heights of the channel and at $\text{Da} = 10^{-5}$.

8,591 nodes and 16,800 elements and is refined near the wall, near the inlet region and at the interface.

Figure 4.38 shows the dimensionless vertical velocity profile at different heights of the channel (y) for a Darcy number equal to 10^{-5} , obtained by using the AC-CBS scheme. The present results are compared with the numerical results of Chang and

Chang [48]. The parameters used for the present investigation are $\text{Ra} = 0.72 \times 10^3$, $\text{Pr} = 0.72$, $\text{Re} = 50$, $\lambda = 2.8$ in the porous region and $\lambda = 1$ in the free fluid region, $\varepsilon = 0.8$. When y is small, there is a large velocity difference at the interface of the composite system (Figure 4.38). When y increases, the flow discharge in the porous layer decreases, and the peak of the fluid velocity profile moves to the central axis of the vertical parallel-plate channel.

References

- [1] Denham, M. K., and Patrik, M. A. Laminar flow over a downstream-facing in a two-dimensional flow channel. *Transactions of the Institution of Chemical Engineers*, 52, pp. 361–367, 1974.
- [2] Aung, W. An experimental study of laminar heat transfer downstream of backsteps. *Journal of Heat Transfer*, 105, pp. 823–829, 1983.
- [3] Ichinose, K., Tokunaga, H., and Satofuka, N. Numerical simulation of two-dimensional backward-facing step flows. *Transactions of JSME B*, 57(543), pp. 3715–3721, 1991.
- [4] de Sampaio, P. A. B., Lyra, P. R. M., Morgan, K., and Weatherill, N. P. Petrov–Galerkin solutions of the incompressible Navier–Stokes equations in primitive variables with adaptive remeshing. *Computer Methods in Applied Mechanics and Engineering*, 106, pp. 143–178, 1993.
- [5] Massarotti, N., Nithiarasu, P., and Zienkiewicz, O. C. Characteristic-Based-Split (CBS) algorithm for incompressible flow problems with heat transfer. *International Journal for Numerical Methods in Heat and Fluid Flow*, 8, pp. 969–990, 1998.
- [6] Morgan, K., Weatherill, N. P., Hassan, O., Brookes, P. J., Said, R., and Jones, J. A parallel framework for multidisciplinary aerospace engineering simulations using unstructured meshes. *International Journal for Numerical Methods in Fluids*, 31, pp. 159–173, 1999.
- [7] Spalart, P. R., and Allmaras, S. R. A one-equation turbulence model for aerodynamic flows. *AIAA 30th Aerospace Sciences Meeting*, AIAA paper 92-0439, 1992.
- [8] Nithiarasu, P., and Liu, C.-B. An artificial compressibility based characteristic based split (CBS) scheme for steady and unsteady turbulent incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 195, pp. 2961–2982, 2006.
- [9] Nithiarasu, P., Liu, C.-B., and Massarotti, N. Laminar and turbulent flow calculations through a model human upper airway using unstructured meshes. *Communications in Numerical Methods in Engineering*, 23, pp. 1057–1069, 2007.
- [10] Denham, M. K., Briard, P., and Patrick, M. A. A directionally-sensitive laser anemometer for velocity measurements in highly turbulent flows. *Journal of Physics E: Scientific Instruments*, 8, pp. 681–683, 1975.
- [11] Gemci, T., Corcoran, T. E., Yakut, K., Shortall, B., and Chigier, N. Spray dynamics and deposition of inhaled medications in the throat, *ILASS-Europe*, Zurich, 2–6 September, 2001.
- [12] Li, W.-I., Perzl, M., Heyder, J., Langer, R., Brain, J. D., Englmeier, K.-H., Niven, R. W., and Edwards, D. A. Aerodynamics and aerosol particle deaggregation phenomena in model oral-pharyngeal cavities. *Journal of Aerosol Science*, 27, pp. 1269–1286, 1996.
- [13] Li, W.-I., and Edwards, D. A. Aerosol particle transport and deaggregation phenomena in the mouth and throat. *Advanced Drug Delivery Reviews*, 26, pp. 41–49, 1997.
- [14] Martonen, T. B., Zhang, Z., Yue, G., and Musante, C. J. 3-D particle transport within the human upper respiratory tract. *Journal of Aerosol Science*, 33, pp. 1095–1110, 2002.

- [15] Whitaker, S. Diffusion and dispersion in porous media. *AICHE Journal*, 13, pp. 420–427, 1961.
- [16] Vafai, K., and Tien, C. L. Boundary and inertia effects on flow and heat transfer in porous media. *International Journal of Heat Mass Transfer*, 24, pp. 195–203, 1981.
- [17] Hsu, C. T., and Cheng, P. Thermal dispersion in a porous medium. *International Journal of Heat Mass Transfer*, 33, pp. 1587–1597, 1990.
- [18] Chen, Y. C., Chung, J. N., Wu, C. S., and Lue, Y. F. Non-Darcy mixed convection in a vertical channel filled with a porous medium. *International Journal of Heat Mass Transfer*, 43, pp. 2421–2429, 2000.
- [19] Lewis, R. W., Nithiarasu, P., and Seetharamu, K. N. *Fundamentals of the Finite Element Method for Heat and Fluid Flow*, Wiley, Chichester, 2004.
- [20] Zienkiewicz, O. C., Taylor, R. L., and Nithiarasu, P. *The Finite Element Method for fluid dynamics*, 6th Edition, Elsevier Butterworth-Heinemann, Oxford 2005.
- [21] Brooks, A. N., and Hughes, T. J. R. Streamline upwind/Petrov-Galerkin formulation for convection dominated flows with particular emphasis on incompressible Navier–Stokes equation. *Computer Methods in Applied Mechanics and Engineering*, 32, pp. 199–259, 1982.
- [22] Hirsch, C. *Numerical Computation of Internal and External Flows*, 1, *Fundamentals of Computational Fluid Dynamics*, 2nd ed., John Wiley & Sons, Oxford, 2007.
- [23] Zienkiewicz, O. C., and Codina, R. A general algorithm for compressible and incompressible flow, Part I, The split characteristic based scheme. *International Journal for Numerical Methods in Fluids*, 20, pp. 869–885, 1995.
- [24] Babuska, I. The finite element method with Lagrange multipliers, *Numerische Mathematik*, 20, pp. 179–192, 1973.
- [25] Brezzi, F., and Fortin, M. *Mixed and Hybrid Finite Element Methods*, Springer, Berlin, 1991.
- [26] Chung, T. J. *Computational Fluid Dynamics*, Cambridge University Press, Cambridge, 2001.
- [27] Nithiarasu, P., An efficient artificial compressibility (AC) scheme based on the characteristic based split (CBS) method for incompressible flows, *International Journal for Numerical Methods in Engineering*, 56, pp. 1815–1845, 2003.
- [28] Zienkiewicz, O. C., Nithiarasu, P., Codina, R., Vazquez, M., and Ortiz, P. An efficient and accurate algorithm for fluid mechanics problems, the characteristic based split (CBS) algorithm. *International Journal for Numerical Methods in Fluids*, 31, pp. 359–392, 1999.
- [29] Chorin, A. J. A numerical method for solving incompressible viscous flow problems. *Journal of Computational Physics*, 2, pp. 12–26, 1967.
- [30] Malan, A. G., Lewis, R. W., and Nithiarasu, P. An improved unsteady, unstructured, artificial compressibility, finite volume scheme for viscous incompressible flows: Part II. Application. *International Journal of Numerical Methods in Engineering*, 54, pp. 715–729, 2002.
- [31] Turkel, E. Review of preconditioning methods for fluid dynamics. *Applied Numerical Mathematics*, 12, pp. 257–284, 1993.
- [32] Weiss, J. W., and Smith, W. A. Preconditioning applied to variable and constant density flows. *AIAA Journal*, 33, pp. 2050–2057, 1995.
- [33] Gaitonde, A. L. A dual time method for two dimensional incompressible flow calculations. *International Journal of Numerical Methods in Engineering*, 41, pp. 1153–1166, 1998.

- [34] Choi, Y. H., and Merkle, C. L. The application of preconditioning in viscous flows. *Journal of Computational Physics*, 105, pp. 207–223, 1993.
- [35] Turkel, E. Preconditioning methods for solving the incompressible and low speed compressible equations. *Journal of Computational Physics*, 72, pp. 227–298, 1987.
- [36] Cabak, H., Sung, C. H., and Modi, V. Explicit Runge-Kutta method for three-dimensional internal incompressible flows. *AIAA Journal*, 30, pp. 2024–2031, 1992.
- [37] Tamamidis, P., Zhang, G., and Assanis, D. N. Comparison of pressure-based and artificial compressibility methods for solving 3D steady incompressible viscous flows. *Journal of Computational Physics*, 124, pp. 1–13, 1996.
- [38] Lin, F. B., and Sotiropoulos, F. Assessment of artificial dissipation models for three-dimensional incompressible flow solutions. *Journal of Fluids Engineering (ASME)*, 119, pp. 331–340, 1997.
- [39] Zhao, Y., and Zhang, B. L. A high-order characteristics upwind FV method for incompressible flow and heat transfer simulation on unstructured grids. *Computer Methods in Applied Mechanics and Engineering*, 190, pp. 733–756, 2000.
- [40] Manzari, M. T. An explicit finite element algorithm for convection heat transfer problems. *International Journal of Numerical Methods Heat Fluid Flow*, 9, pp. 860–877, 1999.
- [41] Kim, E. A mixed Galerkin method for computing the flow between eccentric rotating cylinders. *International Journal of Numerical Methods in Fluids*, 26, pp. 877–885, 1998.
- [42] Wasfy, T., West, A. C., and Modi, V. Parallel finite element computation of unsteady incompressible flows. *International Journal of Numerical Methods in Fluids*, 26, pp. 17–37, 1998.
- [43] Massarotti, N., Nithiarasu, P., and Zienkiewicz, O. C. Natural convection in porous medium-fluid interface problems – A finite element analysis by using the CBS procedure. *International Journal of Numerical Methods for Heat and Fluid Flow*, 11(5), pp. 473–490, 2001.
- [44] Nield, D. A., Kuznetsov, A. V., and Xiong, M. Effect of local thermal non-equilibrium on thermally developing forced convection in a porous medium. *International Journal of Heat Mass Transfer*, 45, pp. 4949–4955, 2002.
- [45] Lauriat, G., and Vafai, K. Forced convection and heat transfer through a porous medium exposed to a flat plate or a channel. *Convective Heat and Mass Transfer in Porous Media*, Kluwer Academic, Dordrecht, pp. 289–327, 1991.
- [46] Basak, T., Roy, S., Paul, T., and Pop, I. Natural convection in a square cavity filled with a porous medium: Effects of various thermal boundary conditions. *International Journal of Heat Mass Transfer*, 49, pp. 1430–1441, 2006.
- [47] Massarotti, N., Arpino, F., Lewis, R. W., and Nithiarasu, P. Explicit and semi-implicit CBS procedures for incompressible viscous flows. *International Journal of Numerical Methods in Engineering*, 66(10), pp. 1618–1640, 2006.
- [48] Chang, W.-J., and Chang, W.-L. Mixed convection in a vertical parallel-plate channel partially filled with porous media of high permeability. *International Journal of Heat and Mass Transfer*, 39(7), pp. 1331–1342, 1996.

5 Equal-order segregated finite-element method for fluid flow and heat transfer simulation

Jianhui Xie

*MFG-Engineering Design & Simulation, Autodesk, Inc.,
San Rafael, CA, USA*

Abstract

This chapter introduces finite element-based equal-order mixed-GLS (Galerkin Least Squares) segregated formulation for fluid flow and heat transfer. This scheme follows the idea used in finite-volume method, which decouples the fluid pressure calculation and velocity calculation by taking the divergence of the vector momentum equation and applying some clear insights regarding incompressible flow. The inherent velocity correction step in “segregated scheme” makes it attractive for solution robustness and much faster compared to original fluid velocity–pressure coupling scheme because mixed-GLS finite-element formulation does not assure the mass conservation between the interface of the elements in comparison to finite-volume method. With these advantages, this formulation has prevailed in recent years in major finite element-based commercial codes used in the industry simulation models that usually import geometry from CAD and have large scale of degree of freedoms to solve. Taking example of fluid–thermal coupling problem, this chapter also discusses the strategies used to solve multiphysics problem in nonlinear iteration level and the block I/O technique that uses finite-element method for handling massive data. Finally, this chapter illustrates some typical industry conjugate heat transfer problems where both solid and fluid parts are involved.

Keywords: Finite elements, Fluid flow, Heat transfer, Segregated formulation

5.1 Introduction

The quest for robust and efficient strategies to solve large, sparse matrix systems has been an active area of research and development for decades. Galerkin finite-element methods, with the advantages of handling irregular geometry and multiphysics coupling, have played an important role in computational fluid dynamics area, but its computational difficulties and perceived shortcomings have led to the development of alternative finite-element models.

Theoretically, the efforts have been to avoid the two major problems encountered. The first problem is the “wiggles” velocity field problem related to the node-to-node oscillation of velocity components emanating from boundaries with large velocity gradient. This problem is mostly pronounced under the condition of high Reynolds numbers and coarse computational mesh, and therefore the inability of the finite-element mesh to handle the steep gradient results in an imbalance between advective and diffusive contribution. With getting best fit to the problem, the weighted residual formulation produces a field that oscillates about the true solution (wiggle pattern). The second problem is the “saddle-point” problem originated by mixed Galerkin finite-element formulation. It is difficult to solve especially in three-dimensional large-scale industry model because the LBB (Ladyzhenskaya–Babuska–Brezzi) stability condition associated with the Galerkin finite-element limits element choice with respect to the velocity and pressure interpolation.

In practical aspect, solution schemes/solvers and algorithms have been explored for better CFD processor efficiency and robustness, which are mostly addressed by commercial codes.

In this chapter, the equal-order mixed-GLS (Galerkin Least Squares) stabilized formulation is presented first. It is termed as a residual method because the added least-squares terms are weighted residuals of the momentum equation, and this form of the least-squares terms implies the consistency of the method since the momentum residual is employed. And this method is a generalization of SUPG (streamline-upwind/Petrov–Galerkin) and PSPG (pressure stabilizing/Petrov–Galerkin) methods, which were developed to target the wiggle stabilization. Since this method plays a significant stabilization role for the coarse mesh and hybrid mesh, the equal-order velocity and pressure interpolation is viable, and the LBB condition is not necessary any more, the convenient bilinear element interpolation becomes practical approximation although it was unstable in the Galerkin finite-element context.

The second emphasis in this chapter is on the practical view: The numerical strategies for the solution of large systems of equations arising from the finite-element discretization of the above formulations are discussed. To solve the nonlinear fluid flow and heat transfer problem, particular emphasis is placed on segregated scheme (which uses SIMPLE algorithm from the finite-volume method) in nonlinear level and iterative methods in linear level. The velocity–pressure coupling formulation not only results in a large dimensioned system but also generates a stiffness matrix radically different from the narrowband type of matrix, which is inefficient to be solved. Segregated scheme was proposed with the idea of decoupling the pressure calculation and velocity calculation by taking the divergence of the vector momentum equation and applying some clear insights regarding incompressible flow. The inherent velocity correction step in “segregated scheme” makes it attractive for solution robustness and faster compared to velocity–pressure coupling scheme because mixed-GLS finite-element formulation does not assure the mass conservation between the interface of the elements in comparison to finite-volume method.

According to how the primitive variables of velocity–pressure are treated, finite-element methods for solving Navier–Stokes equations can be categorized into three groups: (a) the velocity–pressure integrated method; (b) the penalty method; and (c) the segregated velocity–pressure method. The velocity–pressure integrated method in which the governing equations are treated simultaneously needs relatively small number of iterations to achieve convergence, but a large memory and computing time. The penalty method requires less memory and less computing time than the first method, but it needs an additional postprocessor to obtain pressure field and satisfies the continuity equation only approximately. Recently, much attention has been paid to the segregated velocity–pressure method, where velocity and the corresponding pressure field are computed alternatively in an iterative sequence. This method takes advantage of SIMPLE algorithm idea from finite-volume world and uses it in the finite-element world. With the SIMPLE algorithm idea, it needs much less memory and execution time, and with the velocity correction step using continuity equation, it even satisfies the continuity equation better than the first method.

The second categorization of finite-element methods can be made according to the orders of interpolation functions for the velocity and pressure. They are mixed-order interpolation and equal-order interpolation. The mixed-order scheme attempts to eliminate the tendency to produce checkerboard pressure distributions by satisfying LBB condition. In this method the velocity is interpolated linearly whereas the pressure is assumed to be constant within the element. Equal-order interpolation was proposed by Rice and Schnipke [6]. They showed that equal-order scheme performed better for that purpose without exhibiting spurious pressure modes.

A third categorization of the finite-element method for fluid dynamics is related to weight functions. Galerkin method has been widely used in discretizing the momentum equation. Brook and Hughes demonstrated the accuracy of the streamline-upwind/Petrov–Galerkin method for the linear advection-diffusion equation; Rice and Schnipke [5] adopted a monotone streamline-upwind finite-element method, where they discretized the momentum equation by using conventional Galerkin method with the exception of the advection items, which were treated by the monotone streamline-upwind approach.

Followed by the success of the finite-volume methods, several finite-element segregated solution schemes have been proposed. Rice and Schnipke [6] employed equal-order interpolation for all variables and solved pressure directly; however, their original formulation used a streamline unwinding scheme that may not be straightforward enough to extend to three-dimensional flows. Early equal-order methods were in general of transient kind and the equations had to be integrated on time to reach a steady state. Van Zijl [11] adapted the formulation by applying SUPG weighting functions to the convective terms; he later adapted the scheme further by implementing SIMPLEST algorithm. Shaw [7] demonstrated the way to use element matrix construction for equal-order interpolation in segregated scheme. Haroutunian et al. [2] have shown that the implementation of iterative solvers can result in a substantial reduction in the storage requirements and execution

time. Du Toit [10] tried conjugate gradient solver; Wang [13,14] demonstrated a “consistent equal-order discretization method” by introducing element-based node velocity to satisfy mass conservation, while keeping conventional node velocity and temperature to satisfy momentum and energy equations. Wansophark [15] evaluated segregated schemes by combining monotone streamline-upwind method and adaptive meshing technique. However, relaxation factors, which need user’s interpretation, have remained an issue that affects the users.

The classical mixed velocity-pressure interpolation Galerkin finite-element method, often referred to as “mixed v–p form,” was the workhorse of incompressible flow solvers in the 1970s to 1990s and has proved highly successful for two-dimensional and small three-dimensional problems, offering high order of accuracy and strong convergence rates [5]. The high mark of this approach is the discretization of the continuity equation in a manner adherent to certain mathematical constraints, and the ability to simultaneously solve the equation with the discretized momentum equations in either a Picard or a Newton–Raphson iterative scheme. The discretized continuity equation is the known cause of “indefiniteness” in the resulting Jacobian matrix, leading to unbounded large and negative eigenvalues. Despite this poor matrix conditioning, most two-dimensional problems with small bandwidth are amenable to efficient solution with optimized variations of classical Gaussian elimination. For example, skyline method allows for memory-efficient solution and in fact is still used today. But even tremendous technological advances in computer processor speed and random access memory capacity will never make direct methods a viable alternative for large three-dimensional problems. This is one reason why many researchers have abandoned the traditional mixed-interpolation approach in favor of alternatives that allow the ready use of iterative solvers. These alternative formulations, including penalty methods, pressure stabilization methods, and pressure projection methods [2], bring “definiteness” to the matrix system.

Transient analysis is a traditional way to overcome poor performance of iterative solvers [9]; the idea is to take advantage of iterative solvers that thrive on a good initial guess – which transient analysis with small-enough time increment delivers at each time step. Furthermore, the smaller the time step the more diagonally dominant the matrix system, because the time derivatives occur on or near the diagonals. It is noticed that the transient analysis is inefficient for a broad class of problems that want steady solution; hence, the result from integrating in time is not really practical as an end-all cure, instead transient term like inertial relaxation is more effective in steady analysis.

In general, finite-element method based simulation has following basic steps:

Preprocessor phase: The preprocessor phase prepares all the required data for the core processor computation; it includes CAD input, mesh generation, and geometry decoder. After three-dimensional modeling from CAD packages, the model geometry and part/assembly relations are established in parameter control.

The CAD input module reads and transfers the CAD data into the mesh ready geometry data with the ordered hierarchy of assembly/part/surface relations.

Mesh generation, which is responsible for generating valid finite-element meshes and possible special meshes (e.g., boundary mesh for fluid flow), and mesh refinement.

Decoder does the job of producing final working finite-element nodes, element connectivity, node merge among parts, mesh clean up, contact, and setting up part, surface, node-based loadings with priority rules.

Processor phase: Processor phase is the kernel part of the finite-element application; it targets on solving real physical problems which are in discretized partial differential equation (PDE) forms and nonlinear formulations. The processor takes care of several levels of iterations, which includes time iteration for transient problems; nonlinear iteration in a single time step handles the coupling among multiphysics variables like velocity, pressure, temperature, turbulence, and multiphase; the iterative solver for linear equations requires iteration for the converged solution.

In the inner level, element matrices are built up via shape function of the specific element type and system matrix is assembled from the element-level matrix and then eventually global linear algebraic equations are solved to produce the primary result.

The element-based and node-based results are the output from the processor; typical variables are velocity, pressure, temperature, turbulence kinetic energy and dissipation rate, phase fraction, etc.

The processor phase is usually the biggest user of system memory and CPU time for finite-element-based simulation.

Postprocessor phase: Postprocessor also refers to result environment, which provides interactive graphics environment to visually interpret the node- and element-based simulation results from processor phase. Some typical operation includes slice plane, streamline, isoline and isosurface, make animation, annotation, derived date calculation, inquire data, curve plot and report production.

5.2 Finite-Element Description

A finite-element simulation program should select particular elements to form the base element library. Elements for fluid flow and heat transfer analyses are usually categorized by the combination of velocity and pressure approximations used in the element.

In finite element-based fluid flow and thermal problems, which is basically advection-diffusion analysis, following elements are used for two-dimensional and three-dimensional models.

5.2.1 Two-dimensional elements

Quadrilateral element

The four-node quadrilateral element can be used to model either two-dimensional Cartesian or axisymmetric geometries. Each node has four degrees of freedom for

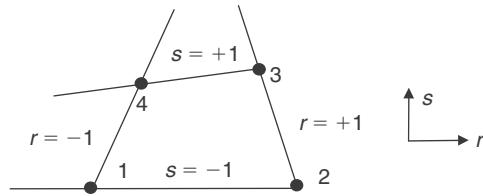


Figure 5.1. Two-dimensional quadrilateral element.

laminar flow: U, V, P, T and six degrees of freedom for turbulent flows: U, V, P, T, K, ω . Quadrilateral element should have nine nodes if midside nodes are used (Figure 5.1).

Take this element as an example for using shape function and Jacobian matrix, the velocity component u_i and temperature T are approximated by using bilinear interpolation function,

$$\varphi = \vartheta = \begin{bmatrix} \frac{1}{4}(1-r)(1-s) \\ \frac{1}{4}(1+r)(1-s) \\ \frac{1}{4}(1+r)(1+s) \\ \frac{1}{4}(1-r)(1+s) \end{bmatrix} \quad (1)$$

Two pressure discretizations are possible in this kind of element: a bilinear continuous approximation, $\psi \in Q_1$, with the pressure degrees of freedom located at the four corner nodes, or a piecewise constant discontinuous pressure approximation, $\psi = 1 \in Q_0$, with the pressure degrees of freedom associated with element centroid.

The construction of the finite-element matrices requires the computation of various derivatives and integrals of the bilinear interpolation function. Since the basis functions are given in terms of the normalized coordinates r and s and the derivatives and integrals are in terms of the physical x, y coordinates, the following relations need to be defined:

$$\begin{bmatrix} \frac{\partial \varphi}{\partial r} \\ \frac{\partial \varphi}{\partial s} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} \\ \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \end{bmatrix} \begin{bmatrix} \frac{\partial \varphi}{\partial x} \\ \frac{\partial \varphi}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial N^T}{\partial r} x & \frac{\partial N^T}{\partial s} y \\ \frac{\partial N^T}{\partial s} x & \frac{\partial N^T}{\partial r} y \end{bmatrix} \begin{bmatrix} \frac{\partial \varphi}{\partial x} \\ \frac{\partial \varphi}{\partial y} \end{bmatrix} = J \begin{bmatrix} \frac{\partial \varphi}{\partial x} \\ \frac{\partial \varphi}{\partial y} \end{bmatrix} \quad (2)$$

where \mathbf{J} is the Jacobian matrix; by inverting it, equation (2) provides the necessary relation for the derivatives of the basis function as below:

$$\begin{bmatrix} \frac{\partial \varphi}{\partial x} \\ \frac{\partial \varphi}{\partial y} \end{bmatrix} = J^{-1} \begin{bmatrix} \frac{\partial \varphi}{\partial r} \\ \frac{\partial \varphi}{\partial s} \end{bmatrix} \quad (3)$$

In the case of integral evaluation, to complete the transformation from physical coordinate to normalized coordinates, an elemental area for quadrilateral element could be evaluated as follows:

$$dx dy = |J| dr ds \quad (4)$$

where $|J| = \text{Determine of } J$.

Equations (3) and (4) allow any integral of functions of x and y to be expressed as integrals of rational functions in the r and s coordinate system. This is extremely important in evaluating element matrices, which is the basic step in finite-element method.

The details for other elements are ignored in this chapter.

Triangular element

The three-node triangular element can be used to model either two-dimensional Cartesian or axisymmetric geometries. Each node has four degrees of freedom for laminar flow: U, V, P, T and six degrees of freedom for turbulent flows: U, V, P, T, K, ω .

Triangular element should have six nodes if midside nodes are used.

5.2.2 Three-dimensional elements

Brick (hexahedral) element

The eight-node brick element is used to model three-dimensional geometries as the general base. Each node has five degrees of freedom for laminar flow, i.e., V, W, P, T and six degrees of freedom for turbulent flows, i.e., V, W, P, T, K, ω .

Brick element has total 21 nodes if midside nodes are used; this is also used as the base element for other degenerated elements with midside nodes.

Tetrahedral element

The four-node tetrahedral element can be used to model three-dimensional geometries as a degenerated form from three-dimensional brick element; the mapping relation is listed in Table 5.1. Each node has five degrees of freedom for laminar flow, i.e., U, V, W, P, T and seven degrees of freedom for turbulent flows, i.e., U, V, W, P, T, K, ω .

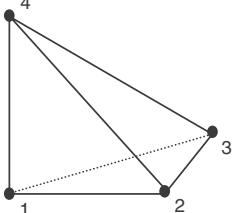
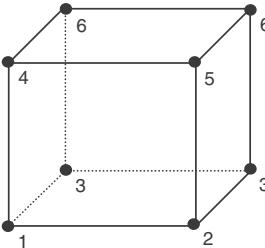
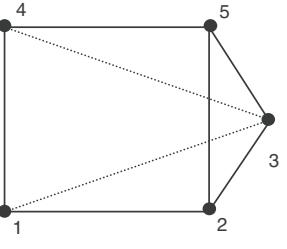
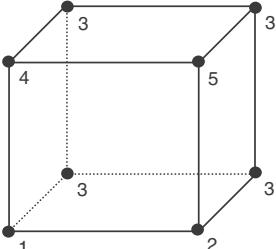
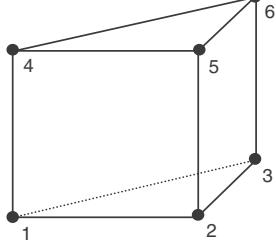
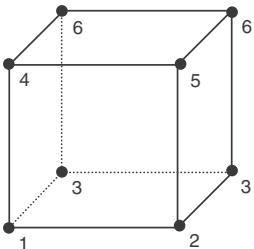
Tetrahedral element has 10 nodes if midside nodes are used.

Pyramid element

The five-node pyramid element can be used to model three-dimensional geometries as a degenerated form from three-dimensional brick element; the mapping relation is listed in Table 5.1. Each node has five degrees of freedom for laminar flow, i.e., U, V, W, P, T and seven degrees of freedom for turbulent flows, i.e., U, V, W, P, T, K, ω .

Pyramid element has 13 nodes if midside nodes are used.

Table 5.1. Mapping from degenerated three-dimensional elements to computational nodal configuration

Degenerated elements	Computational nodal configuration																
3D tetrahedral element	 <table border="1"> <tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>3</td><td>4</td><td>4</td><td>3</td><td>3</td></tr> </table> 	1	2	3	4	5	6	7	8	1	2	3	3	4	4	3	3
1	2	3	4	5	6	7	8										
1	2	3	3	4	4	3	3										
3D pyramid element	 <table border="1"> <tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>3</td><td>4</td><td>5</td><td>5</td><td>3</td></tr> </table> 	1	2	3	4	5	6	7	8	1	2	3	3	4	5	5	3
1	2	3	4	5	6	7	8										
1	2	3	3	4	5	5	3										
3D wedge element	 <table border="1"> <tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>3</td><td>4</td><td>5</td><td>3</td><td>3</td></tr> </table> 	1	2	3	4	5	6	7	8	1	2	3	3	4	5	3	3
1	2	3	4	5	6	7	8										
1	2	3	3	4	5	3	3										

Wedge element

The six-node wedge (prism) element can be used to model three-dimensional geometries as a degenerated form from three-dimensional brick element; the mapping relation is listed in Table 5.1. Each node has five degrees of freedom for laminar

flow, i.e., U, V, W, P, T and seven degrees of freedom for turbulent flows, i.e., U, V, W, P, T, K, .

Wedge element has 15 nodes if midside nodes are used.

5.2.3 Degenerated elements

In general finite-element code development, the three-dimensional six-node wedges, five-node pyramids, and four-node tetrahedral are considered as degenerate form of eight-node brick element by collapsing nodes. Table 5.1 attempts to demonstrate how degenerate elements are formed by collapsing nodes if midside nodes are not used. The first row is node ID in brick element and the second row is for degenerated nodes; the two rows make a mapping between degenerated element nodes and brick element nodes. The high-order elements with midside nodes are not commonly used for finite element-based fluid flow and heat transfer codes, and this is discussed in later sections of this chapter.

In two-dimensional case, the three-node triangle element is degenerated from four-node quadrilateral element in a similar manner.

5.2.4 Special elements (rod and shell)

In heat transfer of some finite-element codes, arbitrary thickness rod and shell elements may be defined as those which allow heat conduction along the rod or in the shell of the element, but offer no resistance to heat flow across the element.

Rod elements have only length and may be used in either two-dimensional or three-dimensional simulations. Shell elements have both length and width and may only be used in three-dimensional simulations.

5.3 Governing Equations for Fluid Flow and Heat Transfer Problems

The numerical modeling process begins with a physical model that is based on a number of simplifying physical assumptions. These assumptions can be made in light of the understanding of the physical processes that are involved. A mathematical model is built from the physical model and then a series of PDEs are developed.

For most of the practical physical processes, these PDEs cannot be solved analytically but must be solved numerically. Analytic solutions have been found only for a small subset of all possible fluid flow problems. Nonlinear nature of the governing equations can produce extremely complex flow fields and multiple time-dependent solutions, even for simplest geometries. Normally, the nature of the mathematical model would dictate the type of numerical model employed and a review of the current literature would suggest the most appropriate numerical model to use. A combination of the mathematical model and numerical model leads to a set of governing equations. Before these governing equations can be solved,

a solution algorithm should be developed that can solve the equations correctly and efficiently.

In this section, the governing equations and auxiliary equations are focused upon for the fluid flow and heat transfer phenomena and dynamic moving boundary problems. The details are discussed as following aspects:

- (a) Formation of the set of nonlinear coupled PDEs for mass and momentum balances and various types of boundary conditions.
- (b) Handling method for four types of terms in the governing equations, including transient, advection, diffusion, and source.
- (c) Uniform stabilized GLS formulation, including SUPG method and PSPG method, which demonstrates viable equal-order velocity and pressure interpolations.

5.3.1 General form of governing equations

In this section, the strong form of the fluid flow problem is presented using a linear constitutive relationship, leading to a set of nonlinear coupled PDEs for mass and momentum balances. Various types of boundary conditions are also introduced.

Let Ω and $(0,T)$ denote the spatial and temporal domains, with x and t representing the coordinates associated with Ω and $(0,T)$. The boundary Γ of the domain Ω may involve several internal boundaries.

Conservation of mass

In solving fluid flow problems, it is important that the mass is conserved in the system. The continuity equation may be written as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{V}) = 0 \quad (5)$$

where the term $\partial \rho / \partial t$ stands for the change in density over time; it also describes the shrinkage-induced flow in heat transfer solidification.

Conservation of momentum

The differential equation governing the conservation of momentum in a given direction for a Newtonian fluid can be written as

$$\frac{\partial(\rho \vec{V})}{\partial t} + \nabla \cdot (\rho \vec{V} \vec{V}) = \nabla \cdot (\mu \nabla \vec{V}) - \nabla P + \vec{B} + S_m \quad (6)$$

where μ is the dynamic viscosity, P is the pressure, \vec{B} is the body force per unit volume, and S_m is momentum source term.

The component form of equation (6) in 3D is

$$\rho \left(\frac{\partial u_1}{\partial t} + u_1 \frac{\partial u_1}{\partial x_1} + u_2 \frac{\partial u_1}{\partial x_2} + u_3 \frac{\partial u_1}{\partial x_3} \right) - \mu \left(\frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial^2 u_1}{\partial x_2^2} + \frac{\partial^2 u_1}{\partial x_3^2} \right) + \frac{\partial p}{\partial x_1} - f_1 = 0 \quad (7)$$

$$\rho \left(\frac{\partial u_2}{\partial t} + u_1 \frac{\partial u_2}{\partial x_1} + u_2 \frac{\partial u_2}{\partial x_2} + u_3 \frac{\partial u_2}{\partial x_3} \right) - \mu \left(\frac{\partial^2 u_2}{\partial x_1^2} + \frac{\partial^2 u_2}{\partial x_2^2} + \frac{\partial^2 u_2}{\partial x_3^2} \right) + \frac{\partial p}{\partial x_2} - f_2 = 0 \quad (8)$$

$$\rho \left(\frac{\partial u_3}{\partial t} + u_1 \frac{\partial u_3}{\partial x_1} + u_2 \frac{\partial u_3}{\partial x_2} + u_3 \frac{\partial u_3}{\partial x_3} \right) - \mu \left(\frac{\partial^2 u_3}{\partial x_1^2} + \frac{\partial^2 u_3}{\partial x_2^2} + \frac{\partial^2 u_3}{\partial x_3^2} \right) + \frac{\partial p}{\partial x_3} - f_3 = 0 \quad (9)$$

If the flow is turbulent, the time-averaged N-S equations have the same form as equation (6), except that the fluid viscosity is replaced by an effective viscosity, μ_{eff} , which is defined as

$$\mu_{\text{eff}} = \mu + \mu_t \quad (10)$$

Here μ is still the fluid dynamic viscosity while μ_t is the turbulence viscosity.

Conservation of energy

The equation governing the conservation of energy in terms of enthalpy can be written in general as

$$\frac{\partial(\rho H)}{\partial t} + \nabla \cdot (\rho \vec{V} H) = \nabla \cdot (k \nabla T) + S_h \quad (11)$$

where H is the specific enthalpy, k is the thermal conductivity, T is the temperature, and S_h is the volumetric rate of heat generation. The first term on the right-hand side of equation (11) represents the influence of conduction heat transfer within the fluid, according to the Fourier's law of conduction.

The energy equation could also be written in terms of static temperature; however, the enthalpy form of energy equation generally considers the multiphase flow such as steam/water, moist gas flow, and phase change like solidification and melting.

Transport equations

The general transport equation for advection–diffusion is described as

$$\frac{\partial}{\partial t}(\rho\phi) + \nabla(\rho\vec{V} \cdot \phi) = \nabla(\rho\Gamma \cdot \nabla\phi) + S \quad (12)$$

Here, ϕ is the phase quantity in general expression, Γ is the diffusion coefficient in general expression, the first left-hand term is the temporal term, and the second left-hand term is the convective term. The first right-hand term is the diffusive term and the second right-hand term is the source term.

Boundary conditions

Different types of boundary conditions can be encountered in fluid flow problems. The simplest one could be the Dirichlet boundary condition, which is given by

$$\mathbf{u} = \mathbf{u}_g \quad (13)$$

Another boundary condition is traction-free boundary condition, which is often applied for the far downstream boundary of the flow domain and is given by

$$\sigma \cdot \mathbf{n} = 0 \quad (14)$$

For some special cases, traction force can also be applied as boundary conditions.

For solid boundaries or symmetric planes, the normal component of velocity must be specified as “no penetration” condition, i.e.,

$$\mathbf{u} \cdot \mathbf{n} = u_n = 0 \quad (15)$$

The shear stress at the symmetric planes must be specified as a natural boundary condition as follows:

$$\mu \frac{\partial u_\tau}{\partial n} = 0 \quad (16)$$

To be more specific, for a Newtonian fluid and a straight edge, equation (16) can be rewritten in terms of its normal and tangential components:

$$\sigma_n = (\mathbf{n} \cdot \boldsymbol{\sigma}) \cdot \mathbf{n} = -p + 2\mu \frac{\partial u_n}{\partial n} \quad (17)$$

$$\sigma_\tau = (\mathbf{n} \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{\tau} = \mu \left(\frac{\partial u_\tau}{\partial n} + \frac{\partial u_n}{\partial \tau} \right) \quad (18)$$

If the symmetry line is aligned with a Cartesian axis, equation (15) becomes a Dirichlet boundary condition while equation (16) is satisfied by setting $\sigma_\tau = 0$. Thus, a symmetry line involves both Dirichlet and Neumann type of boundaries.

5.3.2 Discretized equations and solution algorithm

This section is used to reduce the governing PDEs to a set of algebraic equations. In this method, the dependent variables are represented by polynomial shape functions over a small area or volume (element in finite-element method). These representations are substituted into the governing PDEs and then the weighted integral of these equations over the element is taken where the weight function is chosen to be the same as the shape function. The result is a set of algebraic equations for the dependent variable at discrete points or nodes on every element.

General form of discretization equations

The momentum, energy, temperature, and turbulent equations all have the form of a scalar transport equation. There are four types of terms in the equation, transient, advection, diffusion, and source. For the purpose of describing the discretization

Table 5.2. General meaning of transportation representation

ϕ	Meaning	C_ϕ	Γ_ϕ	S_ϕ
u	X velocity	1	μ_e	$\rho g_x - \frac{\partial p}{\partial x} + Rx$
v	X velocity	1	μ_e	$\rho g_x - \frac{\partial p}{\partial x} + Rx$
w	X velocity	1	μ_e	$\rho g_x - \frac{\partial p}{\partial x} + Rx$
T	Temperature	C_p	K	$Q_v + E^k + W^v + \mu\Phi + \partial p/\partial t$
k	Turbulence kinetic energy	1	μ_t/σ_k	$\mu_t\Phi/\mu - \rho\varepsilon + C_4\beta g_i \left(\frac{\partial T}{\partial x_i} \right) / \sigma_t$
ε	Turbulence dissipation rate	1	μ_t/σ_ε	$C_1\mu_t\varepsilon\Phi/k - C_2\rho\varepsilon^2/k + C_1C_\mu C_3\beta k g_i \left(\frac{\partial T}{\partial x_i} \right) / \sigma_t$

methods, the variable is referred to as ϕ . Then the form of the scalar transport equation is

$$\begin{aligned} & \frac{\partial}{\partial t}(\rho C_\phi \phi) + \frac{\partial}{\partial t}(\rho u C_\phi \phi) + \frac{\partial}{\partial t}(\rho v C_\phi \phi) \frac{\partial}{\partial t}(\rho w C_\phi \phi) \\ &= \frac{\partial}{\partial x} \left(\Gamma_\phi \frac{\partial \phi}{\partial x} \right) + \frac{\partial}{\partial y} \left(\Gamma_\phi \frac{\partial \phi}{\partial y} \right) + \frac{\partial}{\partial z} \left(\Gamma_\phi \frac{\partial \phi}{\partial z} \right) + S_\phi \end{aligned} \quad (19)$$

where

C_ϕ is the transient and advection coefficient, Γ_ϕ is the diffusion coefficient, and S_ϕ are source terms.

The pressure equation can be derived from continuity equation by segregated solution scheme, which is discussed in the next chapter. Equation (19) is a general form of velocity, temperature, turbulent variables k , ε , and species. Table 5.2 lists the corresponding equation representation for C_ϕ , Γ_ϕ , and S_ϕ .

The discretization process, therefore, consists of deriving the element matrices to put together the matrix equation in simplified matrix form as follows:

$$([A_e^{\text{Transient}}] + [A_e^{\text{Advection}}] + [A_e^{\text{Diffusion}}])\{\phi_e\} = \{S_e^\phi\} \quad (20)$$

Galerkin method of weighted residuals is used to form the element integrals, denoted by N^e the weighting function for the element, which is also the shape function.

Temporal term

For transient analyses, the transient terms are discretized using an implicit or backward difference method. Using the matrix algebra notation, a typical steady-state scalar transport equation with momentum, energy, and turbulence variables can be written:

$$A_{ij}u_j = F_i$$

where A_{ij} contains the discretized advection and diffusion terms from the governing equations, u_j is the solution vector or values of the dependent variable (u, v, w, T, K, \dots) and F_j contains the source terms.

The transient terms in the governing equations took the form $\rho\partial\phi/\partial t$, where ϕ represents the dependent variable (u, v, w, \dots). This term is discretized using a backward difference method:

$$\frac{\partial\phi}{\partial t} = \frac{\phi^{\text{new}} - \phi^{\text{old}}}{\Delta t} \quad (21)$$

Add this term to the matrix equation above

$$(A_{ij} + B_{ij})u_j^{\text{new}} = F_i + B_{ii}u_j^{\text{old}} \quad (22)$$

where B_{ij} is a diagonal matrix composed of terms like:

$$B_{ii} = \frac{1}{\Delta t} \int N_i \rho d\Omega_e \quad (23)$$

This discretized transient equation must be solved iteratively at each time step to determine all of the new variables (variable values at the latest time).

Advection term

There are two approaches to discretize this most challengeable term in fluid flow: the monotone streamline-upwind (MSU) approach is the first-order accurate and produces smooth and monotone solutions, and the SUPG approach is the second-order accurate.

Monotone streamline upwind (MSU) approach. In the MSU method, the advection term is handled through a monotone streamline approach based on the idea that advection transport is along characteristic lines. It is useful to think of the advection transport formulation in terms of a quantity being transported in a known velocity field. The velocity field itself could be envisioned as a set of streamlines everywhere tangent to the velocity vectors. The advection terms can be expressed in terms of the streamline velocities.

In advection-dominated transport phenomena, one assumes that no transport occurs across characteristic lines, i.e., all transfer occurs along streamlines.

Therefore, if one assumes that the advection term is expressed in streamline direction:

$$\frac{\partial(\rho c_\phi v_x \phi)}{\partial x} + \frac{\partial(\rho c_\phi v_y \phi)}{\partial y} + \frac{\partial(\rho c_\phi v_z \phi)}{\partial z} = \frac{\partial(\rho c_\phi v_s \phi)}{\partial s} \quad (24)$$

then the advection term is constant throughout an element:

$$[A_e^{\text{advection}}] = \frac{\partial(\rho c_\phi v_s \phi)}{\partial s} \int N d\Omega_e \quad (25)$$

Figure 5.2 illustrates the streamline across a brick element; this formulation could be used for every element, each of which will have one node which gets contributions from inside the element; the derivative is calculated using a simple difference as follows:

$$\frac{\partial(\rho c_\phi v_s \phi)}{\partial s} = \frac{(\rho c_\phi v_s \phi)_{\text{up}} - (\rho c_\phi v_s \phi)_{\text{down}}}{\Delta s} \quad (26)$$

where

down = Subscript for value at downstream node

up = Subscript for value taken at the location at which the streamline through the streamline through the upwind node enters the element. The value is interpolated in terms of the nodes in between.

Δs = Distance from the upstream point to the downstream node.

The whole process goes through all the elements and identifies the downwind nodes. The calculation is made based on the velocities to find where the streamline through the downwind node came from. Weight factors are calculated based on the proximity of the upwind location to the neighboring nodes.

Streamline-upwind/Petrov–Galerkin (SUPG) approach. This method is used to discretize the advection term and an additional diffusion-like perturbation term, which acts only in the advection direction.

For three-dimensional problem, the perturbed interpolation function is given by

$$W_i = N_i + \left(\frac{\beta \bar{h}}{2|\vec{V}|} \right) \left(u \frac{\partial N_i}{\partial x} + v \frac{\partial N_i}{\partial y} + w \frac{\partial N_i}{\partial z} \right) \quad (27)$$

Here, $\vec{V} = (u, v, w)$ is the velocity field in three-dimensional domain, \bar{h} is the averaged size of an element, which, for eight-node brick element shown in Figure 5.2, can be calculated as

$$\bar{h} = \frac{1}{|\vec{V}|} (|h_1| + |h_2| + |h_3|) \quad (28)$$

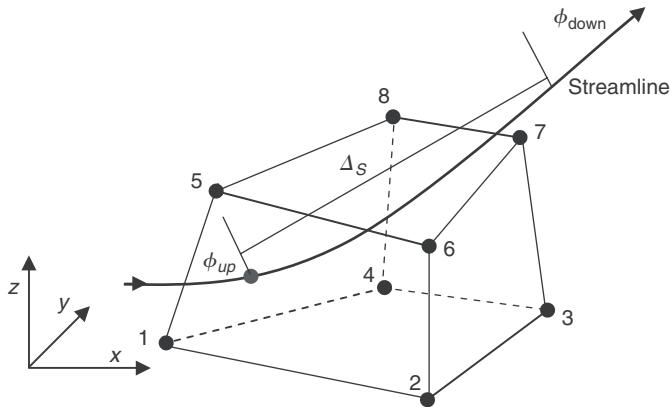


Figure 5.2. Streamline upwind approach for 3D brick element.

where

$$h_1 = \vec{a} \cdot \vec{V}, \quad h_2 = \vec{b} \cdot \vec{V}, \quad h_3 = \vec{c} \cdot \vec{V} \quad (29)$$

Taking an example of a eight-node brick element, the element direction vectors, $\vec{a} = (a_x, a_y, a_z)$, $\vec{b} = (b_x, b_y, b_z)$, and $\vec{c} = (c_x, c_y, c_z)$ are calculated using the brick element nodal coordinates as shown in Figure 5.2.

$$\begin{aligned} a_x &= \frac{1}{2}(x_2 + x_3 + x_6 + x_7 - x_1 - x_4 - x_5 - x_8) \\ a_y &= \frac{1}{2}(y_2 + y_3 + y_6 + y_7 - y_1 - y_4 - y_5 - y_8) \\ a_z &= \frac{1}{2}(z_2 + z_3 + z_6 + z_7 - z_1 - z_4 - z_5 - z_8) \\ b_x &= \frac{1}{2}(x_5 + x_6 + x_7 + x_8 - x_1 - x_2 - x_3 - x_4) \\ b_y &= \frac{1}{2}(y_5 + y_6 + y_7 + y_8 - y_1 - y_2 - y_3 - y_4) \\ b_z &= \frac{1}{2}(z_5 + z_6 + z_7 + z_8 - z_1 - z_2 - z_3 - z_4) \\ c_x &= \frac{1}{2}(x_1 + x_2 + x_5 + x_6 - x_3 - x_4 - x_7 - x_8) \\ c_y &= \frac{1}{2}(y_1 + y_2 + y_5 + y_6 - y_3 - y_4 - y_7 - y_8) \\ c_z &= \frac{1}{2}(z_1 + z_2 + z_5 + z_6 - z_3 - z_4 - z_7 - z_8) \end{aligned}$$

Diffusion term

The diffusion term is obtained by integration over the problem domain after it is multiplied by the weighting function.

$$[A_e^{\text{diffusion}}] = \int N \frac{\partial}{\partial x} \left(\Gamma_\phi \frac{\partial \phi}{\partial x} \right) d\Omega_e + \int N \frac{\partial}{\partial y} \left(\Gamma_\phi \frac{\partial \phi}{\partial y} \right) d\Omega_e + \int N \frac{\partial}{\partial z} \left(\Gamma_\phi \frac{\partial \phi}{\partial z} \right) d\Omega_e \quad (30)$$

Once the derivative of ϕ is replaced by the nodal values and the derivatives of the weighting function, the nodal values will be removed from the integrals, $\frac{\partial \phi}{\partial x} = \frac{\partial N_i}{\partial x} \phi$, for momentum equations, the diffusion terms in x, y, z direction are treated in similar fashion, a typical diffusion item in momentum equation will be $K_{ij} = \int_{\Omega_e} \mu \frac{\partial N_m}{\partial x_j} \frac{\partial N_n}{\partial x_i} d\Omega_e$.

Source term

Source term consists of merely multiplying the source terms by the weighting function and integrating over the volume as follows:

$$S_\phi^e = \int N_i S_\phi d\Omega_e \quad (31)$$

5.3.3 Stabilized method

The difficulties mostly encountered in finite-element method are from the mesh that is generated by general automatic mesh engine. Given a three-dimensional complicated geometry, which is usually from CAD package, e.g., the amount of elements for automatic generated purely brick could be huge and makes the model analysis impractical. So hybrid meshes are practically used, but this approach involves lot of nonbrick elements such as tetrahedron, wedge, and pyramid. This also happens in two-dimensional hybrid element with quadrilateral and triangle elements.

However, since all the two-dimensional and three-dimensional degenerated elements do not satisfy the LBB condition, the fluid flow velocity and pressure solution gets locked. The LBB condition restricts the type of element that can be used for the Galerkin FEM formulation. Both two-dimensional linear triangular element and three-dimensional linear tetrahedral, wedge, and pyramid element do not satisfy this condition.

Approaches to handle LBB condition

There are two approaches to overcome the above difficulties: one is to use high-order elements and the other to use stabilized FEM based on mixed formulation.

Approach one: high-order elements. High-order elements use midside nodes to perform quadratic interpolation over elements, but the big disadvantage is that the total unknowns will increase greatly due to the extra mid-nodes at the element edges, making the model size limitation to be more restricted – for this reason, the high-order element is rarely used in commercial finite-element codes.

Approach two: stabilized mixed FEM formulation. Although the mixed formulation requires the elements to be LBB type, there are lot of research work and efforts to develop new techniques that circumvent this restriction. These techniques include bubble function method, two-level mesh method, and the GLS formulations. These techniques allow the usage of those types of elements that do not satisfy the LBB condition, such as two-dimensional triangular element (linear velocity/linear pressure) and three-dimensional tetrahedron, wedge, and pyramid element (linear velocity/linear pressure interpolation); this is also referred to as equal-order formulation.

Step 1: Taking three-dimensional incompressible flow as an example for the mixed FEM-based fluid flow formulation:

$$\begin{aligned} \left\{ \begin{array}{l} \mathbf{F}_1 \\ \mathbf{F}_2 \\ \mathbf{F}_3 \\ \mathbf{0} \end{array} \right\} &= \left[\begin{array}{cccc} \mathbf{M} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array} \right] \left\{ \begin{array}{l} \dot{\mathbf{u}}_1 \\ \dot{\mathbf{u}}_2 \\ \dot{\mathbf{u}}_3 \\ \dot{\mathbf{P}} \end{array} \right\} + \left[\begin{array}{cccc} \mathbf{C}(\mathbf{u}) & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}(\mathbf{u}) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{C}(\mathbf{u}) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array} \right] \left\{ \begin{array}{l} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \\ \mathbf{P} \end{array} \right\} \\ &+ \left[\begin{array}{cccc} 2\mathbf{K}_{11} + \mathbf{K}_{22} + \mathbf{K}_{33} & \mathbf{K}_{12} & \mathbf{K}_{13} & -\mathbf{Q}_1 \\ \mathbf{K}_{21} & \mathbf{K}_{11} + 2\mathbf{K}_{22} + \mathbf{K}_{33} & \mathbf{K}_{23} & -\mathbf{Q}_2 \\ \mathbf{K}_{31} & \mathbf{K}_{32} & \mathbf{K}_{11} + \mathbf{K}_{22} + 2\mathbf{K}_{33} & -\mathbf{Q}_3 \\ -\mathbf{Q}_1^T & -\mathbf{Q}_2^T & -\mathbf{Q}_3^T & \mathbf{0} \end{array} \right] \left\{ \begin{array}{l} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \\ \mathbf{P} \end{array} \right\} \end{aligned} \quad (32)$$

A more compact form of this equation is given as

$$\left[\begin{array}{cc} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{array} \right] \left\{ \begin{array}{l} \dot{\mathbf{u}} \\ \dot{\mathbf{P}} \end{array} \right\} + \left[\begin{array}{cc} \mathbf{C}(\mathbf{u}) + \mathbf{K}(\mathbf{u}) & \mathbf{Q}^T \\ -\mathbf{Q}^T & \mathbf{0} \end{array} \right] \left\{ \begin{array}{l} \mathbf{u} \\ \mathbf{P} \end{array} \right\} = \left\{ \begin{array}{l} \mathbf{F} \\ \mathbf{0} \end{array} \right\} \quad (33)$$

or in a more symbolic format as

$$\bar{\mathbf{M}}\dot{\mathbf{U}} + \bar{\mathbf{K}}\mathbf{U} = \bar{\mathbf{F}} \quad (34)$$

The nature of the matrix is sparse, generally unsymmetric, and positive semidefinite. The zero entries on the diagonal of the matrix lead to difficulties for the solution of the system of algebraic equations. Direct solver needs to have pivoting technique.

Step 2: Implementation of equal-order elements for the mixed FEM-based code first, in terms of the implementation of so-called bubble function to stabilize the scheme.

In this approach, the basic formulation is still the Galerkin FEM, but velocity interpolation is carried out with one additional DOF, which is the centroid of triangular element (2D) and tetrahedral element (3D). The bubble function for the central node velocity can be chosen as a piecewise linear, with its value being zero at the element boundaries and piecewise linear on two-dimensional subtriangles

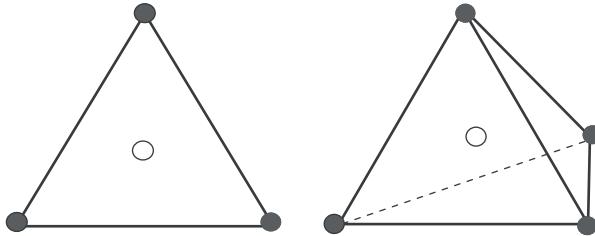


Figure 5.3. Bubble function for linear triangular and tetrahedral elements.

or three-dimensional subtetrahedron. The pressure is interpolated linearly over the element (Figure 5.3).

Step 3: Implementation of stabilized FEM formulation GLS for two-dimensional and three-dimensional fluid flow using two-dimensional and three-dimensional degenerated elements, i.e., using equal-order interpolation (linear velocity/linear pressure) for both velocity and pressure.

The GLS variation form of the momentum and continuity equation is given by

$$\begin{aligned} \int_{\Omega} \mathbf{w} \cdot (\rho_0 \mathbf{u} \cdot \nabla \mathbf{u}) d\mathbf{x} + \int_{\Omega} \nabla \mathbf{w} \cdot (-P + 2\mu \mathbf{D}(\mathbf{u})) d\mathbf{x} - \int_{\Omega} \rho_0 \mathbf{w} \mathbf{f} d\mathbf{x} + Q \nabla \cdot \mathbf{u} d\mathbf{x} \\ + \sum_{n=1}^{nel} \int_{\Omega_n} R_{GLS} dx = - \int_{\Gamma} \mathbf{w} T_i ds \end{aligned} \quad (35)$$

Here, w and Q are the weight functions for the momentum equation and the continuity equation, respectively. The element residual is defined as

$$\begin{aligned} R_{GLS} = & (\tau_{SUPG} \rho_0 u \cdot \nabla w + \tau_{PSPG} \nabla Q - \tau_{GLS} 2\mu \mathbf{D}(\mathbf{w})) \\ & \cdot (\rho_0 \mathbf{u} \cdot \nabla \mathbf{u} + \nabla P - 2\mu \mathbf{D}(\mathbf{u})) \end{aligned} \quad (36)$$

The τ coefficients are weighting parameters, which could be functions of the element size, element Reynolds number, and/or element average velocity. If these coefficients are selected properly, equal-order elements can be used and LBB condition is not necessary.

For momentum equation, the final equations are given by

$$\begin{aligned} & (\mathbf{C}(\mathbf{u}) + \mathbf{C}_{SUPG}(\mathbf{u}) + \mathbf{C}_{GLS}(\mathbf{u})) \mathbf{u} + (\mathbf{K} + \mathbf{K}_{SUPG} + \mathbf{K}_{GLS}) \mathbf{u} \\ & - (\mathbf{Q} + \mathbf{Q}_{SUPG} + \mathbf{Q}_{GLS}) \mathbf{P} = \mathbf{F} \end{aligned} \quad (37)$$

For continuity equation, we have

$$-\mathbf{Q}^T \mathbf{u} - \mathbf{Q}_{PSPG} \mathbf{P} + \mathbf{C}_{PSPG}(\mathbf{u}) \mathbf{u} + \mathbf{K}_{PSPG} \mathbf{u} = \mathbf{0} \quad (38)$$

Galerkin/least-squares formulation

The stability of the mixed Galerkin finite element for incompressible flows can be enhanced by the addition of various least-squares terms to the original Galerkin variational statement. The Galerkin/Least-squares approach is sometimes termed a residual method because the added least-squares terms are weighted residuals of the momentum equation; this form of the least-squares term implies the consistency of the method since the momentum residual is employed. The GLS – a Petrov–Galerkin method – is also known as a perturbation method since the added terms can be viewed as perturbations to the weighting functions. Development and popularization of the GLS methods for flow problems are primarily due to Hughes, Tezduyar, and co-workers and are a generalization of their work on SUPG and PSPG methods.

The usual approach to the GLS begins with the discontinuous approximation in time Galerkin method and considers a finite-element approximation for a space–time slab. Continuous polynomial interpolation is used for the spatial variation while a discontinuous time function is used within the space–time slab. The assumed time representation obviates the need to independently consider time integration methods. To simplify the present discussion of the GLS, the time independent form of the incompressible flow problem is considered which avoids the complexity of the space–time finite-element formulation. Using vector notation and following the weak form development, the GLS variational form for the momentum and continuity equations can be written as

$$\int_{\Omega} N_m(\rho u \cdot \nabla u) dx + \int_{\Omega} \nabla N_m \cdot (-P + 2\mu \nabla D(u)) dx - \int_{\Omega} \rho N_m f_x dx + \int_{\Omega} N_n \nabla \cdot u) dx \\ + \sum_{n=1}^{nel} \int_{\Omega} R_{GLS} = - \int_{\Gamma} N_m \tau_i ds dx \quad (39)$$

where N_m and N_n are the weight functions for the momentum and continuity equations and the element residual is defined by

$$R_{GLS} = (\delta + \varepsilon + \beta) \cdot (\rho u \cdot \nabla u + \nabla(P - 2\mu \nabla D(u))) \quad (40)$$

$$\delta = \tau_{SUPG} u \cdot \nabla N_m$$

$$\varepsilon = \tau_{PSPG} \frac{1}{\rho} \nabla N_n$$

$$\beta = -\tau_{GLS} 2\mu \nabla(D(N_m))$$

The τ coefficients are weighting parameters; if the definitions for δ , ε , and β are substituted into equation (40) and the various weighting parameters are made equivalent to a single parameter τ , the R_{GLS} will be

$$R_{GLS} = \tau [\rho u \cdot \nabla N_m + \nabla N_n - 2\mu \nabla \cdot D(u)] \cdot [\rho u \cdot \nabla u + \nabla(P - 2\mu \nabla \cdot D(u))] \quad (41)$$

This is the standard definition for the residual contribution to the GLS formulation. The splitting of the first residual into three separate contributions in equation (41) was done to allow the original SUPG and PSPG formulations to be easily recovered.

If β and ε are set to zero, the stabilized SUPG method is recovered.

If β and δ are set to zero, the stabilized PSPG method is recovered.

If only β is set to zero, both SUPG and PSPG stabilization methods are recovered.

Note that in equation (39) the first four integrals and the right-hand side define a standard Galerkin-weighted residual method that is written in terms of global shape functions; the integrals are over the simulation domain which is composed by the discretized finite elements. The added residual term in equation (39) is defined over the interior of each element and basically contains the square of all or parts of the momentum residual. The various τ parameters are positive coefficients that have the dimension of time. The forms of these parameters are usually developed from error estimates, convergence proofs, and dimensional analysis. Particular constants within each parameter are selected by optimizing the method on simple problems and generalizing to multidimensions. It should be noted that the development of the τ parameters is not a unique process. For the steady problems considered here a typical τ for the GLS method is

$$\tau = \left[\left(\frac{2\|u\|}{h} \right)^2 + \left(\frac{4\nu}{h^2} \right)^2 \right]^{-1/2} \quad (42)$$

where h is an appropriate element length. For the SUPG and PSPG formulations, the τ values are the function of an element Reynolds number and the ratio of an element length to a velocity scale.

The error and convergence analyses for the GLS and PSPG methods have demonstrated that the equal-order velocity and pressure interpolation for these methods is viable and the LBB condition is not required [6]. In finite-element implementation, convenient elements, such as the bilinear Q1/Q1, linear P1/P1, biquadratic Q2/Q2, and quadratic P2/P2, are now useful approximations while being unstable in the Galerkin finite-element context. The low-order elements have an advantage over their higher-order counterparts because some of the stabilization terms associated with viscous diffusion are identically zero. This considerably simplifies the element equation building process.

The matrix form of velocity-pressure equations

The matrix form of the stabilized GLS formulation could be summarized as the Momentum equation and Continuity equation, respectively.

Momentum: $(C(u) + C_\delta(u) + C_\beta(u))u + (K + K_\delta + K_\beta)u - (Q + Q_\delta + Q_\beta)P = F$

Continuity: $-Q^T u - Q_\varepsilon P + (C_\varepsilon(u) + K_\varepsilon)u = 0$

Expended discretization form of equations for velocity-pressure coupling and SUPG stabilization, which are summarized in Table 5.3, will be used as the base

Table 5.3. Element-level matrices

$M = \int_{\Omega_e} \rho N_m N_n d\Omega_e$	$C(\mathbf{u}) = \int_{\Omega_e} \rho N_m \bar{u}_j \frac{\partial N_n}{\partial x_j} d\Omega_e$
$Su(\mathbf{u}) = \int_{\Omega_e} \left[\rho \tau_s \bar{u}_i \frac{\partial N_m}{\partial x_i} \right] \bar{u}_j \frac{\partial N_n}{\partial x_j} d\Omega_e$	$Sp_i(\mathbf{u}) = \int_{\Omega_e} \left[\tau_s \bar{u}_j \frac{\partial N_m}{\partial x_j} \right] \frac{\partial N_n}{\partial x_i} d\Omega_e$
$K_{ij} = \int_{\Omega_e} \mu \frac{\partial N_m}{\partial x_j} \frac{\partial N_n}{\partial x_i} d\Omega_e$	$Q_i = \int_{\Omega_e} \frac{\partial N_m}{\partial x_i} N_n d\Omega_e$
$S\delta_{ij} = \int_{\Omega_e} \rho \delta \frac{\partial N_m}{\partial x_i} \frac{\partial N_n}{\partial x_j} d\Omega_e$	$K_{\tau_i}(\mathbf{u}) = \int_{\Omega_e} \tau_p \frac{\partial N_m}{\partial x_i} \left[\bar{u}_j \frac{\partial N_n}{\partial x_j} \right] d\Omega_e$
$Q_\tau = \int_{\Omega_e} \frac{\tau_p}{\rho} \left[\frac{\partial N_m}{\partial x_j} \frac{\partial N_n}{\partial x_j} \right] d\Omega_e$	$SF_i(\mathbf{u}) = \int_{\Omega_e} \rho \tau_s \left[\bar{u}_j \frac{\partial N_m}{\partial x_j} \right] d\Omega_e f_{bi}$
$SF_4 = \int_{\Omega_e} \tau_p \left[f_{bj} \frac{\partial N_m}{\partial x_j} \right] d\Omega_e$	$F_i = \int_{\partial\Omega_e} N_m \Lambda_i d\Gamma_e + \int_{\partial\Omega_e} \rho N_m f_{Bi} d\Omega_e$

for our further segregated formulation in the next chapter.

$$\begin{aligned}
 & \begin{bmatrix} M_1 & & & \\ & M_2 & & \\ & & M_3 & \\ & & & 0 \end{bmatrix} \begin{Bmatrix} \dot{\hat{\mathbf{u}}} \\ \dot{\hat{\mathbf{v}}} \\ \dot{\hat{\mathbf{w}}} \\ \dot{\hat{\mathbf{p}}} \end{Bmatrix} \\
 & + \begin{bmatrix} C(\mathbf{u}) + Su(\mathbf{u}) & & & Sp_1(\mathbf{u}) \\ & C(\mathbf{u}) + Su(\mathbf{u}) & & Sp_2(\mathbf{u}) \\ & & C(\mathbf{u}) + Su(\mathbf{u}) & Sp_3(\mathbf{u}) \\ -K_{\tau_1}(\mathbf{u}) & -K_{\tau_2}(\mathbf{u}) & -K_{\tau_3}(\mathbf{u}) & 0 \end{bmatrix} \begin{Bmatrix} \hat{\mathbf{u}} \\ \hat{\mathbf{v}} \\ \hat{\mathbf{w}} \\ \hat{\mathbf{p}} \end{Bmatrix} \\
 & + \begin{bmatrix} 2K_{11} + K_{22} + K_{33} & K_{12} & K_{13} & -Q_1 \\ K_{21} & K_{11} + 2K_{22} + K_{33} & K_{23} & -Q_2 \\ K_{31} & K_{32} & K_{11} + K_{22} + 2K_{33} & -Q_3 \\ -Q_1^T & -Q_2^T & -Q_3^T & 0 \end{bmatrix} \begin{Bmatrix} \hat{\mathbf{u}} \\ \hat{\mathbf{v}} \\ \hat{\mathbf{w}} \\ \hat{\mathbf{p}} \end{Bmatrix} \\
 & + \begin{bmatrix} S\delta_{11} & S\delta_{12} & S\delta_{13} & 0 \\ S\delta_{21} & S\delta_{22} & S\delta_{23} & 0 \\ S\delta_{31} & S\delta_{32} & S\delta_{33} & 0 \\ 0 & 0 & 0 & -Q_\tau \end{bmatrix} \begin{Bmatrix} \hat{\mathbf{u}} \\ \hat{\mathbf{v}} \\ \hat{\mathbf{w}} \\ \hat{\mathbf{p}} \end{Bmatrix} = \begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ 0 \end{Bmatrix} + \begin{Bmatrix} SF_1(\mathbf{u}) \\ SF_2(\mathbf{u}) \\ SF_3(\mathbf{u}) \\ -SF_4 \end{Bmatrix} \quad (43)
 \end{aligned}$$

It is noted that since all of the SUPG τ parameters depend on the element size, in the limit as the element size gets smaller, the stabilized method reduces to

the Galerkin form. In another words, stabilization plays a significant role only for coarse mesh.

5.4 Formulation of Stabilized Equal-Order Segregated Scheme

This section discusses the strategies for the solution of large systems of equations arising from the finite-element discretization of the formulations. To solve the nonlinear fluid flow and heat transfer problem, particular emphasis is placed on segregated scheme in nonlinear level and iterative methods in linear level. The reality of velocity–pressure coupling formulation in the last section not only results in a large dimensioned system but also generates a stiffness matrix radically different to the narrowband type of matrix, which is inefficient to solve and the advection term causes solution instability.

The first part in the section is the kernel part; it is a uniformed consistent solution scheme for nonlinear and linear level to support any fluid flow/heat transfer and transport phenomena. The combination of “segregated scheme,” “iterative solver,” and “Block I/O” composes the whole solution picture in a very efficient and robust way. First, the decoupled pressure–velocity segregated scheme is discussed and expended in very detail under finite-element method by nonlinear level, and then the Generalized Minimum Residual Method (GMRES)-based iterative technique and matrix preconditioners are used to solve linear equations. With respect to implementation, the condensed storage and block I/O technique is demonstrated to boost solution efficiency especially for large systems.

5.4.1 Introduction

Several categorizations can be made by using finite-element method to solve the Navier–Stokes equation for fluid flow and thermal problems. They are mainly (a) velocity–pressure coupling and (b) orders of interpolation functions for velocity and pressure.

The first categorization is based on how the primitive variables of velocity and pressure are treated. The solution schemes are categorized into three groups: velocity–pressure integrated method, penalty method, and segregated velocity–pressure method. The velocity–pressure integrated scheme treats velocity and pressure simultaneously; it needs less iteration number in nonlinear level, but larger global matrix requires longer execution time and a large memory in linear level; and usually its overall runtime performance is slow. The penalty scheme eliminates the direct pressure computation by introducing penalty function, thus speeding up the execution time and memory requirement; however, pressure field from simple post-process is not satisfying especially for large Reynolds number fluid flow problem. Recently, much attention has been paid to the segregated velocity–pressure scheme, where velocity and corresponding pressure field are computed alternatively in an iterative sequence. This scheme has the advantage of speed and memory usage with

the price of complicated formulation and possible less solution robustness due to velocity–pressure decoupling (addressed later).

The second categorization is based on the mixed-order interpolation and equal-order interpolation according to the orders of interpolation functions for the velocity and pressure in element level. The mixed-order interpolation attempts to eliminate the tendency to produce checkerboard pressure pattern and to satisfy LBB condition; the velocity in this method is interpolated linearly, whereas the pressure is assumed to be constant; this interpolation is referred to as Q1P0. The equal-order interpolation in Ref. [5] uses linear interpolation for velocity–pressure formulation (referred to as Q1P1) without exhibiting spurious pressure modes, by employing nonconsistent pressure equations (NCPEs) for pressure correction. This method makes the solver more effective than constant element pressure in an element for Q1P0 method.

The objective of this chapter is to explore further the merits of the finite-element methods in simulating the flow where the upstream effects play important roles. Thus, we are presenting the details of making use of both the segregated velocity–pressure and equal-order formulations on the basis of SUPG method.

The basic idea of segregated algorithms is to decouple the pressure calculation from the velocity calculation by taking the divergence of the vector momentum equation and applying some clever insights regarding incompressible flow. Early motivation for this approach was largely twofold: to mitigate memory requirements of fully coupled algorithms and to enable semi-implicit time integration.

Pressure–velocity segregation methods have been reviewed by several researchers in the context of the finite-element method; most notable are the papers by Gresho and Haroutunian et al. [2]. Basically, all current segregated algorithm variants are distinguished by the way in which the pressure is decoupled and projected from one time step to the next. Haroutunian and Engelman [2] proposed three consistent finite-element counterparts to the SIMPLE and SIMPLER algorithm. To further reduce the size of the submatrix systems, each individual component of the momentum equations was solved separately and successively by iterative techniques. Overall at each Newton iteration or Picard iteration they solved four matrix subsystems, one for each of three velocity components and one for the pressure. Interestingly, the most challenging matrix system to solve happens to be one arising from the discretization of the pressure equation; here, the right-hand side SF_4 in equation (5.43) is lagged from the last iteration so that this equation is solved solely for the pressure. The resulting matrix, despite being symmetric, is actually very poorly conditioned due to poor scaling. Nonetheless, these challenging matrix systems can be readily solved by modern iterative solvers and reordering/preconditioner strategies, and the success of this algorithm over the last decade has been enormous.

In our view this approach is still a compromise to the favorable convergence properties of a fully coupled technique advocated here. Convergence to stable solutions at successive time steps is linear at best case and sometimes asymptotic, sometimes resulting in large number of required segregation iterations (albeit fast iterations). Moreover, the method introduces several relaxation parameters that

must be tuned to the application. If one chooses alternatives to the primitive variable formulation, then boundary conditions on velocity become challengeable to apply accurately. Finally, codes centered around this algorithm are more complex in structure, as they still contain the intricacies of matrix solution services but involve more than one data structure (for the mixed interpolation) and more inner and outer iteration loops.

5.4.2 FEM-based segregated formulation

This formulation is based on the algorithm of Rice and Schnipke's [5] and Shaw's [7] pioneer works; they use non-consistent pressure equations (NCPE) for the pressure step and pressure correction step rather than the conventional weak form of the flow equations resulting from Galerkin finite element context. This approach has gained popularity in recent years with the very desirable advantage of precluding spurious pressure modes from pressure solution. This consequently allows them to employ the desirable feature of "equal-order interpolation for the velocity and pressure." For this reason, the algorithm can solve the flow without using very refined meshes and hybrid element types highly desirable for commercial code obtained from CAD model. However, the expanse of this algorithm includes the extra work of imposing boundary conditions properly and their complex formulation. In this section, we show the formulation derivation details starting from the N–S equation.

Bilinear interpolation

The fluid flow continuity equation and the momentum equation follow same form as listed in Section 3.1. For implementation using finite-element method, velocity and pressure at arbitrary position could be interpolated using "equal-order bilinear interpolation" to adapt brick or other degenerated elements.

$$u(x, y) = N_i u_i \quad (44a)$$

$$v(x, y) = N_i v_i \quad (44b)$$

$$w(x, y) = N_i w_i \quad (44c)$$

$$p(x, y) = N_i p_i \quad (44d)$$

Discretization and system assembly

Using the standard Galerkin approach, the momentum equation is multiplied by weighting functions, and using the Green–Gauss theorem, the diffusion terms are integrated by parts in x momentum equation as follows:

$$\begin{aligned} \int_{\Omega} \rho N \left(u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) d\Omega &= - \int_{\Omega} N \frac{\partial p}{\partial x} d\Omega + \int_{\Omega} N F_x d\Omega \\ &+ \mu \int_{\Omega} N \left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right] d\Omega \end{aligned} \quad (45)$$

Construct a function,

$$\begin{aligned} & \int_{\Omega} \left[\frac{\partial \left(\frac{\partial u}{\partial x} N \right)}{\partial x} + \frac{\partial \left(\frac{\partial u}{\partial y} N \right)}{\partial y} + \frac{\partial \left(\frac{\partial u}{\partial z} N \right)}{\partial z} \right] d\Omega \\ &= \int_{\Omega} \left(\frac{\partial u}{\partial x} \frac{\partial N}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial N}{\partial y} + \frac{\partial u}{\partial z} \frac{\partial N}{\partial z} \right) d\Omega + \int_{\Omega} N \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) d\Omega \end{aligned} \quad (46)$$

Using Green–Gauss theorem,

$$\int_{\Omega} \left[\frac{\partial \left(\frac{\partial u}{\partial x} N \right)}{\partial x} + \frac{\partial \left(\frac{\partial u}{\partial y} N \right)}{\partial y} + \frac{\partial \left(\frac{\partial u}{\partial z} N \right)}{\partial z} \right] d\Omega = \int_{\Gamma} N \left(\frac{\partial u}{\partial x} n_x + \frac{\partial u}{\partial y} n_y + \frac{\partial u}{\partial z} n_z \right) d\Gamma \quad (47)$$

Equations (45), (46), and (47) are combined to get the new form of the discretized x -momentum equation:

$$\begin{aligned} & \int_{\Omega} \rho N \left(u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) d\Omega = - \int_{\Omega} N \frac{\partial p}{\partial x} d\Omega + \int_{\Omega} N F_x d\Omega \\ &+ \int_{\Gamma} N \left(\frac{\partial u}{\partial x} n_x + \frac{\partial u}{\partial y} n_y + \frac{\partial u}{\partial z} n_z \right) d\Gamma - \int_{\Omega} \left(\frac{\partial u}{\partial x} \frac{\partial N}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial N}{\partial y} + \frac{\partial u}{\partial z} \frac{\partial N}{\partial z} \right) d\Omega \end{aligned}$$

Move the last item in the right-hand side to the left-hand side, then

$$\begin{aligned} \text{LHS} = [A_u]\{u\} &= \int_{\Omega} \left[\rho N \left(u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) \right. \\ &\quad \left. + \left(\mu \frac{\partial u}{\partial x} \frac{\partial N}{\partial x} + \mu \frac{\partial u}{\partial y} \frac{\partial N}{\partial y} + \mu \frac{\partial u}{\partial z} \frac{\partial N}{\partial z} \right) \right] d\Omega \\ &= \rho \int_{\Omega} N_i \left[u \frac{\partial N_j}{\partial x} + v \frac{\partial N_j}{\partial y} + w \frac{\partial N_j}{\partial z} \right] d\Omega \{u\} + \left[\int_{\Omega} \mu \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} d\Omega \right] \{u\} \\ &\quad + \left[\int_{\Omega} \mu \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} d\Omega \right] \{u\} + \left[\int_{\Omega} \mu \frac{\partial N_i}{\partial z} \frac{\partial N_j}{\partial z} d\Omega \right] \{u\} \\ &= [C(u, v, w) + K_{11} + K_{22} + K_{33}] \{u\} \end{aligned} \quad (48a)$$

where, $C(U)$ is the advection item and K_{11}, K_{22}, K_{33} are the diffusion items. To obtain $C(U)$, velocities u, v, w are considered the average values in an element.

$$\begin{aligned} \text{RHS} &= - \int_{\Omega} N_i \frac{\partial p}{\partial x} d\Omega + \int_{\Omega} N_i F_x d\Omega + \int_{\Gamma} N_i \left(\frac{\partial u}{\partial x} n_x + \frac{\partial u}{\partial y} n_y + \frac{\partial u}{\partial z} n_z \right) d\Gamma \\ d\Gamma &= -M_{14}\{p\} + \{F_{xA}^{\Gamma}\} + \{F_x\} \end{aligned} \quad (48b)$$

$[A_v]$ and $[A_w]$ could be achieved by similar approaches.

SUPG items The SUPG weighting function proposed by Brooks is defined by:
 $W_k = N_k + \bar{p}_k$

Here the streamline-upwind contribution \bar{p}_k is defined as:

$$\bar{p}_k = \frac{\tilde{k}^e}{(u_1^e)^2 + (u_2^e)^2} \left(u_j^e \frac{\partial N_i}{\partial x_j} \right) \quad (49)$$

Only convective items are weighted with the SUPG functions for the inconsistent approach, which is believed to be more stable than the consistent SUPG formulation [10]. The extra items from \bar{p}_k should be added into K_{11}, K_{22}, K_{33} in equation (49) in the implementation.

Equation of pseudo velocity (prediction phase)

In matrix form of the segregated momentum equations:

$$[M_{11}]\{u\} = -[M_{14}]\{p\} + \{F_{xA}^\Gamma\} + \{F_x\} \quad (50a)$$

$$[M_{22}]\{v\} = -[M_{24}]\{p\} + \{F_{yA}^\Gamma\} + \{F_y\} \quad (50b)$$

$$[M_{33}]\{w\} = -[M_{34}]\{p\} + \{F_{zA}^\Gamma\} + \{F_z\} \quad (50c)$$

where

$$[M_{11}] = [C(u, v, w) + K_{11} + K_{22} + K_{33}] + [\text{SUPG}(u, v, w)]$$

$$[M_{22}] = [C(u, v, w) + K_{11} + K_{22} + K_{33}] + [\text{SUPG}(u, v, w)]$$

$$[M_{33}] = [C(u, v, w) + K_{11} + K_{22} + K_{33}] + [\text{SUPG}(u, v, w)]$$

$$[M_{14}] = \int_{\Omega} N_i \frac{\partial p}{\partial x} d\Omega$$

$$[M_{24}] = \int_{\Omega} N_i \frac{\partial p}{\partial y} d\Omega$$

$$[M_{34}] = \int_{\Omega} N_i \frac{\partial p}{\partial z} d\Omega$$

$$\{F_{xA}^\Gamma\} = \int_{\Gamma} N_i \left(\frac{\partial u}{\partial x} n_x + \frac{\partial u}{\partial y} n_y + \frac{\partial u}{\partial z} n_z \right) d\Gamma$$

$$\{F_{yA}^\Gamma\} = \int_{\Gamma} N_i \left(\frac{\partial u}{\partial x} n_x + \frac{\partial u}{\partial y} n_y + \frac{\partial u}{\partial z} n_z \right) d\Gamma$$

$$\{F_{zA}^\Gamma\} = \int_{\Gamma} N_i \left(\frac{\partial u}{\partial x} n_x + \frac{\partial u}{\partial y} n_y + \frac{\partial u}{\partial z} n_z \right) d\Gamma$$

$$\{F_x\} = \int_{\Omega} N_i F_x d\Omega$$

$$\{F_y\} = \int_{\Omega} N_i F_y d\Omega$$

$$\{F_z\} = \int_{\Omega} N_i F_z d\Omega$$

where Ω is the fluid domain (volume); and Γ is the flow boundaries (surface).

There are no SUPG items in $\{F_x\}\{F_y\}\{F_z\}$ for the inconsistent approach.

The above three equations can be solved separately to obtain the velocity estimate (prediction phase).

Hat Velocity (V_{hat}) Rewrite equations (50a), (50b), (50c):

$$a_{uii}u_i = -\sum_{i \neq j} a_{uij}u_j - \int_{\Omega} N_i \frac{\partial p}{\partial x} d\Omega + \mu \int_{\Gamma} N_i \left[\frac{\partial u}{\partial x} n_x + \frac{\partial u}{\partial y} n_y + \frac{\partial u}{\partial z} n_z \right] d\Gamma + \{F_x\}_i \quad (51a)$$

$$a_{vii}v_i = -\sum_{i \neq j} a_{vij}v_j - \int_{\Omega} N_i \frac{\partial p}{\partial y} d\Omega + \mu \int_{\Gamma} N_i \left[\frac{\partial v}{\partial x} n_x + \frac{\partial v}{\partial y} n_y + \frac{\partial v}{\partial z} n_z \right] d\Gamma + \{F_y\}_i \quad (51b)$$

$$a_{wii}w_i = -\sum_{i \neq j} a_{wij}w_j - \int_{\Omega} N_i \frac{\partial p}{\partial z} d\Omega + \mu \int_{\Gamma} N_i \left[\frac{\partial w}{\partial x} n_x + \frac{\partial w}{\partial y} n_y + \frac{\partial w}{\partial z} n_z \right] d\Gamma + \{F_z\}_i \quad (51c)$$

For constant pressure gradient in an element, rewrite equation (51a):

$$u_i = \frac{1}{a_{uii}} \left[-\sum_{i \neq j} a_{uij}u_j + \mu \int_{\Gamma} N_i \left[\frac{\partial u}{\partial x} n_x + \frac{\partial u}{\partial y} n_y + \frac{\partial u}{\partial z} n_z \right] d\Gamma + \{F_x\}_i \right] \\ - \frac{1}{a_{uii}} \left[\int_{\Omega} N_i d\Omega \right] \frac{\partial p}{\partial x}$$

i.e.,

$$u_i = \hat{u}_i - k_i^p \frac{\partial p}{\partial x} \quad (52a)$$

where

$$k_i^p = \frac{1}{a_{uii}} \int_{\Omega} N_i d\Omega \quad (53a)$$

$$\hat{u}_i = \frac{1}{a_{uii}} \left[-\sum_{i \neq j} a_{uij}u_j + \mu \int_{\Gamma} N_i \left[\frac{\partial u}{\partial x} n_x + \frac{\partial u}{\partial y} n_y + \frac{\partial u}{\partial z} n_z \right] d\Gamma + \{F_x\}_i \right] \quad (54a)$$

Rewrite equation (51b):

$$v_i = \frac{1}{a_{vii}} \left[-\sum_{i \neq j} a_{vij}v_j + \mu \int_{\Gamma} N_i \left[\frac{\partial v}{\partial x} n_x + \frac{\partial v}{\partial y} n_y + \frac{\partial v}{\partial z} n_z \right] d\Gamma + \{F_y\}_i \right] \\ - \frac{1}{a_{vii}} \left[\int_{\Omega} N_i d\Omega \right] \frac{\partial p}{\partial y} \quad (50)$$

i.e.,

$$v_i = \hat{v}_i - k_i^p \frac{\partial p}{\partial y} \quad (52b)$$

where

$$k_{vi}^p = \frac{1}{a_{vii}} \int_{\Omega} N_i d\Omega \quad (53b)$$

$$\hat{v}_i = \frac{1}{a_{vii}} \left[-\sum_{i \neq j} a_{vij} v_j + \mu \int_{\Gamma} N_i \left[\frac{\partial v}{\partial x} n_x + \frac{\partial v}{\partial y} n_y + \frac{\partial v}{\partial z} n_z \right] d\Gamma + \{F_y\}_i \right] \quad (54b)$$

Rewrite equation (51c):

$$w_i = \frac{1}{a_{wii}} \left[-\sum_{i \neq j} a_{wij} w_j + \mu \int_{\Gamma} N_i \left[\frac{\partial w}{\partial x} n_x + \frac{\partial w}{\partial y} n_y + \frac{\partial w}{\partial z} n_z \right] d\Gamma + \{F_z\}_i \right] \\ - \frac{1}{a_{wii}} \left[\int_{\Omega} N_i d\Omega \right] \frac{\partial p}{\partial z}$$

i.e.,

$$w = \hat{w}_i - k_i^p \frac{\partial p}{\partial z} \quad (52c)$$

where

$$k_{wi}^p = \frac{1}{a_{wii}} \int_{\Omega} N_i d\Omega \quad (53c)$$

$$\hat{w}_i = \frac{1}{a_{wii}} \left[-\sum_{i \neq j} a_{wij} w_j + \mu \int_{\Gamma} N_i \left[\frac{\partial w}{\partial x} n_x + \frac{\partial w}{\partial y} n_y + \frac{\partial w}{\partial z} n_z \right] d\Gamma + \{F_z\}_i \right] \quad (54c)$$

To ensure mass continuity, the velocity should be updated (corrected) at each iteration by which velocity correction is reformulated from the new pressure value, refer to equation 62a, 62b and 62c for velocity correction details.

$$V_{Corrected} = V_{hat} + V_{Correction_Term_By_Pressure} \quad (55)$$

Pressure Poisson equation

On applying Green–Gauss theorem and integration to the continuity equation, the integral of the divergence in a domain equals the net flux across the domain boundary:

$$\int_{\Omega} \left[\frac{\partial(uN)}{\partial x} + \frac{\partial(vN)}{\partial y} + \frac{\partial(wN)}{\partial z} \right] d\Omega = \int_{\Gamma} N \vec{v} \cdot \vec{n} d\Gamma \quad (56)$$

The continuity equation becomes:

$$\int_{\Omega} N \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) d\Omega = - \int_{\Omega} \left(\frac{\partial N}{\partial x} u + \frac{\partial N}{\partial y} v + \frac{\partial N}{\partial z} w \right) d\Omega \\ + \int_{\Gamma} N (u n_x + v n_y + w n_z) d\Gamma = 0 \quad (57)$$

The modified continuity equation (57) using green function transformation is targeted to solve pressure.

Step 1: Substitute interpolated velocity from equations (44a), (44b), and (44c) to continuity equation (57)

$$-\int_{\Omega} \left(\frac{\partial N_i}{\partial x} (N_j u_j) + \frac{\partial N_i}{\partial y} (N_j v_j) + \frac{\partial N_i}{\partial z} (N_j w_j) \right) d\Omega + \int_{\Gamma} N_i (u n_x + v n_y + w n_z) d\Gamma = 0 \quad (58)$$

Step 2: Substitute velocity prediction from equations (53a), (53b), and (53c) into equation (58)

$$\begin{aligned} & \int_{\Omega} \left[\frac{\partial N_i}{\partial x} \left(N_j K_j^p \frac{\partial p}{\partial x} \right) + \frac{\partial N_i}{\partial y} \left(N_j K_j^p \frac{\partial p}{\partial y} \right) + \frac{\partial N_i}{\partial z} \left(N_j K_j^p \frac{\partial p}{\partial z} \right) \right] d\Omega \\ &= \int_{\Omega} \left(\frac{\partial N_i}{\partial x} (N_j \hat{u}_j) + \frac{\partial N_i}{\partial y} (N_j \hat{v}_j) + \frac{\partial N_i}{\partial z} (N_j \hat{w}_j) \right) d\Omega - \int_{\Gamma} N_i (u n_x + v n_y + w n_z) d\Gamma \end{aligned} \quad (59)$$

Step 3: Rewrite to final form

Notice that $p = N_i p_i$, so substituting $\frac{\partial p}{\partial x} = \frac{\partial(N_i p_i)}{\partial x} = p_i \frac{\partial N_i}{\partial x}$ (assume that p is constant in an element here) in equation (59) (using similar way for and) we obtain $\frac{\partial p}{\partial y}$ and $\frac{\partial p}{\partial z}$.

$$\begin{aligned} & \left\{ \int_{\Omega} \left[\frac{\partial N_i}{\partial x} (N_k K_k^p) \left(\frac{\partial N_j}{\partial x} \right) \right] d\Omega + \int_{\Omega} \left[\frac{\partial N_i}{\partial y} (N_k K_k^p) \frac{\partial N_j}{\partial y} \right] d\Omega \right. \\ & \quad \left. + \int_{\Omega} \left[\frac{\partial N_i}{\partial z} (N_k K_k^p) \frac{\partial N_j}{\partial z} \right] d\Omega \right\} \cdot \{p\} \\ &= \int_{\Omega} \frac{\partial N_i}{\partial x} (N_j \hat{u}_j) d\Omega + \int_{\Omega} \frac{\partial N_i}{\partial y} (N_j \hat{v}_j) d\Omega + \int_{\Omega} \frac{\partial N_i}{\partial z} (N_j \hat{w}_j) d\Omega \\ & \quad - \int_{\Gamma} N_i (u n_x + v n_y + w n_z) d\Gamma \end{aligned} \quad (60)$$

Moving V_{hat} out of the integration in equation (60), we obtain:

$$\begin{aligned} & \left\{ \int_{\Omega} \left[\frac{\partial N_i}{\partial x} (N_k K_k^p) \left(\frac{\partial N_j}{\partial x} \right) \right] d\Omega + \int_{\Omega} \left[\frac{\partial N_i}{\partial y} (N_k K_k^p) \frac{\partial N_j}{\partial y} \right] d\Omega \right. \\ & \quad \left. + \int_{\Omega} \left[\frac{\partial N_i}{\partial z} (N_k K_k^p) \frac{\partial N_j}{\partial z} \right] d\Omega \right\} \cdot \{p\} \\ &= \int_{\Omega} \frac{\partial N_i}{\partial x} N_j d\Omega \{\hat{u}\} + \int_{\Omega} \frac{\partial N_i}{\partial y} N_j d\Omega \{\hat{v}\} + \int_{\Omega} \frac{\partial N_i}{\partial z} N_j d\Omega \{\hat{w}\} \\ & \quad - \int_{\Gamma} N_i (u n_x + v n_y + w n_z) d\Gamma \end{aligned} \quad (61)$$

Now equation (61) is the final form of pressure Poisson equation.

It is interesting to note that the element pressure matrices are identical to those obtained in classical diffusion-type problems, with the term K replacing the diffusion coefficient.

Here are more discussions on segregated solution scheme.

- (1) The coefficient matrix for the pressure equation is similar to that obtained for the diffusion term in the conventional finite-element formulation if the viscosity is replaced by $N_i K_j^P$, the formulation developed by Rice [5] and Vellando [12]; this fact is indicative of the stability and robust nature of the resulting pressure field.
- (2) The LHS and RHS of equation (69) could be constructed and evaluated in element level and then assembled into global matrix LHS and RHS; this is typical finite-element behavior.
- (3) In equation (60), k_i^P is a node-based property, it should be assembled from the saved element-level matrix, $\int_{\Omega} N_i d\Omega$, a_{uii} is also a global property from assembly.
- (4) Equation (60) is in the form of Poisson equation; the LHS matrix has the properties of symmetric and positive definite as indicated by Shaw [7].
- (5) Special boundary integrations on all inflow and outflow boundaries are required to solve pressure, which corresponds to the fourth item in equation (61).

Velocity correction

Write equations (51a), (51b), and (51c) using weighting function $p(x,y) = N_i p_i$ and chain law, and discarding velocity gradient items on boundaries. \hat{u}_i , \hat{v}_i , \hat{w}_i are used for the velocity prediction as “hat” velocity.

$$u_i = \frac{1}{a_{uii}} \left[-\sum_{i \neq j} a_{uij} u_j + \{F_x\}_i \right] - \frac{1}{a_{uii}} \left[\int_{\Omega} N_i \frac{\partial N_j}{\partial x} d\Omega \right] \{p\}$$

$$\text{Or, } u_i = \hat{u}_i - \frac{1}{a_{uii}} \left[\int_{\Omega} N_i \frac{\partial N_j}{\partial x} d\Omega \right] \{p\} = \hat{u}_i - \frac{1}{a_{uii}} \{M_{14}\} \{p\} \quad (62a)$$

Get corrected velocity v , w in the same way as u ,

$$v_i = \hat{v}_i - \frac{1}{a_{vii}} \left[\int_{\Omega} N_i \frac{\partial N_j}{\partial y} d\Omega \right] \{p\} = \hat{v}_i - \frac{1}{a_{vii}} \{M_{24}\} \{p\} \quad (62b)$$

$$w_i = \hat{w}_i - \frac{1}{a_{wii}} \left[\int_{\Omega} N_i \frac{\partial N_j}{\partial z} d\Omega \right] \{p\} = \hat{w}_i - \frac{1}{a_{wii}} \{M_{34}\} \{p\} \quad (62c)$$

At this step, u , v , w are directly corrected from the predicted and with trivial effort, the second item is the pressure coupling effect on the velocity.

Special treatments on boundary conditions

There are some special treatments in the segregated formulation.

Wall boundary: For wall boundary, the surface integral or natural boundary condition for the pressure equation is zero. To incorporate the known velocities into

pressure equation, it is required that “hat” velocity be zero. In implementation, the pressure coefficients, $k_{ui}^p, k_{vi}^p, k_{wi}^p$ at the wall nodes should be set as zero.

Specified velocity boundary: The surface integration of velocity boundary conditions should be added into the pressure equation (61)’s right hand side as in the fourth item in the right hand side; for specified velocity boundary, the contribution is fixed.

Inflow/Outflow boundary Zero gradient velocity boundary condition is applied at the inflow/outflow boundaries (including inlet/outlet boundary and pressure boundary), i.e., no special treatment in the momentum equation is required at this kind of boundary. However, the contribution from the surface integration of natural boundary conditions should be added to the right-hand side of the pressure equation as done in the fourth item on the right-hand side of equation (60).

Segregated scheme not only simply divides the big coupling matrix into smaller matrices, but also sets up a pressure equation (Poisson equation) with positive-definite symmetric properties, which can greatly expedite the pressure solution with better accuracy. Most of the fluid flow problems are driven by pressure, and therefore the velocity components will follow a good solution of pressure in practice.

Solution stability control

Inertial relaxation. Inertial relaxation is used in the governing equations to slow down the convergence rate in the same manner as transit terms are used in nonsteady problems, so the inertial relaxation factor is also called virtual time step.

The inertial relaxation is added to the governing equations in the following manner:

$$\left[A_{ii} + \frac{\rho_i \int N d\Omega}{\Delta t_{inertia}} \right] \phi_i + \sum_{j \neq i} A_{ij} \phi_j = F_i + \frac{\rho_i \int N d\Omega}{\Delta t_{inertia}} \phi_{iold} \quad (63)$$

The second term in parentheses and the last term on the right-hand side are the inertial relaxation terms. $\Delta t_{inertia}$ can be automatically adjusted according to the global matrix condition number.

- Small $\Delta t_{inertia}$ value will enhance the solution stability and slow down the convergence rate.
- $\Delta t_{inertia}$ is applied to momentum equations only, and it will indirectly affect the pressure equation because of the slow marching of the velocity change.
- For convergence solution, the inertial term has no affect on the final solution because it was eliminated in both LHS and RHS.

Explicit relaxation for momentum and pressure equations. Explicit relaxation for momentum and pressure equations is the regular technique, which is also called under-relaxation. In this form, the new solution is weighted by the old solution using the formula:

$$\phi = (1 - \alpha)\phi_{old} + \alpha\phi_{new} \quad (64)$$

where ϕ_{new} is the updated current solution and ϕ_{old} is the previous value. α is the relaxation factor. If $\alpha = 1$, all previous values are ignored, and all updated values are

used. For convergence difficulties, low value of α will help. Automatic parameter control was adapted to eliminate the user's input and adjustment effort. The control method based on the convergence history identifies the convergence pattern via an intelligence control theory as proposed by Xie [16].

Solution strategy in fluid flow and heat transfer problems

Because the governing equations are nonlinear, they must be solved iteratively. Picard successive substitution or Newton–Raphson iterative methods are used. In Picard successive substitution method, estimates of the solution variables (U, V, W, P, K, E, T , etc.) are substituted into the governing equations. The equations are solved for new values that are then used as the estimates for the next pass. The global solver controller will either perform a fixed number of these global iterations or check for the convergence criterion. The controller will stop when either is reached. The convergence criterion is the level at which the specified variable's residual norm must reach.

Method A: For weak (one way) coupled flow and heat transfer problem

```

Loop (nonlinear) until convergence
Solve momentum equations by segregated scheme
Solve pressure equations and velocity correction by segregated scheme
End Loop
Solve energy equations

```

Method B: For fully coupled flow and heat transfer problem

```

Loop (nonlinear) until convergence
Solve momentum equations by segregated scheme
Solve pressure equations and velocity correction by segregated scheme
Solve energy equations
End Loop

```

Method C: For weak (one way) coupled flow and heat transfer problem

```

Loop (nonlinear) until convergence
Solve coupled continuity, momentum equations
End Loop
Solve energy equations

```

Method D: For fully coupled flow and heat transfer problem

```

Loop (nonlinear) until convergence
Solve coupled continuity, momentum equations
Solve energy equations
End Loop

```

Method E: For fully coupled flow and heat transfer problem

Loop (nonlinear) until convergence
 Solve wholly coupled continuity, momentum and energy equations
 End Loop

Owing to the importance of the segregated pressure–velocity solution scheme versus coupled scheme in most of the convective heat transfer problems, the majority of computational efforts in heat transfer problems lie in solving the flow and pressure fields.

Scenario I demonstrates the decoupled flow and heat transfer, i.e., there are no temperature-dependent properties in the flow. We can first solve the isothermal flow (energy equation turned off) to yield a converged flow-field solution and then solve the energy transport equation alone; this is a naturally segregated approach. Scenario II depicts the coupled flow and heat transfer (typically natural-convection problem); the common solution practice is to realize the flow–thermal coupling using the nonlinear iteration until convergence. This is exactly the advantage of the velocity–pressure “segregated scheme.”

Figure 5.4 shows the five solution schemes. Methods A and B are segregated schemes described in this chapter, methods C and D are coupled momentum and pressure schemes, and method E is completely coupled velocity–pressure–temperature solution scheme. Method E is rarely used because of the bulk properties in global matrix by nature especially for large industry application. In any case, all methods include solving energy equation.

For multiphysics system of equations to be solved by segregated manner, the solution scheme is illustrated in Figure 5.4:

1. Velocity–pressure sequential solver (momentum-continuity)
2. Turbulence subiteration
3. Nonlinear iterations
4. Time stepping iteration

At each time step, the nonlinear iterations are performed for the momentum-continuity, turbulence, front-tracking equation, and temperature. Subiterations of turbulence transport equations are also used to accelerate the overall convergence of the iterative process.

In more general and wider range multiphysics applications with structure, electromagnetic, acoustics disciplines coupling with fluid flow and heat transfer, typical application such as fluid–structure interaction (FSI), two-tier approaches may be applied for the architecture with one unified simulation environment (workbench). In tier one, CFD package tightly integrate fluid flow, heat transfer, mass transfer, and acoustics components, and structure package tightly integrate linear dynamics and explicit dynamics analysis components. In top level tier two, customized interprocess communication technology or general interprocess coupling tools like MPCCI

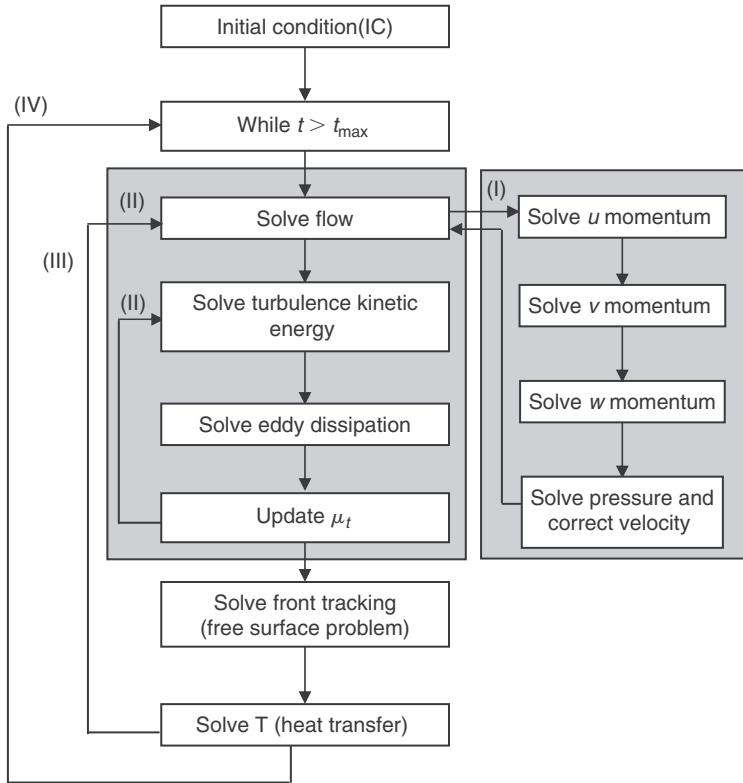


Figure 5.4. General solution procedure of the segregated solver.

could be used for the communication among different independent packages such as CFD, structure, electromagnetic analysis.

5.4.3 Data storage and block I/O process

Data storage

The FEM results in a system of linear equations containing a large number of equations and unknown degree of freedom. A big amount of information requires a clever data-keeping strategy to avoid whole matrix storage. Knowing that the sparse properties of global matrix, we are using data storing which is the so-called skyline or column profile storage. Instead of storing every single matrix element, we could think of storing only the nonzero hits and corresponding position index in the matrix assembly phase (Figure 5.5).

However, even with the use of such a reduced storage mode, the matrix is usually so large that it cannot be stored in core, so that it must be segmented in blocks and stored on low-speed disk storage.

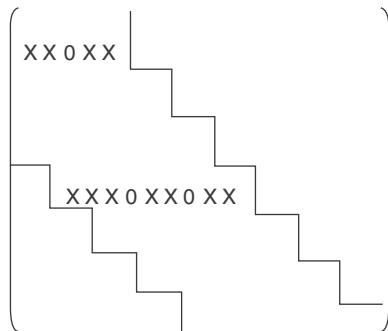


Figure 5.5. Skyline matrix structure.

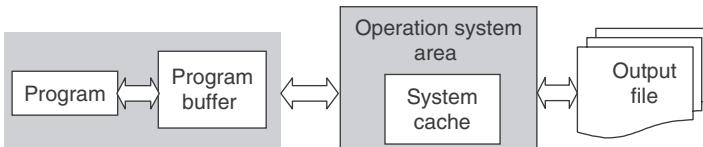


Figure 5.6. Block buffer I/O for disk writing.

Block I/O process

Improving overall disk I/O performance can minimize both device I/O and actual CPU time. The techniques listed here can significantly improve FEA code performance.

The slowest process (routines) could be detected by using profiling tools such as AQTIME. After running a typical size of FEA model, the routines using most time in analysis, the top routines, and even lines are the processor bottleneck. These are the targets to be improved for better performance.

“Block I/O” is a cost-efficient record buffer technique for large amount of data in the FEA processor. For writing, it saves a fixed number of element matrix information into buffer, and then writes the whole buffer into disk if the buffer is full, whereas reading is the reverse procedure. The buffer size is related to file system in operation system and disk cache; we found that the optimal buffer size is 8 kbytes for Windows XP WIN32 NTFS file system.

Here is an example for storage size calculation of one block element stiffness matrix with three-dimensional brick element (eight nodes) (Figure 5.6).

Block size

$$= 128 \text{ matrices}^* (8*8 \text{ entries/matrix}) * 8 \text{ bits /entry}$$

$$= 65,536 \text{ bits}$$

$$= 65,536/8 \text{ bytes}$$

$$= 8,192 \text{ bytes}$$

Table 5.4. Data structure for Block I/O

	Part ID	Element type	Data mapping	Block numbers
Simulation domain	Part 1	Tetrahedral elements (4 nodes)	Block 1 (32 kB)	6 blocks
			Block 2	5 blocks
			Block 3	
			Block 4	
			Block 5	
	Part 2	Other hybrid elements (8 nodes)	Block 6	7 blocks
			Block 6 (32 kB)	
			Block 2	
			Block 3	
			Block 4	
			Block 5	3 blocks
			Block 5 (32 kB)	
			Block 7	
		Other hybrid elements (8 nodes)	Block 7 (32 kB)	
			Block 1 (32 kB)	
			Block 2 (32 kB)	
			Block 3	Block 3

The data structure for block buffer I/O is shown in Table 5.4. The fluid domain is composed of multiple parts, while each part is decoded as the a group of tetrahedral elements and the a group of hybrid elements, hybrid element could be the combination of any brick, pyramid, wedge elements which could be considered as the degraded element from eight-node brick element. For each group, 128*4 matrices from tetrahedral element composed a block; each matrix has 4*4 element entries, which also has the block size of 8 kbytes.

Because the fluid domain is composed of parts and each part has a different number of brick element and tetrahedral element while the number may not be the even times of the optimal block element number, a special counter was used if there were a remainder of element number over block size number.

Apart from the efficient use of record buffers and disk I/O, some other tips as the following were used in the code implementation:

- (a) Using unformatted files instead of formatted files.
- (b) Writing whole arrays.
- (c) Using pointer shallow copy (pointer) instead of the complete deep copy.

By using the above-mentioned comprehensive methods, the I/O performance can be dramatically improved, but when compared to other operations, it becomes trivial.

5.5 Case Studies

This section shows two examples of conjugate heat transfer problems where both solid and fluid are involved in the heat transfer analysis. The results are obtained by using finite element-based equal-order segregated solution scheme discussed in this chapter, and the SUPG stabilization term is enforced in both fluid and thermal processors.

For simple geometry, the high cell Peclet number did not cause severe problem for the simulation results. That is because the temperature gradient in the flow direction happened to be small for the natural boundary condition applied at the exit. Consequently, the numerical diffusion introduced by the discretization also becomes small. For complex geometry, however, heat transfer occurs through both convection and diffusion in all directions. Whenever the convection effects are strong, the numerical oscillation of the computed temperature would happen if the Galerkin formulation were used.

The reason for this numerical difficulty has been well described. The conclusion basically is that the Galerkin formulation is of the same nature as the central finite difference scheme. It does not consider the influence of the flow velocity on the temperature distribution in a cell through its basic interpolation functions. To overcome this difficulty, either an upwind scheme or optimized upwind scheme (SUPG) should be adopted for the convection term. In the upwind schemes, a bias is given to the upwind nodal temperature by modifying the interpolation functions according to the local fluid flow velocity, both the direction and the magnitude.

5.5.1 Two-dimensional air cooling box

The first example shown in Figure 5.7 is a two-dimensional square-shaped electronic box with airflow cooling. Two small copper blocks are heat sources with a constant volumetric heat generation rate. The airflow comes in from an opening at the top edge near the left corner, and the outlet is located at the right side edge near the bottom corner. The flow rate and the temperature at the inlet are specified. The influence of the inlet flow rate on the maximum temperature in the entire system,

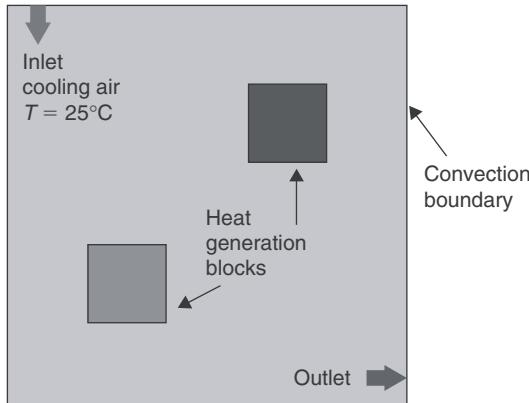


Figure 5.7. Schematic illustration for airflow cooling in two-dimensional electronic case.

Table 5.5. Summary of parameters for airflow cooling system

Inlet velocity U	0.01 or 0.001 (m/s)
Inlet temperature T_{in}	25 ($^\circ\text{C}$)
Case dimensions	5 by 5 (m by m)
Heating block 1	(1, 1) to (2, 2) (x, y) – (x, y)
Heating block 2	(3, 3) – (4, 4) (x, y) – (x, y)
Convection film coefficient	0.2 ($\text{J}/\text{m}^2 \text{s } ^\circ\text{C}$)
Ambient temperature	25 ($^\circ\text{C}$)
Heat generation rate	20 ($\text{J}/\text{s}/\text{m}^3$)

caused by the two heat sources, can be predicted. To verify the effects of the SUPG formulation, this case has also been analyzed using the Galerkin formulation.

The mesh used for the entire domain is 100×100 , so the number of elements for this model is 10,000. Two element groups are used in order to specify heat generation rate for the two heat sources. The flow media is dry air and the two heating blocks are copper. The edges of the two-dimensional box transfer heat to the ambient atmosphere through the natural convection. The convection film coefficient is assumed as a known value. A summary of the parameters used in this case study is given in Table 5.5.

The computed results for two different inlet flow velocities shown in Figure 5.7a and b are obtained if the simulation was carried out based on the Galerkin formulation. For (a) the cell Peclet number is about 20.0, while for (b) it is about 200.0. Temperature oscillations in some area of the flow domain may become severer for the higher cell Peclet number case if SUPG stabilized term is not enforced.

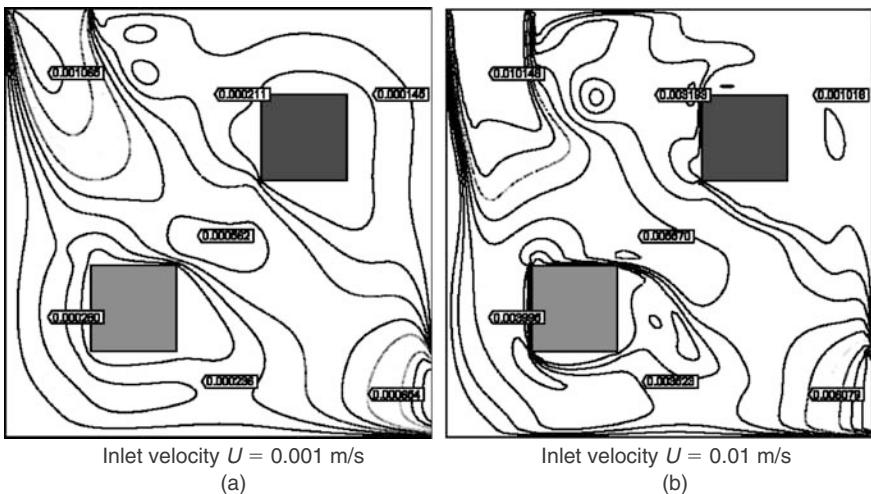


Figure 5.8. Velocity isolines in cooling box.

The cooling effect can be observed by comparing Figure 5.8a and b. The result was achieved by using SUPG stabilized Galerkin discretization; the temperature oscillations were completely eliminated by the optimized upwind SUPG scheme.

The higher inlet velocity makes more effective cooling result, the maximum temperature for (a) is about 113.76°C if inlet velocity is 0.001 m/s , while for (b) it is only about 64.50°C if inlet velocity is 0.01 m/s . Relatively regular velocity and temperature patterns are maintained at smaller velocity since the flow stays in laminar range. With increase in velocity, the flow becomes irregular and more vortices are developed, and the flow convection makes the corresponding temperature patterns.

Because the inlet velocity enters the box from the top side, the momentum lets the flow to hit the bottom heat generation block easily. This effect becomes more obvious with increase in velocity, so that the temperature in the bottom heat generation block is lower than the top heat generation block even if the two blocks have the same amount of heat generation rate (Figure 5.9).

5.5.2 CPU water cooling analysis

A three-dimensional finite-element analysis is established for a CPU water cooling system, as shown in Figure 5.10. The bottom surface of the heat exchange block is actually in contact with the surface of a CPU chip.

A constant heat flux is therefore specified for this surface. It is assumed that the CPU has 130 W power dissipation and the heat is dissipated through the base surface of the heat exchange block. Depending on the surface size and total power, the heat flux boundary condition is set as 38.5 . Other external surfaces of the block are exposed to the ambient atmosphere with a given convection film coefficient of 5 .

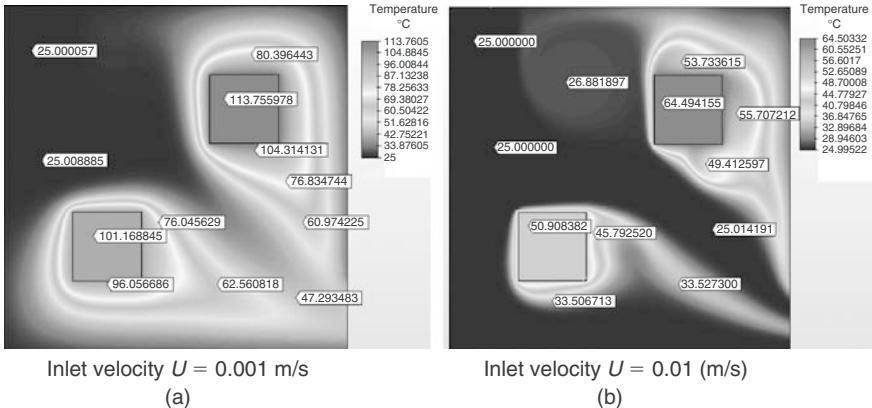


Figure 5.9. Temperature contours in cooling box.

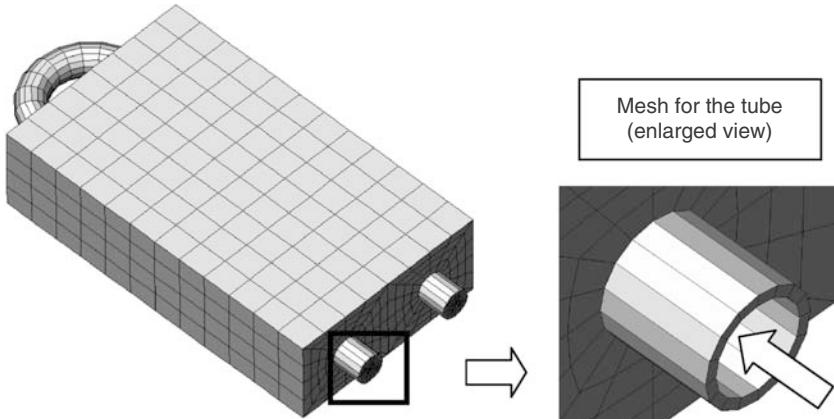


Figure 5.10. 3D model for a CPU water cooling system.

The tube wall is also modeled in this analysis, a part of the tube surface is exposed to the ambient atmosphere, and it has the same convection film coefficient as that of the block surfaces. Both the block and the tube are made of the same material, which is aluminum in this case.

Cooling water with average velocity of 0.06 m/s at inlet goes through the tube to convect the heat out of the system; the velocity profile in the water tube is calculated by fluid flow analysis and then passed to thermal analysis as the convection term. The temperature at the entrance of the tube is specified as 25°C in the analysis. The steady-state conjugated heat transfer problem is calculated by using the equal-order segregated solution scheme and SUPG stabilized term discussed earlier. This example illustrates effective water cooling in the electronic system.

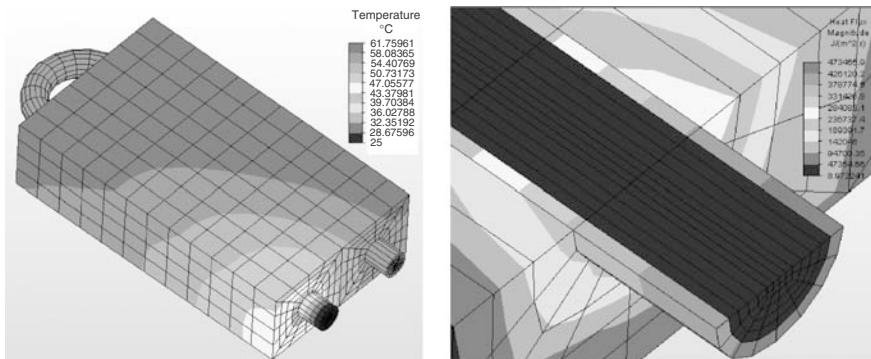


Figure 5.11. Result contour for the CPU water cooling system.

The computed temperature contours are shown in Figure 5.11a. The temperature ranges from 25°C to 61°C. The numerical oscillation of the temperature solution in traditional Galerkin formulation has been eliminated by using the SUPG term. This indicates that the SUPG formulation does provide a much better solution over the Galerkin formulation, especially for convection-dominated heat transfer problems. The heat flux contour in the vicinity of the cooling flow inlet is shown in Figure 5.11b. It is noticed that the maximal heat flux is located in the vicinity of the inlet tube wall, which implies the maximal temperature gradient is there since the heat flux is proportional to the temperature gradient.

References

- [1] Choi, H. G., and Yoo, J. Y. Streamline upwind scheme for the segregated formulation of the Navier-Stokes equation. *Numerical Heat Transfer, Part B*, pp. 145–161, 1994.
- [2] Haroutunian, V., Engelman, M. S., and Hasbani, I. Segregated finite element algorithms for the numerical solution of large-scale incompressible flow problems, *International Journal for Numerical Methods in Fluids*, 17, pp. 323–348, 1993.
- [3] Hill, D. L., and Baskharone, E. A. A monotone streamline upwind method for quadratic finite elements, *International Journal for Numerical Methods in Fluids*, 17, pp. 463–475, 1993.
- [4] Reddy, J. N., and Gartling, D. K. *The Finite Element Method in Heat Transfer and Fluid Dynamics*, 2nd Edition, CRC Press, 2000.
- [5] Rice, J. G., and Schnipke, R. J. A monotone streamline upwind finite element method for convection-dominated flows, *Computer Methods in Applied Mechanics and Engineering*, 48, pp. 313–327, 1985.
- [6] Rice, J. G., and Schnipke, R. J. An equal-order velocity-pressure formulation that does not exhibit spurious pressure modes, *Computer Methods in Applied Mechanics and Engineering*, 58, pp. 135–149, 1986.
- [7] Shaw, C. T. Using a segregated finite element scheme to solve the incompressible Navier-Stokes equations, *International Journal for Numerical Methods in Fluids*, 12, pp. 81–92, 1991.

- [8] Schunk, P. R., and Heroux, M. A. Iterative solvers and preconditioners for fully-coupled finite element formulations of incompressible fluid mechanics and related transport problems, Sandia National Laboratories, Albuquerque, New Mexico, 87185–0835, 2001.
- [9] Tezduyar, T., and Sathe, S. Stabilization parameters in SUPG and PSPG formulations, *Journal of Computational and Applied Mechanics*, 4(1), pp. 71–88, 2003.
- [10] du Toit, C. G. A segregated finite element solution for non-isothermal flow, *Computer Methods in Applied Mechanics and Engineering*, 182(3), pp. 457–481(25), 2000.
- [11] Van Zijl, and du Toit, C. G. A SIMPLEST finite element solution of the incompressible Navier-Stokes equations, In: Proc. Symposium of Finite Element Method, South Africa, 1992.
- [12] Vellando, P. On the resolution of the viscous incompressible flow for various SUPG finite element formulations, European congress on the Computational Methods in the Applied Sciences and Engineering – (ECCOMAS) 2000, Barcelona, 2000.
- [13] Wang, G. A fast and robust variant of the simple algorithm for finite-element simulations of incompressible flows, *Computational Fluid and Solid Mechanics*, 2, pp. 1014–1016, Elsevier, 2001.
- [14] Wang, G. A conservative equal order finite element method using the collocated GALERKIN approach, 8th AIAA/ASME Thermo-physics and heat transfer conference, St. Louis, Missouri, AIAA 2002-3206, 2001.
- [15] Wansophark, N. Enhancement of segregated finite element method with adaptive meshing technique for viscous incompressible thermal flow analysis, *Science Asia*. 29(2), pp. 155–162, 2003.
- [16] Xie, J. Equal-order segregated finite element formulation for flow and phase change problems, Doctoral Dissertations, University of Wisconsin, 2007.

This page intentionally left blank

III. Turbulent Flow Computations/Large Eddy Simulation/Direct Numerical Simulation

This page intentionally left blank

6 Time-accurate techniques for turbulent heat transfer analysis in complex geometries

Danesh K. Tafti

Department of Mechanical Engineering, Virginia Polytechnic Institute and State University, VA, USA

Abstract

With the large increase in computational power over the past two decades, time-dependent techniques in turbulent heat transfer are becoming more attractive and feasible. The chapter describes the numerical and theoretical background to enable the use of these methods in complex geometries.

Keywords: Heat exchangers, Large-eddy simulations (LES), Parallel computing, Turbomachinery, Unsteady CFD

6.1 Introduction

Heat transfer devices span a wide range of applications in the automotive, aerospace, electronics, power generation, and biomedical industries with characteristic length scales ranging from a few hundred microns to meters. The characteristic Reynolds numbers (Re) can range from low and largely viscous ($Re < 10$) to high and fully turbulent ($Re > 2,000$). While predictions in the laminar regime do not pose significant computational challenges, barring strong property variations, non-Newtonian flows, or significant body forces, predictions in the transitional and turbulent regime pose considerable challenges. In these flow regimes common techniques employed are direct numerical simulations (DNS), large-eddy simulations (LES), Reynolds-averaged Navier–Stokes (RANS) or unsteady RANS (URANS). DNS by definition requires that the full spectrum of turbulence down to the smallest dissipative scales be resolved in the calculation. As the Reynolds number increases beyond $O(10^3 - 10^4)$ in wall-bounded flows, resolving the full spectrum becomes exceedingly expensive, which limits the application of DNS. LES extends its practical range to Reynolds number of $O(10^4 - 10^5)$ by resolving scales only up to the inertial range and modeling the smaller scales via a subgrid stress model. RANS extends this range further to very high Reynolds numbers by modeling the full spectrum of turbulent scales and incorporating their time-averaged effect into the mean

momentum and energy balance. URANS is used in cases where external forcing is present at a given frequency or the flow has an intrinsic characteristic frequency (as in shedding of large-scale structures). By resolving only the large-scale time-dependent structures and modeling the smaller-scale turbulence, the required grid density is still much less than other time-dependent methods.

The high computational cost of DNS and LES on the one hand and the prediction uncertainty of RANS on the other has led to the development and application of hybrid RANS-LES methods, which have tried to combine the low cost of RANS with the superior accuracy of LES. Out of this family of hybrid techniques, detached Eddy simulation (DES) has gained popularity since it was proposed by Spalart et al. [1] in 1997 for external flows with massive separation. DES involves sensitization of a RANS model to grid length scales, thereby allowing it to function as a subgrid stress model in critical regions.

All the above methods are valid in the range of flows encountered in heat transfer devices. For example, RANS and URANS and maybe DES are more readily applicable to system-level simulations, whereas DNS and LES are more appropriate for component-level simulations.

The chapter focuses on the application of these techniques to complex turbulent flows highlighting the related theoretical and computational background from the author's research experience. It starts with the transformed conservation equations in general coordinate systems and their solution in a multiblock computational framework. The nonstaggered grid pressure equation is related to its staggered grid counterpart giving perspective to different approximations used in the literature and the formulation finally used. A section is also devoted to the discretization of the convection term and its effect on solution accuracy. This is followed by a section on subgrid modeling in LES and the DES method. Finally, the solution of linear systems and parallelization strategies on modern computer architectures are discussed. The chapter is closed by referring to some recent application of these techniques to complex heat transfer problems.

6.2 General Form of Conservative Equations

The continuum conservation of mass, momentum, and energy and the ideal gas equation of state are described by the following time-dependent, conservation equations (the superscript * is used to denote dimensional variables throughout this chapter):

Mass conservation:

$$\frac{\partial \rho^*}{\partial t^*} + \vec{\nabla} \cdot (\rho^* \vec{u}^*) = 0 \quad (1)$$

Momentum conservation:

$$\frac{\partial(\rho^* \vec{u}^*)}{\partial t^*} + \vec{\nabla} \cdot (\rho^* \vec{u}^* \vec{u}^*) = -\vec{\nabla} P^* + \vec{\nabla} \cdot \begin{pmatrix} \mu^* (\vec{\nabla} \vec{u}^* + \vec{\nabla} \vec{u}^{*T}) \\ -\frac{2}{3} \mu^* (\vec{\nabla} \cdot \vec{u}^*) \mathbf{I} \end{pmatrix} + \rho^* \vec{g}^* \quad (2)$$

Energy conservation:

$$\frac{\partial(\rho^* c_p^* T^*)}{\partial t^*} + \vec{\nabla} \cdot (\rho^* c_p^* \vec{u}^* T^*) = \vec{\nabla} \cdot (\kappa^* \vec{\nabla} T^*) + \left(\frac{\partial P^*}{\partial t^*} + \vec{u}^* \cdot \vec{\nabla} P^* \right) + \Phi^* + \rho^* \vec{g}^* \cdot \vec{u}^* \quad (3)$$

where

$$\Phi^* = \left[\mu^* (\vec{\nabla} \vec{u}^* + \vec{\nabla} \vec{u}^{*T}) - \frac{2}{3} \mu^* (\vec{\nabla} \cdot \vec{u}^*) \mathbf{I} \right] : \vec{\nabla} \vec{u}^*$$

is the viscous-dissipation term.

Equation of state:

$$\rho^* = \frac{P^*}{R^* T^*} \quad (4)$$

Targeting applications in free and forced convection with variable fluid properties, the equations are cast in nondimensional form by using the following parametrization:

$$\begin{aligned} \rho &= \frac{\rho^*}{\rho_{\text{ref}}^*} & \mu &= \frac{\mu^*}{\mu_{\text{ref}}^*} & \kappa &= \frac{\kappa^*}{\kappa_{\text{ref}}^*} & c_p &= \frac{c_p^*}{c_{p_ref}^*} \\ \vec{x} &= \frac{\vec{x}^*}{L_{\text{ref}}^*} & \vec{u} &= \frac{\vec{u}^*}{U_{\text{ref}}^*} & t &= \frac{t^* U_{\text{ref}}^*}{L_{\text{ref}}^*} & P &= \frac{P^* - P_{\text{ref}}^*}{\rho_{\text{ref}}^* U_{\text{ref}}^{*2}} & T &= \frac{T^* - T_{\text{ref}}^*}{T_o^*} \end{aligned}$$

The reference temperature, T_{ref}^* and pressure P_{ref}^* are used for calculating all reference property values, whereas T_o^* is the nondimensionalizing temperature scale. The dynamic viscosity μ^*/μ_{ref}^* and thermal conductivity $\kappa^*/\kappa_{\text{ref}}^*$ variations with temperature are represented by Sutherland's law for gases (White [2]). While these two quantities vary substantially with temperature, specific heat has a much weaker dependence. For example, viscosity and conductivity for air in the temperature range 240 to 960 K vary by more than 100%, whereas c_p^* only varies by 12%. Hence, assuming that $c_p^*/c_{p_ref}^* = 1$, and that $\vec{g}^* = g^* \vec{e}_y$, the *nondimensional* equations are written as:

Mass conservation:

$$\frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot (\rho \vec{u}) = 0 \quad (5)$$

Momentum conservation:

$$\begin{aligned} \frac{\partial(\rho \vec{u})}{\partial t} + \vec{\nabla} \cdot (\rho \vec{u} \vec{u}) &= -\vec{\nabla} P + \frac{1}{\text{Re}} \vec{\nabla} \cdot (\mu (\vec{\nabla} \vec{u} + \vec{\nabla} \vec{u}^T)) - \frac{2}{3} \mu (\vec{\nabla} \cdot \vec{u}) \mathbf{I} \\ &+ \frac{\text{Ra}}{\text{Re}^2 \text{Pr} (\Delta T)} \frac{\rho}{\text{Re}^2 \text{Pr} (\Delta T)} \vec{e}_y \end{aligned} \quad (6)$$

Energy conservation:

$$\frac{\partial(\rho T)}{\partial t} + \vec{\nabla} \cdot (\rho \vec{u} T) = \frac{1}{\text{RePr}} \vec{\nabla} \cdot (\kappa \vec{\nabla} T) + \text{Ec} \left(\frac{\partial P}{\partial t} + \vec{u} \cdot \vec{\nabla} P \right) + \frac{\text{Ec}}{\text{Re}} \Phi + \left(\frac{\text{Ec}}{\text{Fr}} \right) \rho \vec{e}_y \cdot \vec{u} \quad (7)$$

Equation of state:

$$\rho = \frac{\rho_{\text{ref}}^* U_{\text{ref}}^{*2} P + P_{\text{ref}}^*}{R^* \rho_{\text{ref}}^* (T \cdot T_o^* + T_{\text{ref}}^*)} \quad (8)$$

where ΔT in equation (6) is the characteristic nondimensional temperature difference that drives the buoyant flow,

$$\begin{aligned} \text{Re} &= \frac{\rho_{\text{ref}}^* U_{\text{ref}}^* L_{\text{ref}}^*}{\mu_{\text{ref}}^*}, \quad \text{Pr} = \frac{\mu_{\text{ref}}^* c_{p,\text{ref}}^*}{\kappa_{\text{ref}}^*}, \quad \text{Ra} = \frac{g^* \beta^* (\Delta T^*) L_{\text{ref}}^{*3}}{\alpha_{\text{ref}}^* \nu_{\text{ref}}^*} \\ \text{Ec} &= \frac{U_{\text{ref}}^{*2}}{c_{p,\text{ref}}^* T_o^*}, \quad \text{and} \quad \text{Fr} = \frac{U_{\text{ref}}^{*2}}{g^* L_{\text{ref}}^*} \end{aligned}$$

specify the Reynolds, Prandtl, Rayleigh, Eckert, and Froude number, respectively, and β^* is the thermal expansion coefficient. The reference velocity U_{ref}^* is selected based on the dominating convection force. In forced convection, the reference velocity is specified from the driving flow, e.g., the free-stream velocity in external flow. In the case of pure natural convection, the reference velocity is calculated from the characteristic length and the driving temperature difference, as defined by the following (dimensional) equation:

$$U_{\text{ref}}^* = \sqrt{g^* L_{\text{ref}}^* \beta^* \Delta T^*} \quad (9)$$

In equation (6), the nondimensional coefficient of the gravitational body force term, $\text{Ra}/(\text{Re}^2 \text{Pr})$, determines the gravitational effect on the flowfield. In pure natural convection, the term takes on a value of unity after substitution of the reference velocity (equation (9)), into the Reynolds number, whereas in forced convection, when $\text{Re} \gg \text{Ra}$, the coefficient would be negligibly small. In mixed convection, both the specified driving temperature difference and reference convective velocity determine the relative magnitude of the gravitational source term and hence of natural versus forced convection.

6.2.1 Incompressible constant property assumption

Low-speed incompressible flow with constant fluid properties results in a number of simplifications in the conservation equations. One of the main numerical consequences of this assumption is the decoupling of density from pressure and the

implicit assumption that pressure waves travel at infinite speed relative to the flow velocities (Mach number, $\text{Ma} \rightarrow 0$). The Eckert number, $\text{Ec} = \text{Ma}^2(\gamma - 1) \rightarrow 0$, and the pressure work term in equation (7) makes a negligible contribution to energy conservation and can be neglected. In addition it is also customary to neglect the viscous dissipation term ($\text{Ec}/\text{Re} \rightarrow 0$) and the potential energy term ($\text{Ec}/\text{Fr} \rightarrow 0$). However, it is noteworthy that for very low Reynolds number highly viscous flow conditions might exist at which the viscous dissipation term can no longer be neglected. After simplifications, the nondimensional equations take the form:

$$\rho = \frac{\rho^*}{\rho_{\text{ref}}^*} = 1; \quad \mu = \frac{\mu^*}{\mu_{\text{ref}}^*} = 1; \quad \kappa = \frac{\kappa^*}{\kappa_{\text{ref}}^*} = 1; \quad c_p = \frac{c_p^*}{c_{p,\text{ref}}^*} = 1$$

$$\vec{x} = \frac{\vec{x}^*}{L_{\text{ref}}^*}; \quad \vec{u} = \frac{\vec{u}^*}{U_{\text{ref}}^*}; \quad t = \frac{t^* U_{\text{ref}}^*}{L_{\text{ref}}^*}; \quad P = \frac{P^* - P_{\text{ref}}^*}{\rho_{\text{ref}}^* U_{\text{ref}}^{*2}} \quad T = \frac{T^* - T_{\text{ref}}^*}{T_o^*}$$

Mass conservation:

$$\nabla \cdot \vec{u} = 0 \quad (10)$$

Momentum conservation:

$$\frac{\partial \vec{u}}{\partial t} + \vec{\nabla} \cdot (\vec{u} \vec{u}) = -\vec{\nabla} P + \vec{\nabla} \cdot \left(\frac{1}{\text{Re}} \vec{\nabla} \vec{u} \right) + g \vec{e}_y \quad (11)$$

Energy conservation:

$$\frac{\partial T}{\partial t} + \vec{\nabla} \cdot (\vec{u} T) = \vec{\nabla} \cdot \left(\frac{1}{\text{Re} \text{Pr}} \vec{\nabla} T \right) \quad (12)$$

where $g = 1/\text{Fr} = g^* L_{\text{ref}}^* / U_{\text{ref}}^{*2}$.

For small temperature and density differences ($\rho^* = \rho_{\text{ref}}^* + \Delta\rho^*$) with other properties held constant, the Boussinesq approximation can be accommodated in equation (11) by replacing the last term with $g \vec{e}_y (1 - T)$ if $T_{\text{ref}}^* = T_0^*$.

Fully developed assumption

In a number of applications the geometry lends itself to considerable computational simplification by considering a single period of a repeating feature in space such as in an array of fins and in ducts with repeating roughness elements. This is done by invoking a fully-developed assumption in the flow direction, thus allowing simulation of a characteristic unit of the flow. This formulation was developed for steady flow by Patankar et al. [3] for both a constant heat flux and a constant temperature boundary condition. Because of the usefulness of this formulation for detailed analysis of flow physics and heat transfer with time-dependent techniques, it is repeated here for a constant heat flux boundary condition in the context of unsteady flow and heat transfer under conditions of a fixed pressure gradient and a developing flow field. Under these conditions, the wall friction velocity $u_\tau^* = \sqrt{\tau_{w,\text{eq}}^* / \rho^*}$ is

used as the reference velocity to nondimensionalize the governing equations. For a fully-developed flow a mean momentum balance in the flow direction (x) gives the following relationship:

$$U_{\text{ref}}^* = u_\tau^* = \sqrt{\tau_{w_{\text{eq}}}^*/\rho^*} = \sqrt{(-\Delta P_x^*/L_x^*)(D_H^*/4\rho^*)} \quad (13)$$

where $\tau_{w_{\text{eq}}}^*$ is an equivalent mean wall shear which also includes form losses in the domain, ΔP_x^* is the mean pressure drop in the flow direction across the computational domain length of L_x^* , and D_H^* is the hydraulic diameter. The characteristic temperature scale is defined as $T_o^* = q_w^* L_{\text{ref}}^*/\kappa^*$, where $-q_w^*$ is the applied wall heat flux. Since both pressure and temperature have a x -directional dependence, the assumed periodicity of the domain in the streamwise or x -direction requires the mean gradient of pressure and temperature to be isolated from the fluctuating periodic components as follows:

$$\begin{aligned} P^*(\vec{x}, t) &= P_{\text{ref}}^* - \beta^* x^* + p^*(\vec{x}, t) \\ T^*(\vec{x}, t) &= T_{\text{ref}}^* + \gamma^*(t) \cdot x^* + \theta^*(\vec{x}, t) \end{aligned} \quad (14)$$

where $\beta^* = -\Delta P_x^*/L_x^*$ is the mean pressure gradient, p^* is the periodic pressure fluctuations, γ^* is a time-dependent temperature gradient, and θ^* is the fluctuating or periodic temperature component. Nondimensionalizing equation (14) gives

$$\begin{aligned} P(\vec{x}, t) &= -\beta x + p(\vec{x}, t) \\ T(\vec{x}, t) &= \gamma(t)x + \theta(\vec{x}, t) \end{aligned} \quad (15)$$

where $\beta = 4/D_H$ from equation (13).

Substituting equation (15) in the nondimensional conservation equations (10)–(12) gives the following modified momentum and energy equations:

Momentum conservation:

$$\frac{\partial \vec{u}}{\partial t} + \vec{\nabla} \cdot (\vec{u}\vec{u}) = -\vec{\nabla}p + \vec{\nabla} \cdot \left(\frac{1}{\text{Re}} \vec{\nabla} \vec{u} \right) + g\vec{e}_y + \beta\vec{e}_x \quad (16)$$

Energy conservation:

$$x \frac{d\gamma}{dt} + \frac{\partial \theta}{\partial t} + \vec{\nabla} \cdot (\vec{u}\theta) = \vec{\nabla} \cdot \left(\frac{1}{\text{Re Pr}} \vec{\nabla} \theta \right) - \gamma u_x \quad (17)$$

with modified boundary conditions

$$\phi(x, t) = \phi(x + L_x, t), \phi = \vec{u}, p, \text{ and } \theta$$

and

$$-\vec{\nabla}\theta \cdot \vec{n} = -1 + \gamma \vec{e}_x \cdot \vec{n}$$

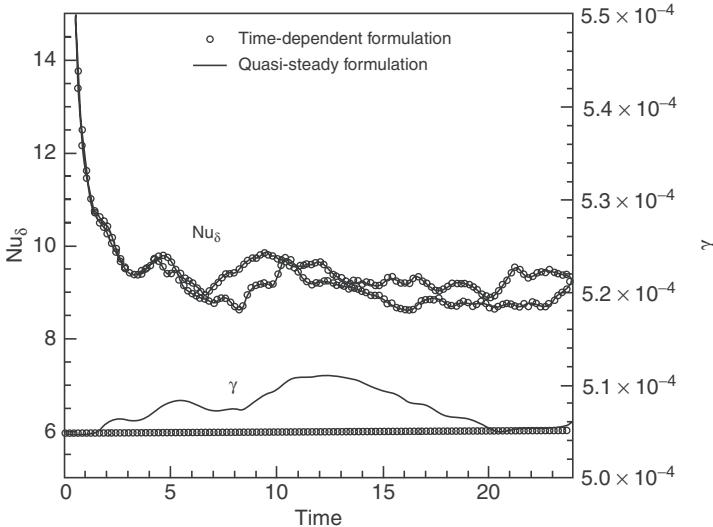


Figure 6.1. Time evolution of γ using the unsteady and quasi-steady procedure. The time-varying Nusselt number calculated on the two walls is identical between the two formulations.

The time evolution of γ is obtained by a global integration of equation (17). Under impermeable wall conditions

$$\Omega_V \frac{L_x}{2} \frac{d\gamma}{dt} + \int_{\Omega_V} \frac{\partial \theta}{\partial t} d\Omega_V = - \int_{\Omega_S} \frac{1}{\text{RePr}} (-1 + \gamma \vec{e}_x \cdot \vec{n}) d\Omega_S - \gamma Q_x L_x \quad (18)$$

where Ω_V is the fluid volume in the computational domain, Ω_S is the heat transfer surface area, and Q_x is the x -directional flow rate. The left-hand side represents the time-dependent terms and under assumed quasi-steady conditions the expression for γ is given as:

$$\gamma = \frac{\Omega_S}{\text{RePr} Q_x L_x} \quad (19)$$

which represents an instantaneous energy balance between the heat added to the domain at heat transfer surfaces and that removed volumetrically by the (sink) term in the energy equation ($-\gamma u_x$). The reduction of equation (18) to (19) uses $\int_{\Omega_S} \vec{e}_x \cdot \vec{n} d\Omega_S = 0$ for closed heat transfer surfaces.

Figure 6.1 plots the time variation of γ in the heat transfer calculations of turbulent channel flow at $\text{Re} = \text{Re}_\tau = 180$ (based on friction velocity u_τ and channel half-width δ). In one calculation, γ is calculated based on the time-dependent evolution using equation (18) and in the other it is based on the quasi-steady assumption given by equation (19). The temperature field is initiated at Time = 0 with constant heat flux boundary conditions at the channel walls after the turbulent velocity field

is fully developed. Whereas the quasi-steady formulation is affected by the instantaneous variations in the flow rate, the unsteady formulation exhibits near constant values of γ . In spite of these differences, the instantaneous average Nusselt numbers based on channel half-width on the two channel walls are identical between the two calculations. From these and other similar results, the quasi-steady assumption in calculating γ can be deemed adequate for most cases when there is no extreme driving unsteadiness in the applied wall heat flux or in the applied pressure gradient.

6.2.2 Modeling turbulence

For turbulent flows, the appropriate ensemble averaged equations for RANS or filtered equations for LES are constructed in physical space to obtain the Reynolds stress terms and the subgrid terms, respectively. For a RANS formulation the Reynolds stresses to be modeled take the form:

$$[\tau]; \tau_{ij} = \overline{u'_i u'_j} \quad (20)$$

whereas the subgrid stresses take the form:

$$[\tau] \tau_{ij} = \overline{\underline{\bar{u}_i \bar{u}_j}} - \overline{\bar{u}_i \bar{u}_j} + \overline{\underline{\bar{u}'_i \bar{u}'_j}} + \overline{\bar{u}'_i \bar{u}'_j} + \overline{\underline{\bar{u}''_i \bar{u}''_j}} \text{ Or } \tau_{ij} = \overline{\bar{u}_i \bar{u}_j} - \overline{\bar{u}_i \bar{u}_j} \quad (21)$$

where the overbar represents the Reynolds-averaging operator for RANS and the filtering operation for LES, respectively. In filtering the LES equations it is assumed that the filter is commutative, i.e., $\partial f / \partial x = \partial \bar{f} / \partial x$. Similarly, u' represents the turbulent fluctuating quantities for RANS and u'' the unresolved subgrid velocities. In equation (21) the underlined term is the Leonard stress term, which denotes interactions between the resolved scales contributing to subgrid terms, followed by the interaction between grid and subgrid scales, and between subgrid scales only. Within the framework of a linear isotropic eddy-viscosity model for incompressible flow, both stress tensors are modeled as:

$$\tau_{ij}^a = \tau_{ij} - \frac{1}{3} \delta_{ij} \tau_{kk} = -\frac{2}{\text{Re}_t} \bar{S}_{ij} \quad (22)$$

where $1/\text{Re}_t$ is the nondimensional turbulent eddy viscosity for RANS and the subgrid eddy viscosity for LES and $\bar{S}_{ij} = 1/2(\partial \bar{u}_i / \partial x_j + \partial \bar{u}_j / \partial x_i)$ or $[\bar{S}] = 1/2(\vec{\nabla} \bar{u} + \vec{\nabla} \bar{u}^T)$ is the mean or filtered strain rate, respectively. The energy equation is treated analogously to the momentum equations to give

$$\tau_j = -\frac{1}{\text{Pr}_t \text{Re}_t} \frac{\partial \bar{T}}{\partial x_j} = -\frac{1}{\text{Pr}_t \text{Re}_t} \vec{\nabla} \bar{T} \quad (23)$$

where Pr_t is the turbulent Prandtl number. For convenience, in the rest of the chapter the overbar notation is dropped in the rest of the chapter being mindful of the fact that the same set of equations are used for RANS and LES, which are ensemble averaged for RANS and filtered for LES.

6.3 Transformed Equations in Generalized Coordinate Systems

For generalization to complex geometries, equations (10–12) are mapped from physical coordinates (\vec{x}) to logical/computational coordinates ($\vec{\xi}$) by a boundary conforming transformation $\vec{x} = \vec{x}(\vec{\xi})$, where $\vec{x} = (x, y, z)$ and $\vec{\xi} = (\xi, \eta, \zeta)$. Based on the developments in Thompson et al. [4], a set of covariant basis vectors are defined which are tangents to the coordinate curves ξ_i

$$\vec{a}_i = \vec{x}_{\xi_i}, \quad i = 1, 2, 3$$

Also defined is the symmetric covariant metric tensor \vec{G} , whose elements are

$$g_{ij} = \vec{a}_i \cdot \vec{a}_j = g_{ji}$$

where $\sqrt{g} = \sqrt{\det |g|} = \vec{a}_1 \cdot (\vec{a}_2 \times \vec{a}_3)$ is the Jacobian of the transformation, which is the ratio of element volume in physical space to that in computational space.

Defined further are a set of contravariant vectors that are perpendicular to coordinate surfaces, $\xi_i = \text{const.}$,

$$\vec{a}^i = \vec{\nabla} \xi_i = \frac{1}{\sqrt{g}} (\vec{a}_j \times \vec{a}_k), \quad i = 1, 2, 3 \text{ and } (i, j, k) \text{ are cyclic}$$

where $\sqrt{g} \vec{a}^i$ is a measure of the area of the coordinate surface $\xi_i = \text{const.}$ The corresponding symmetric contravariant metric tensor with elements:

$$g^{ij} = \vec{a}^i \cdot \vec{a}^j = g^{ji}$$

is then defined.

Under these transformations the conservative divergence operator is defined as

$$\vec{\nabla} \cdot \vec{A} = \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi_i} (\sqrt{g} \vec{A} \cdot \vec{a}^i), \quad i = 1, 2, 3$$

the conservative gradient operator as

$$\vec{\nabla} \phi = \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi_i} (\sqrt{g} \vec{a}^i \phi), \quad i = 1, 2, 3$$

and the conservative Laplace operator as:

$$\vec{\nabla} \cdot \vec{\nabla} \phi = \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi_i} \left(\frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi_k} (\sqrt{g} g^{ik} \phi) \right), \quad i, k = 1, 2, 3$$

Finally, the transformed governing equations (10–12) are written as:

Mass conservation:

$$\frac{\partial}{\partial \xi_j} (\sqrt{g} U^j) = 0, \quad (24)$$

Momentum conservation:

$$\begin{aligned} \frac{\partial}{\partial t} (\sqrt{g} u_i) + \frac{\partial}{\partial \xi_j} (\sqrt{g} U^j u_i) &= - \frac{\partial}{\partial \xi_j} (\sqrt{g} (\vec{a}^j)_i P) \\ &\quad + \frac{\partial}{\partial \xi_j} \left(\left(\frac{1}{\text{Re}} + \frac{1}{\text{Re}_t} \right) \sqrt{g} g^{jk} \frac{\partial u_i}{\partial \xi_k} \right) + \sqrt{g} S_{u_i} \end{aligned} \quad (25)$$

Energy equation:

$$\frac{\partial}{\partial t} (\sqrt{g} T) + \frac{\partial}{\partial \xi_j} (\sqrt{g} U^j T) = \frac{\partial}{\partial \xi_j} \left(\left(\frac{1}{\text{PrRe}} + \frac{1}{\text{Pr}_t \text{Re}_t} \right) \sqrt{g} g^{jk} \frac{\partial T}{\partial \xi_k} \right) + \sqrt{g} S_T \quad (26)$$

Here $\sqrt{g} U^j = \sqrt{g} (\vec{a}^j)_k u_k$ is the contravariant flux vector where $(\vec{a}^j)_k$ denotes the k th element of the vector \vec{a}^j , and u_i the Cartesian velocity basis vector. S_{u_i} and S_T are additional source terms in the momentum and energy equation, respectively.

For the fully-developed formulation, while the mass continuity equation remains unchanged, the momentum and energy equation take the form:

Momentum conservation:

$$\begin{aligned} \frac{\partial}{\partial t} (\sqrt{g} u_i) + \frac{\partial}{\partial \xi_j} (\sqrt{g} U^j u_i) &= - \frac{\partial}{\partial \xi_j} (\sqrt{g} (\vec{a}^j)_i p) + \frac{\partial}{\partial \xi_j} \left(\left(\frac{1}{\text{Re}} + \frac{1}{\text{Re}_t} \right) \sqrt{g} g^{jk} \frac{\partial u_i}{\partial \xi_k} \right) \\ &\quad + \sqrt{g} \beta \vec{e}_x + \sqrt{g} S_{u_i} \end{aligned} \quad (27)$$

Energy equation:

$$\begin{aligned} \frac{\partial}{\partial t} (\sqrt{g} \theta) + \frac{\partial}{\partial \xi_j} (\sqrt{g} U^j \theta) &= \frac{\partial}{\partial \xi_j} \left(\left(\frac{1}{\text{PrRe}} + \frac{1}{\text{Pr}_t \text{Re}_t} \right) \sqrt{g} g^{jk} \frac{\partial \theta}{\partial \xi_k} \right) \\ &\quad - \sqrt{g} \gamma u_x - \sqrt{g} x \frac{\partial \gamma}{\partial t} + \sqrt{g} S_T \end{aligned} \quad (28)$$

where $\text{Re} = \text{Re}_t$ is the Reynolds number based on the friction velocity u_τ^* (Equation 13).

6.3.1 Source terms in rotating systems

Equations (24–26) are the base incompressible conservation equations in a generalized coordinate system. Additional terms due to body forces can be included in the source terms S_{u_i} and S_T . In noninertial systems as that encountered in the internal cooling of turbine blades additional Coriolis and centrifugal force terms are included in nondimensional form:

$$\sqrt{g} S_{u_i} = -2\sqrt{g} \text{Ro}_k u_m \epsilon_{ikm} - \sqrt{g} \epsilon_{ilm} \epsilon_{jlk} \text{Ro}_j \text{Ro}_m r_k \quad (29)$$

where Ro_i is the rotation number given by $\omega_i^* L_{\text{ref}}^* / U_{\text{ref}}^*$, ω_i^* is the component of angular velocity in the i -direction, r_k is the radial component from the axis of rotation in the k -direction, and ϵ_{jkl} is the permutation tensor. In the case of orthogonal rotation about one axis (say the z -axis; $\omega_1^* = \omega_2^* = 0$), equation (29) reduces to

$$\begin{aligned}\sqrt{g}S_{u_1} &= 2\sqrt{g}Ro_3u_2 + \sqrt{g}Ro_3^2r_1; & \sqrt{g}S_{u_2} &= -2\sqrt{g}Ro_3u_1 + \sqrt{g}Ro_3^2r_2 \\ \sqrt{g}S_{u_3} &= 0\end{aligned}\quad (30)$$

Coriolis forces have a preferential effect on turbulence, augmenting or attenuating it depending on the orientation of the rotational axis with respect to the primary strain rate in the flow. The preferential effect has a profound impact on heat transfer coefficients, and fluid temperatures can vary substantially in the domain making the density differences nontrivial. The density differences give rise to differential centrifugal forces which are aptly referred to as centrifugal buoyancy forces. This is usually accommodated by using the Boussinesq approximation $\Delta\rho^*/\rho_{\text{ref}}^* = -\Delta T^*/T_{\text{ref}}^*$ to give

$$\sqrt{g}S_{u_i} = -2\sqrt{g}Ro_k u_m \epsilon_{ikm} - \sqrt{g} \epsilon_{ilm} \epsilon_{jlk} Ro_j Ro_m r_k \left(1 - T \frac{T_0^*}{T_{\text{ref}}^*} \right) \quad (31)$$

where the second term in the parenthesis represents the centrifugal buoyancy term. For developing and fully-developed flow $T_{\text{ref}}^* = T_{in}^*$, and $T_0^* = (T_w^* - T_{in}^*)$ or $T_0^* = q_{\text{ref}}^* L_{\text{ref}}^* / \kappa^*$ for constant temperature and constant heat flux boundary conditions, respectively.

6.4 Computational Framework

The transformed equations can either be solved on a staggered or nonstaggered grid. Here the equations are discretized and solved on a nonstaggered grid using a finite-volume approach. The dependent variables u , v , w , P , or p and T or θ are calculated at the cell center, while the contravariant flux vector, which is representative of the normal flux to a coordinate surface, is defined and calculated at the cell faces, $\sqrt{g}U^i$ at $\xi_i = \text{const}$. Typically, the grid metrics are calculated using appropriate second-order central (SOC) difference approximations. If high-order accuracy is desired, high-order approximations are used to calculate the grid metrics. The covariant basis vectors $(\vec{a}_1, \vec{a}_2, \vec{a}_3)$, the Jacobian \sqrt{g} , the contravariant basis vectors $(\vec{a}^1, \vec{a}^2, \vec{a}^3)$, the diagonal elements of the covariant metric tensor (g^{11}, g^{22}, g^{33}) , and off-diagonal elements of the contravariant metric tensor (g^{12}, g^{23}, g^{13}) are calculated and stored at the cell center. Further, to calculate the contravariant fluxes at the cell faces the contravariant basis vectors and diagonal elements of the contravariant metric tensor are calculated and stored at the appropriate cell face (\vec{a}^i, g^{ii} at $\xi_i = \text{const}$).

The above equations are discretized and solved in a multiblock framework. This is accomplished by introducing an overlap region or ghost cells at interblock faces. Typically, the overlap region is dictated by the order of discretization used in the

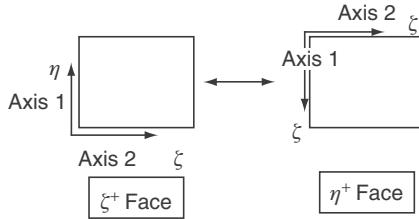


Figure 6.2. An example of unstructured block topology. East face (ξ^+) of one block exchanges information with the north face (η^+) of an adjoining block. The axes are also arbitrarily oriented with each other.

domain – for second-order discretizations, a single ghost cell region is needed at an interblock boundary; for second-order upwind approximations and fourth-order central approximations, the ghost cells increase to two and so on. Obviously, the expected increase in accuracy has to be balanced against the increased cost of the scheme itself, to which updating the overlap regions would add to, particularly in a parallel distributed computing framework.

While each block has a structured mesh, the interblock topology can either be structured or unstructured. In a structured topology, a ξ^+ block face can only interface with an adjoining ξ^- block face with coordinate axes perfectly aligned with each other on the two faces. In an unstructured topology, this restriction is relaxed and a ξ^+ block face can interface with any of the six adjoining block faces with any axes orientations. Figure 6.2 shows such an example in which the ξ^+ face of one block interfaces with η^+ face of another block.

In this instance, data transferred from one face to the other has to be sorted and realigned with the receiving coordinate system. The sorting operation is abstracted as:

$$\vec{A}|_{\xi^+} = [S] \vec{A}|_{\eta^+} \quad (32)$$

where \vec{A} is the data-vector (of size equal to the number of nodes in the surface) to be transferred from one face to the other and $[S]$ is a symmetric sorting matrix.

If the elements of \vec{A} have a directional dependence on the transformed coordinates such as components of the covariant vectors $\vec{x}_\xi, \vec{x}_\eta, \vec{x}_\zeta$ or other derivatives w.r.t. ξ, η , and ζ , an additional transformation is performed. For example in Figure 6.2, $\phi_\eta|_{\xi^+ \text{face}} = -\vec{\phi}_\xi|_{\eta^+ \text{face}}$. A general transformation is represented as:

$$\begin{bmatrix} \vec{\phi}_\xi \\ \vec{\phi}_\eta \\ \vec{\phi}_\zeta \end{bmatrix}_{\xi^+ \text{face}} = [S][T] \begin{bmatrix} \vec{\phi}_\xi \\ \vec{\phi}_\eta \\ \vec{\phi}_\zeta \end{bmatrix}_{\eta^+ \text{face}} \quad (33)$$

where $[T]$ is a 3×3 transformation matrix. For the example in Figure 6.2

$$[T] = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

An additional feature that is useful is the capability to handle nonconforming or nonmatching block interfaces, i.e., when there is no one-to-one match between node locations in two adjoining block interfaces. This capability is useful in zonal refining or coarsening of meshes from one block to another. In such cases, in addition to the sorting matrix, an interpolation matrix is constructed. Consider the η^+ face in Figure 6.2 to have n total nodes, which need to be transferred to ξ^+ face with m nodes. Then a further generalization of equation (32) can be written as:

$$\vec{A}|_{\xi^+ m \times 1} = [\mathfrak{I}]_{m \times n} [S]_{n \times n} \vec{A}|_{\eta^+ n \times 1} \quad (34)$$

where $[\mathfrak{I}]$ is the $m \times n$ interpolation matrix.

The interpolation operator should at a minimum have the same order of accuracy as the discrete approximations in the code. Bi-linear interpolation is the simplest giving close to second-order accuracy. Higher-order methods based on quadratic or cubic splines can also be used but at a higher cost [5]. During interpolation it is also important to conserve the mass, momentum, and energy fluxes moving across the interface. Some elements of this are explained in Section 5.3.

Figure 6.3 shows results from a zonal mesh used for the simulation of turbulent channel flow at $Re_\tau = 180$. The results are compared to the DNS data of Kim et al. [6] (KMM). Turbulent channel flow in spite of its geometrical simplicity embodies all the essential characteristics of wall-bounded turbulence and is an ideal test case for numerical developments in the simulation of turbulent flow. The zones are demarcated at $y^+ = 50$ and 100, with resolutions of $64 \times 14 \times 80$, $48 \times 9 \times 64$, and $32 \times 9 \times 48$ cells, respectively, for $L_x = 2\pi$ and $L_z = \pi$ and half-channel height for a total resolution of 226,304 cells on a six-block mesh. Because of the relatively small-scale turbulence present in the flow, it provides a stringent test case for the treatment of nonmatching boundaries. The results are compared to a single zone mesh of $64 \times 64 \times 64$ or 262,144 cells. Bi-linear interpolations are used at block boundaries, together with Dirichlet boundary conditions for pressure with the conservation of integrated mass flux and pressure gradients across zonal boundaries. The overall prediction accuracy is satisfactory for the zonal decomposition, but peak values are slightly underpredicted compared to the single zone mesh. In addition, at the zonal boundary $y^+ = 50$, the normal and shear stresses exhibit discontinuities in slope.

This can be attributed to the highly energetic small-scale eddies in this region and the resulting interpolation errors. Hence, unless high-order interpolations are used, nonmatching boundaries should be used in regions where the flow has relatively large-scale features that can be captured relatively well on the grid [7].

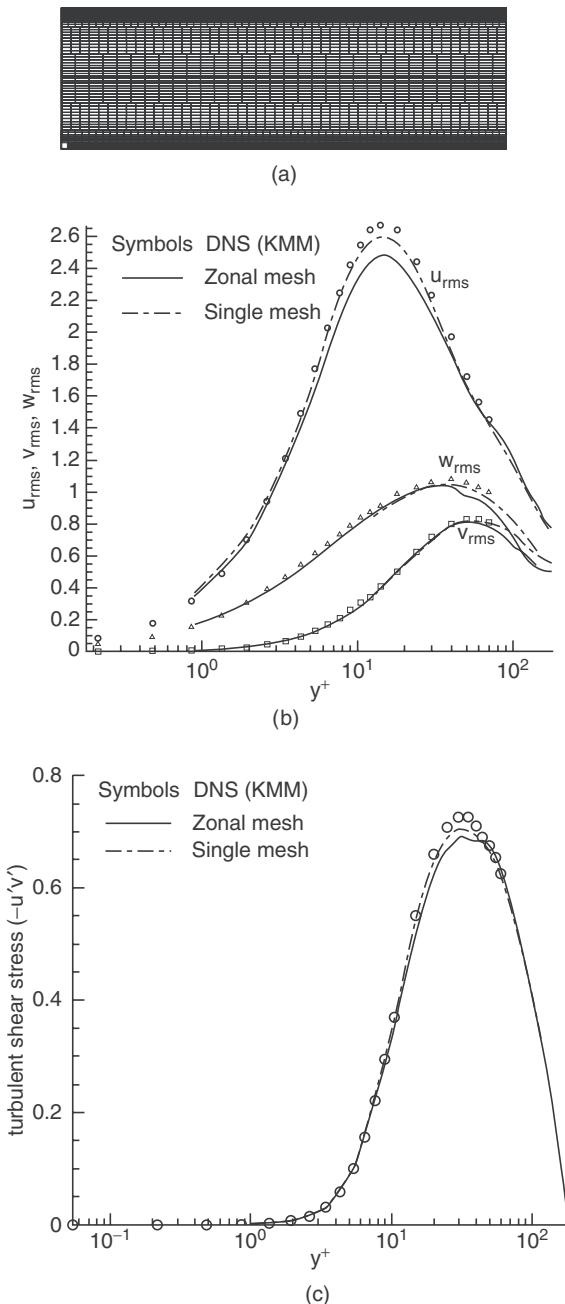


Figure 6.3. (a) Zonal mesh in $x-y$ plane; (b) rms quantities in wall coordinates; (c) turbulent shear stress distribution in wall coordinates. Zonal interfaces are at $y^+ = 50$ and 100.

6.5 Time-Integration Algorithm

A number of algorithms are available for time integration of the conservative equations. The most common methods for incompressible flow are the so-called “pressure” based methods, which solve for pressure by formulating a pressure equation using the mass continuity equation. Typically within this framework, the momentum equations are integrated in time using some guess for the pressure gradient (from a previous iterate or time step) to give an intermediate velocity field, which is then made to satisfy mass continuity by correcting with the calculated pressure field. Whereas for steady-flow calculations, algorithms based on the semi-implicit pressure-linked equations (SIMPLE) method are quite popular (Patankar [8]; Versteeg and Malalasekera [9]), for unsteady solutions the fractional step approach finds widespread use (Ferziger and Peric [10]). Whereas methods based on the SIMPLE algorithm solve the full set of nonlinear equations iteratively at each time step and hence technically do not have any time step restrictions, methods based on fractional step can be fully explicit with time step limited by both convective Courant–Frederichs–Levy (CFL) condition and viscous limits (Chorin [11]), semi-implicit in which only the viscous terms are treated implicitly (Kim and Moin [12]), and fully implicit in which both convection and viscous terms are treated implicitly (Choi and Moin [13]). Although time step limitations in the fractional step approach can be an impediment to efficient solution, for unsteady turbulent flows small time steps are indeed desirable to resolve the small turbulent time scales. In fact a balanced spatio-temporal discretization dictates a CFL of $O(\sim 1)$ and large time steps can have the same undesirable effect on the solution as a coarse spatial grid or a dissipative spatial discretization.

Here three algorithmic implementations within the fractional step approach are given. In the predictor step an intermediate velocity field \tilde{u}_i is computed explicitly, semi-implicitly, or implicitly by neglecting the effect of pressure gradient. In the explicit treatment, a second-order accurate in time Adams-Bashforth scheme is used to approximate both the convective and viscous terms. In the semi-implicit treatment, which is primarily used for low Reynolds number flows, the viscous terms are treated implicitly by a second-order Crank–Nicolson approximation. In the implicit treatment, both convective and viscous terms are treated by a Crank–Nicolson approximation. In all three methods the convection terms are linearized by using the contravariant fluxes at time level n or $n-1$. Additionally, built into the algorithm is the explicit conservation of surface-integrated mass, momentum, and energy at nonmatching block interfaces, which yield conserved variables $\langle \sqrt{g} U^j \rangle$, $\langle u_i^n \rangle$, and $\langle T^n \rangle$, respectively, where $\langle \rangle$ denotes conserved variables. The energy equation needs only the predictor step to obtain T^{n+1} and takes a form similar to the equations (35)–(37) below.

6.5.1 Predictor step

First, momentum and energy fluxes across nonmatching interfaces are conserved to obtain the conserved quantity $\langle u_i^n \rangle$ or $\langle T^n \rangle$.

Explicit formulation:

$$\frac{\tilde{u}_i - \langle u_i^n \rangle}{\Delta t} = \frac{3}{2} H_i^n - \frac{1}{2} H_i^{n-1} \quad (35)$$

where

$$H_i = -\frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi_j} (\langle \sqrt{g} U^j \rangle \langle u_i \rangle) + \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi_j} \left(\left(\frac{1}{\text{Re}} + \frac{1}{\text{Re}_t} \right) \sqrt{g} g^{jk} \frac{\partial \langle u_i \rangle}{\partial \xi_k} \right) + S_{u_i}$$

Semi-implicit formulation:

$$\frac{\tilde{u}_i - \langle u_i^n \rangle}{\Delta t} = \frac{3}{2} H_i^n - \frac{1}{2} H_i^{n-1} + \frac{1}{2} \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi_j} \left[\left(\frac{1}{\text{Re}} + \frac{1}{\text{Re}_t} \right) \sqrt{g} g^{jk} \left(\frac{\partial \tilde{u}_i}{\partial \xi_k} + \frac{\partial \langle u_i^n \rangle}{\partial \xi_k} \right) \right] \quad (36)$$

$$\text{where } H_i = -\frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi_j} (\langle \sqrt{g} U^j \rangle \langle u_i \rangle) + S_{u_i}$$

Implicit formulation:

$$\begin{aligned} \frac{\tilde{u}_i - \langle u_i^n \rangle}{\Delta t} &= -\frac{1}{2} \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi_j} \left(\overline{\sqrt{g} U^j} \tilde{u}_i + \langle \sqrt{g} U^j \rangle \langle u_i^n \rangle \right) \\ &\quad + \frac{1}{2} \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi_j} \left[\left(\frac{1}{\text{Re}} + \frac{1}{\text{Re}_t} \right) \sqrt{g} g^{jk} \left(\frac{\partial \tilde{u}_i}{\partial \xi_k} + \frac{\partial \langle u_i^n \rangle}{\partial \xi_k} \right) \right] + S_{u_i} \end{aligned} \quad (37)$$

where $\overline{\sqrt{g} U^j} = 2\sqrt{g} U^{jn} - \sqrt{g} U^{j^{n-1}}$ is a linearized estimation of $\sqrt{g} \tilde{U}^j$.

The superscript \sim denotes the intermediate field, $\langle \rangle$ denotes quantities based on conserved fluxes at nonmatching boundaries, and n the present time level. In the semi-implicit and implicit formulation, at inhomogeneous boundaries $\tilde{u}_i = u_i^{n+1}$ in the momentum equations and $\tilde{T} = T^{n+1}$ are used in the energy equation. The semi-implicit and implicit splitting of the momentum equations introduce an error of $O(\Delta t^2)$, which is of the same order as the temporal discretization.

6.5.2 Pressure formulation and corrector step

After obtaining the intermediate Cartesian velocity field, the continuity equation is used to derive the pressure equation, which is then solved to obtain the pressure field at time $(n+1)\Delta t$. The calculated pressure field is then used to correct the intermediate velocity field to satisfy the divergence-free condition. On a staggered grid, this procedure is straightforward since the intermediate velocities are available on the cell face of the pressure node and there is no ambiguity in satisfying discrete continuity. On the other hand, on a nonstaggered grid, interpolations of nodal velocities and pressure gradient terms are necessary at the cell face. This leads to a discrete pressure Laplacian, which does not exhibit good ellipticity as dictated by the continuous Laplace operator and leads to the classic problem of grid scale

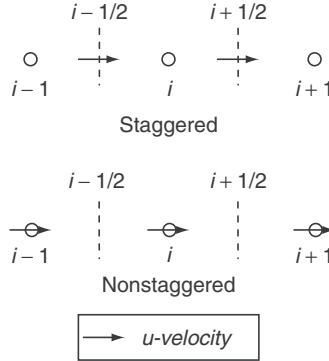


Figure 6.4. One-dimensional staggered and nonstaggered grids.

pressure oscillations (Tafti et al. [14]). The alleviation or elimination of this problem is an important issue on nonstaggered grids and is dealt with in some detail here. To understand it better, a general framework is developed in which the nonstaggered grid approximations are written in terms of their staggered grid counterparts. This framework is then used to elucidate on some of the techniques used in the literature to approach the staggered grid formulation.

Figure 6.4 shows a one-dimensional (1-D) Cartesian staggered and nonstaggered grid, with the locations of velocity and pressure. On a staggered grid, the nondimensional x -momentum equation is solved at $i \pm 1/2$ location as follows:

$$\left(\frac{\partial u}{\partial t} \right)_{i+1/2} = f(u) - \frac{p_{i+1} - p_i}{\Delta} \quad (38)$$

where $f(u)$ are the convection and viscous terms. Here, the pressure gradient term is approximated by a compact gradient operator. On the nonstaggered grid, the x -momentum equation with a SOC difference approximation for the pressure gradient at location i is discretized as follows:

$$\left(\frac{\partial u}{\partial t} \right)_i = f(u) - \frac{p_{i+1} - p_{i-1}}{2\Delta} \quad (39)$$

In this case, the pressure gradient is approximated by the noncompact form of the gradient operator. By using Taylor series approximations, we can rewrite this equation in a compact gradient operator form and an “error” term as follows:

$$\left(\frac{\partial u}{\partial t} \right)_i = f(u) - \frac{p_{i+1/2} - p_{i-1/2}}{\Delta} - \frac{\Delta^2}{8} \left(\frac{\partial^3 p}{\partial x^3} \right)_i + \dots \quad (40)$$

In essence, the nodal momentum equation on a non-staggered grid has an additional third derivative pressure term implicitly included in its formulation which can be

written as:

$$\left(\frac{\partial u}{\partial t}\right)^{\text{ns}} = \left(\frac{\partial u}{\partial t}\right)^s - \frac{\Delta^a}{\varepsilon_1} \left(\frac{\partial^{a+1} p}{\partial x^{a+1}}\right) + \dots \quad (41)$$

where the superscript ns refers to nonstaggered and s refers to staggered grid. In this case $a = 2$, and $\varepsilon_1 = 8$.

In deriving the pressure equation, the 1-D equivalent of the discrete continuity equation is of the form:

$$\frac{u_{i+1/2} - u_{i-1/2}}{\Delta} = 0 \quad \text{or} \quad \frac{1}{\Delta} \left[\left(\frac{\partial u}{\partial t}\right)_{i+1/2} - \left(\frac{\partial u}{\partial t}\right)_{i-1/2} \right] = 0 \quad (42)$$

In the staggered grid system, $u_{i+1/2}$ and $u_{i-1/2}$ are available at the cell faces from the x -momentum equation. However, in the nonstaggered grid, these velocities have to be interpolated to the cell faces. Using second-order symmetric interpolations for $(\partial u / \partial t)_{i+1/2}$ and equation (39) gives

$$\left(\frac{\partial u}{\partial t}\right)_{i+1/2} = \frac{1}{2}(f(u)_{i+1} + f(u)_i) - \frac{p_{i+2} + p_{i+1} - p_i - p_{i-1}}{4\Delta} \quad (43)$$

Using Taylor series expansions for $f(u)$ about the staggered node $i + 1/2$, and expressing the pressure gradient term in terms of its compact form ($p_{i+1} - p_i$), equation (43) can be written as:

$$\left(\frac{\partial u}{\partial t}\right)_{i+1/2} = f(u)_{i+1/2} - \frac{p_{i+1} - p_i}{\Delta} + \frac{\Delta^b}{\varepsilon_2} \left(\frac{\partial^b f(u)}{\partial x^b}\right)_{i+1/2} - \frac{\Delta^c}{\varepsilon_3} \left(\frac{\partial^{c+1} p}{\partial x^{c+1}}\right)_{i+1/2} + \dots \quad (44)$$

or in general

$$\left(\frac{\partial u}{\partial t}\right)^{\text{ns}} = \left(\frac{\partial u}{\partial t}\right)^s + \frac{\Delta^b}{\varepsilon_2} \left(\frac{\partial^b f(u)}{\partial x^b}\right) - \frac{\Delta^c}{\varepsilon_3} \left(\frac{\partial^{c+1} p}{\partial x^{c+1}}\right) + \dots \quad (45)$$

where $b = 2$, $\varepsilon_2 = 8$, $c = 2$, and $\varepsilon_3 = 4$. Equation (45) gives a general expression for the equivalent cell face momentum equation used on a nonstaggered grid in terms of its staggered grid counterpart to construct the pressure equation. The expression includes two additional terms, one of which is a result of the cell face interpolation, and the other, as in the expression for the nodal velocity, is a result of the noncompact form of the gradient operator. It is to be noted that the order of the interpolation error term can in fact be reduced by using high-order interpolations. Zang et al. [15] have used a QUICK type upwind biased approximation to interpolate the cell face values which gives $b = 3$ and $\varepsilon_2 = 24$ in equation (45). However, although the

magnitude of the error term introduced by the noncompact form of the pressure gradient can be reduced (Tafti et al. [14]), the order of the error always remains $O(\Delta^2)$ with the third derivative of pressure.

Finally, using the 2-D equivalent forms of equations (45) and (42) a modified pressure equation in two dimensions is derived as:

$$\begin{aligned} \nabla^2 p^s + \frac{\Delta_x^c}{\varepsilon_3} \left(\frac{\partial^{c+2} p}{\partial x^{c+2}} \right) + \frac{\Delta_y^c}{\varepsilon_3} \left(\frac{\partial^{c+2} p}{\partial y^{c+2}} \right) + \dots &= \frac{1}{\Delta t} (\nabla \cdot \tilde{u})^s \\ &+ \frac{\Delta_x^b}{\varepsilon_2} \left(\frac{\partial^{b+1} f(u, v)}{\partial x^{b+1}} \right) + \frac{\Delta_y^b}{\varepsilon_2} \left(\frac{\partial^{b+1} g(u, v)}{\partial y^{b+1}} \right) + \dots \end{aligned} \quad (46)$$

where $\nabla^2 p^s$ is a compact Laplacian and $(\nabla \cdot \tilde{u})^s$ is the compact divergence operator as obtained on a staggered grid. The constants b , c , ε_2 , and ε_3 have the same values as in equation (45). It is clear that the extra terms in the modified pressure equation introduce high frequency errors of $O(\Delta^2)$ into the pressure field, which contribute to the grid scale pressure oscillations. The elimination of the spurious pressure terms in equation (46) has been researched extensively. For example, Sotiropoulos and Abdallah [16] use regularizing terms in the pressure equation (of type $-\frac{\varepsilon}{4} \Delta^2 \frac{\partial^4 p}{\partial x^4}$, $0 \leq \varepsilon \leq 1$) which partially or exactly cancel the spurious terms in equation (46). In a 2-D-driven cavity they found that $\varepsilon = 0.1$ was sufficient to obtain smooth pressure fields. However, by arbitrarily “regularizing” the pressure equation, discrete continuity is not exactly satisfied. In order to overcome this deficiency, Tafti et al. [14] used equation (42) for discretizing the divergence operator in the continuity equation, and directionally biased the pressure gradient operator to obtain a discrete Laplacian that exhibited better ellipticity (L_{23} Laplacian in reference [14]). This Laplacian was found to eliminate pressure oscillations in a 2-D-driven cavity and also satisfied discrete continuity. For this approximation, equation (39) took the equivalent form:

$$\left(\frac{\partial u}{\partial t} \right)_i = f(u) - \frac{-p_{i+2} + 6p_{i+1} - 3p_i - 2p_{i-1}}{6\Delta} \quad (47)$$

resulting in $a = 2$, and $\varepsilon_1 = -24$ in equation (41) for the nodal momentum equation, $b = c = 2$, and $\varepsilon_2 = 8$ and $\varepsilon_3 = 12$ in equations (45) and (46). Hence, the directionally biased pressure gradient approximation was successful in reducing the error in the spurious pressure term by a factor of 3; however, as noted previously, the order of the error term remained the same. A limitation of this method is that the discrete Laplacian has a large stencil and hence makes implementation in a general coordinate frame quite cumbersome.

An additional criterion that needs to be satisfied when solving the pressure equation is the integrability constraint. It states that the discrete pressure Poisson

equation should satisfy Gauss divergence theorem, i.e.,

$$\int_{\Omega_V} D \cdot (Gp) d\Omega_V = \int_{\Omega_S} (Gp) \cdot \vec{n} d\Omega_S$$

and

$$\int_{\Omega_V} D \cdot \vec{u} d\Omega_V = \int_{\Omega_S} \vec{u} \cdot \vec{n} d\Omega_S \quad (48)$$

where D and G are the discrete divergence and gradient operators, \vec{u} is the nondimensional velocity vector, \vec{n} is the outward pointing unit normal at the surface Ω_S bounding the computational domain of volume Ω_V . This simply states that the discrete divergence operator when summed over the domain must reduce to a surface integral at the domain boundaries. This is always satisfied when a finite-volume form of the divergence operator is used in the form of equation (42). However, deviations from the effectively second-order divergence operator (high-order approximations) are difficult to reconcile and result in artificial mass sources/sinks in the computational domain and incomplete convergence. It is noteworthy that equation (48) places no restriction on the discrete gradient operator. Following from equation (48) is the additional condition that results from Gauss divergence theorem applied to the discrete pressure equation

$$\int_{\Omega_S} (Gp) \cdot \vec{n} d\Omega_S = \frac{1}{\Delta t} \int_{\Omega_S} \vec{u} \cdot \vec{n} d\Omega_S \quad (49)$$

Hence, if the intermediate incompressible velocity field in the fractional step method satisfies global mass conservation, which it should, then the LHS of equation (49) should sum to zero. The use of a zero gradient condition on pressure at inhomogeneous boundaries for known boundary velocities is consistent with this equation. The condition is also satisfied at homogeneous or periodic boundaries.

Under the requirements of eliminating grid scale oscillations on a nonstaggered grid and satisfying the integrability constraint and discrete continuity, one formulation that is quite effective is to interpolate the nodal Cartesian velocities to the cell faces (\vec{u}_i) to obtain the contravariant fluxes $\sqrt{g} U^j$. However, the interpolated contravariant cell face fluxes are treated as velocities on a staggered grid and are corrected by using the compact form of the pressure gradient. The net result of this formulation is that the spurious pressure terms in equations (45) and (46) are eliminated. However, the equivalent nodal momentum equation still carries the burden of the spurious third derivative pressure term (see equation (41)). In spite of this, the problem of grid scale oscillations is considerably reduced and in sufficiently well-resolved flows is non-existent. This is referred to as “current formulation” in the following discussion.

The different formulations are tested in turbulent channel flow at $Re_\tau = 180$ on a $64 \times 64 \times 64$ mesh with $\Delta z^+ = 8.8$ and $\Delta x^+ = 35.2$. The pressure spectrum is

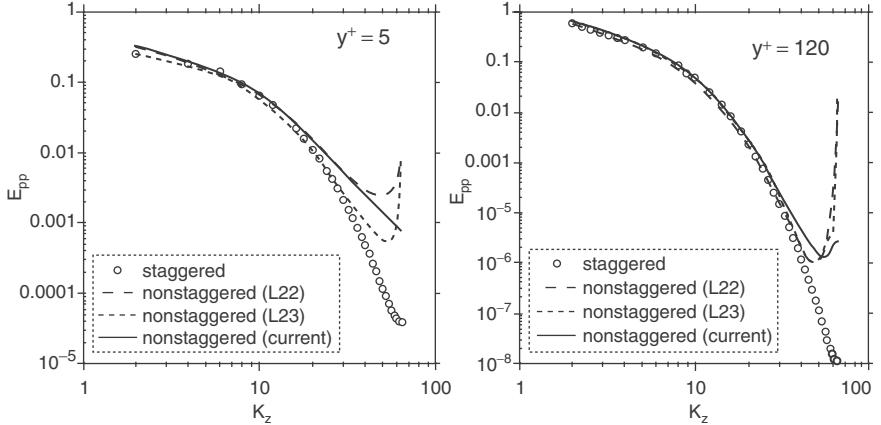


Figure 6.5. One-dimensional pressure spectra in turbulent channel flow at $\text{Re}_\tau = 180$ for different pressure formulations.

calculated near the wall at $y^+ = 5$ and toward the center of the channel at $y^+ = 120$ and plotted in Figure 6.5. Different nonstaggered grid formulations are compared to the staggered grid. L_{22} is the default noncompact form of the Laplacian constructed on a nonstaggered grid [14], L_{23} is the Laplacian constructed by Tafti et al. [14], and lastly the current formulation is shown. The pressure spectrum at the two locations show that while there is some energy accumulation in the near-grid scales for all the nonstaggered grid formulations, the current formulation prevents a large buildup. Between the default formulation L_{22} and the designed formulation L_{23} , L_{23} is somewhat better. At the same time, the presence of spurious oscillations near and at grid scales in pressure do not necessarily carry through to the power spectrum of the velocity components as seen in Figure 6.6. In fact it is only in the spanwise velocity that any accumulation of energy is present.

Hence, the following series of steps are used in the corrector step. First, the intermediate cell face contravariant fluxes are constructed as follows:

$$\sqrt{g} \tilde{U}^j = \sqrt{g} (\vec{a}^j)_i \tilde{u}_i \quad (50)$$

by interpolating the nodal velocities at the cell face. The contravariant flux is then conserved at nonmatching boundary faces to obtain $\langle \sqrt{g} \tilde{U}^j \rangle$. Then, the correction form of the nodal Cartesian velocities and cell face contravariant fluxes are written as:

$$u_i^{n+1} = \tilde{u}_i - \Delta t (\vec{a}^j)_i \left\langle \frac{\partial P^{n+1}}{\partial \xi_j} \right\rangle \quad (51)$$

$$\langle \sqrt{g} (U^i)^{n+1} \rangle = \langle \sqrt{g} \tilde{U}^i \rangle - \Delta t \sqrt{g} g^{ik} \left\langle \frac{\partial P^{n+1}}{\partial \xi_k} \right\rangle \quad (52)$$

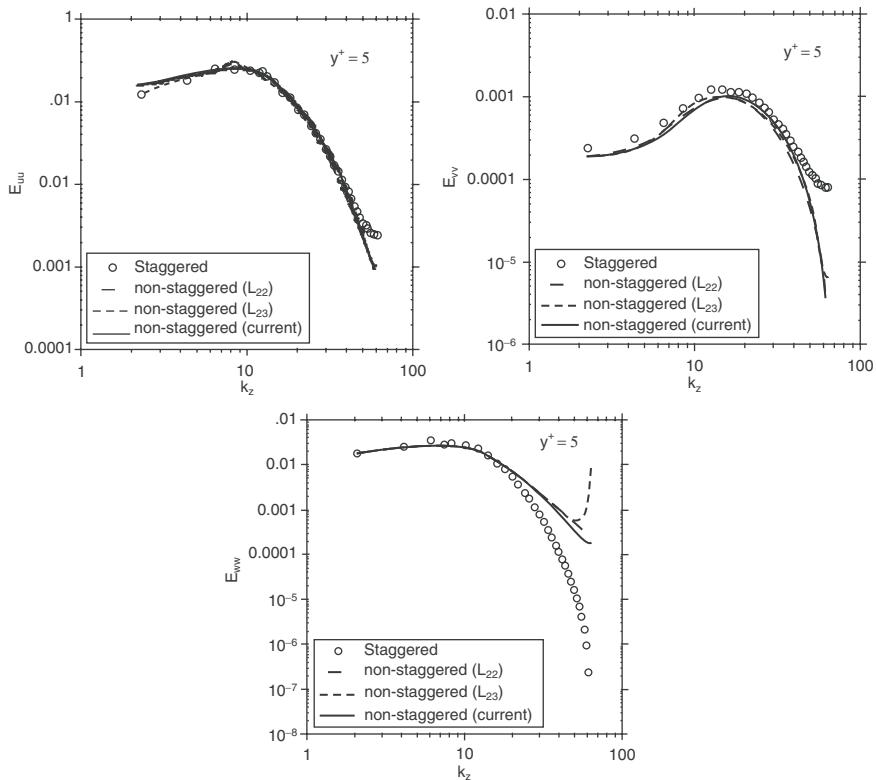


Figure 6.6. One-dimensional velocity spectra in turbulent channel flow at $\text{Re}_\tau = 180$ for different pressure formulations.

Using the finite-volume divergence operator in the mass continuity equation to satisfy the integrability constraint and substituting equation (52) for the cell face contravariant fluxes, the pressure equation takes the form:

$$\frac{\partial}{\partial \xi_j} \left(\sqrt{g} g^{jk} \left\langle \frac{\partial P^{n+1}}{\partial \xi_k} \right\rangle \right) = \frac{1}{\Delta t} \frac{\partial \langle \sqrt{g} \tilde{U}^j \rangle}{\partial \xi_j} \quad (53)$$

The calculated pressure field at level $n+1$ is then used to correct the nodal Cartesian velocities and the cell face contravariant fluxes using equations (51) and (52), respectively. The use of $\langle \rangle$ on the pressure gradient term implies the integral conservation of this quantity at nonmatching interface boundaries, during the solution of equation (53). Hence, $\sqrt{g}(U^i)^{n+1}$ is automatically conserved when the correction in equation (52) is applied to the conserved intermediate field.

6.5.3 Integral adjustments at nonmatching boundaries

To eliminate the buildup of small interpolation errors, integral adjustments at non-matching interfaces are used for conserved variables. At the beginning of each time step, momentum and energy fluxes are conserved at nonmatching boundaries by imposing the equality of mass or volume fluxes. With reference to Figure 6.2,

$$\sum_{\xi^+} \langle \sqrt{g} U^1 \rangle \phi = \sum_{\eta^+} \langle \sqrt{g} U^2 \rangle \phi \quad (54)$$

in the ξ - η direction, where $\phi = u_i$ or T are the interpolated values. The equality is used to obtain the conserved quantity $\langle \phi \rangle$.

The intermediate contravariant fluxes in equation (50) are first calculated from the interpolated intermediate velocities. The calculated fluxes, however, do not exactly satisfy the integral balance across the nonmatching interface. The conserved fluxes $\langle \sqrt{g} \tilde{U}^1 \rangle$ are obtained by imposing the integral balance across the two faces as in

$$\sum_{\xi^+} \sqrt{g} \tilde{U}^1 = \sum_{\eta^+} \sqrt{g} \tilde{U}^2 \quad (55)$$

There are two choices available in imposing equality of the conserved quantity in equation (54). One is proportional scaling and the other is an additive adjustment. Proportional scaling may have difficulties when the mean mass flux across the boundary is close to zero, whereas additive adjustments may smear weak recirculation zones at or near the interface. Since most of the time in a well-resolved flow, the interpolation error is small, the contravariant flux is adjusted linearly for the sake of robustness. However, for small values of $\sqrt{g} U^1$ in equation (54), even an additive adjustment has the potential of leading to instabilities, since to obtain $\langle \phi \rangle$ the total flux needs to be divided by this quantity.

For the pressure equation (53), the pressure at the ghost cell is first interpolated from the relevant boundary that serves as a boundary condition for the interior values. Since in incompressible flows the pressure gradient is the relevant quantity that drives the flow, this quantity is conserved at nonmatching interfacial boundaries. For each iteration in the pressure equation solver, the normal gradient of pressure is first evaluated and then integrated over the relevant interfacial boundary. Based on the difference of the two integral values, a constant value is then added/subtracted to the pressure at the ghost cells in order to guarantee the equality of the two integral values. Since this procedure is integrated into the linear solution of the pressure equation, it automatically guarantees an integral balance of the contravariant fluxes across nonmatching interfaces calculated through equation (52).

6.6 Discretization of Convection Terms

By far, from the numerical standpoint, the discretization of the nonlinear convective terms has the most profound effect on the accuracy with which turbulence is represented on a given mesh. An important aspect of turbulent flow simulations is the role of truncation and aliasing errors, which usually manifest themselves in the high wavenumbers of the resolved scales. Ideally, the discretization should have the capability of representing all the grid-resolved wavenumbers with accuracy, conserve quadratic quantities such as kinetic energy, have minimal aliasing errors, and be stable and robust at the same time. While truncation errors can be directly estimated from Taylor series expansions in physical space and modified wavenumber representations in wave space (Lele [17], Beudan and Moin [18]), the quantification of aliasing errors in time integration algorithms and their effect on the accuracy and numerical stability of the algorithm is not straightforward (Kravchenko and Moin [19]).

The most popular method for approximating convection terms has been an SOC difference discretization on staggered as well as nonstaggered grids. Although not perfect, it provides a reasonable and economical alternative to other high-order schemes. When the SOC scheme is applied to the divergence or conservative form of the convective terms in a finite-volume framework, it has the desirable property of conserving global kinetic energy. Although total kinetic energy is conserved, also present is the artificial redistribution of energy among the resolved scales. Related to this are aliasing errors that result from the spurious accumulation of unresolved numerical subgrid energy in the resolved scales due to the high-wavenumber nonlinear interactions of the velocity field. Early work in the meteorological and geophysical community (Phillips [20], Lilly [21], and Arakawa [22]) has addressed the role of aliasing in the development of nonlinear numerical instabilities, which led to the development of global energy-conserving finite-difference schemes. Global energy-conserving schemes require that the advection of a constant velocity field should not produce nor destroy global kinetic energy. According to Arakawa [22], the two conditions that a convective differencing scheme should satisfy to conserve quadratic quantities are:

$$\begin{aligned} A_{i'', j'', k'': i, j, k} &= -A_{i, j, k: i'', j'', k''} \\ A_{i, j, k: i, j, k} &= 0 \end{aligned} \quad (56)$$

The first equation requires that the discrete effect of one node on another should be equal and opposite, and the second condition requires that the node should not influence its own value. In 1-D, $A_E(i) = -A_W(i+1)$ and $A_P = 0$, where E, W, and P refer to a three point finite-difference stencil. For the finite-volume approximations, the first condition is always satisfied, whereas the second condition is linked to the satisfaction of discrete continuity in a form that will lead to it being zero. Hence, the energy-conserving property is inexplicably linked to the consistent solution of the pressure equation. It can be easily shown that the SOC scheme used with a consistent second-order construction of the divergence operator will satisfy the second condition. Any deviation from SOC to high-order approximations requires that the divergence operator in continuity and pressure be tailored to satisfy the

second condition. In fact most other conventional high-order schemes based on upwinding or central-differencing do not satisfy the second condition and special considerations have to be given to construct high-order schemes that are energy conserving (Morinishi et al. [23]).

On the other hand, in spite of not conserving global energy, high-order finite-difference schemes have been formulated and used to reduce the truncation errors associated with the finite-difference approximations. A number of such formulations have been used in the past based on high-order compact differences (Lele [17]), conventional high-order upwind biased approximations (Rai and Moin [24]), and fourth-order central differences. Accompanying the high-order convective schemes are also high-order approximations of the viscous terms and the pressure equation. Tafti et al. [25] have investigated a number of high-order schemes based on conservative and nonconservative formulations of the convection term combined with different pressure equation formulations.

A few of the high-order formulations were tested for the numerical simulations of turbulent channel flow at $Re_\tau = 180$ (based on wall friction velocity and channel half-width) using a staggered grid. Results from the 2^c22 , 5^c22 , $5^{nc}22$, and $5^{nc}44$ formulation are compared with the spectral simulations of KMM [6] and experiments of Kreplin and Eckelmann [26] (KE). Here c stands for the conservative or divergence form of the convection terms, which is discretized in a finite-volume framework, and nc stands for a nonconservative form of the convection term, which is discretized using finite-difference approximations. The first number denotes the order of approximation of the convection term, 2 stands for SOC, and 5 for fifth-order upwind biased approximations. The last two numbers denote the order of approximation of the divergence and gradient operator used to construct the pressure equation Laplacian: 22 refers to SOC approximations whereas 44 denotes fourth-order central approximations. The high-order methods are accompanied by sixth-order central interpolation operators and sixth-order central treatment of the diffusion terms. The $5^{nc}44$ formulation does not satisfy the integrability constraint in the inhomogeneous wall normal direction. Only the 2^c22 scheme is globally energy conserving.

The channel computational domain extends from 0 to 4π in x , -1 to 1 in y , and 0 to 2π in z , and calculations are performed with grids of $64 \times 64 \times 64$ and $128 \times 96 \times 128$. A semi-implicit formulation is used to obtain the intermediate velocity field. The pressure equation is solved by using 2-D FFTs in the homogeneous x - and z -directions with a line inversion in the y or inhomogeneous direction for each combination of wavenumbers. A nondimensional time step of $\Delta t = 2.5 \times 10^{-3}$ and 1.0×10^{-3} is used for the coarse and fine grid simulations, respectively. Velocity fields from previous simulations are used as initial conditions and each case is run for 10 nondimensional time units before any statistics are collected. The statistics are collected for additional 10 nondimensional time units. Figures 6.7 and 6.8 show some representative results for the coarse grid and fine grid simulations, respectively. The predicted mean streamwise velocity profiles in wall coordinates are compared to the theoretical laminar sublayer profile ($u^+ = y^+$) and the log-law profile of $u^+ = 2.5\ln(y^+) + 5.5$, together with predictions of kinetic energy

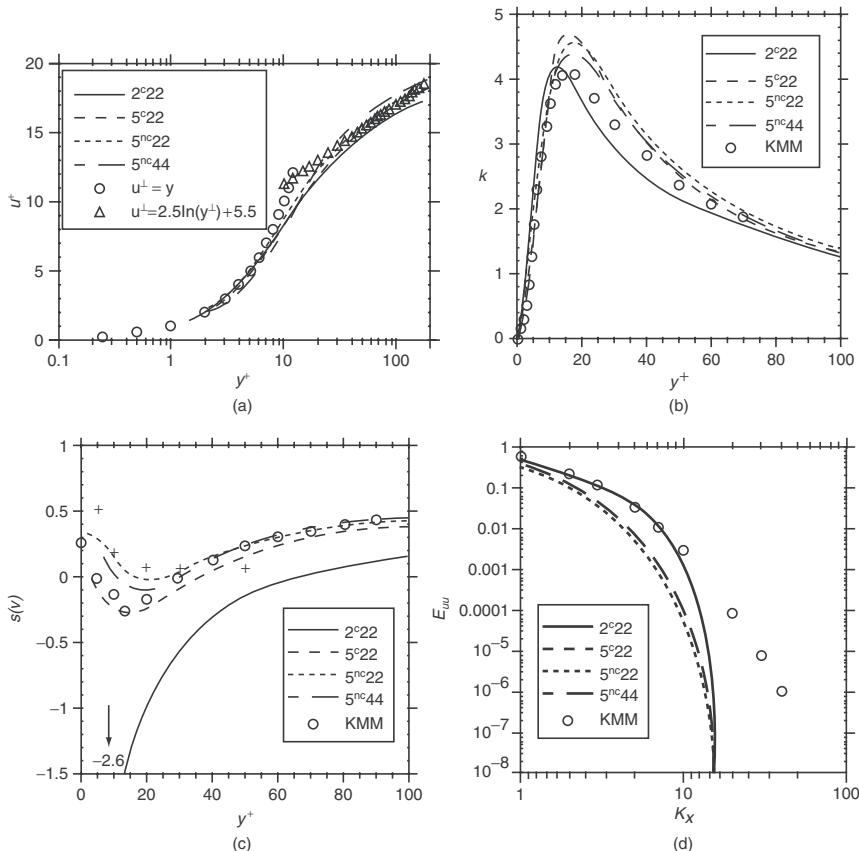


Figure 6.7. Comparison of different convection and pressure formulations in turbulent channel flow at $Re_\tau = 180$ at a resolution of 64^3 . (a) Mean velocity in wall coordinates; (b) turbulent kinetic energy distribution; (c) one-dimensional streamwise velocity power spectrum in z -direction; and (d) skewness of cross-stream velocity fluctuations.

distribution in wall coordinates, 1-D energy spectrum of the streamwise velocity (E_{uu}) at $y^+ = 5.6$, and third-order moment or skewness of the cross-stream velocity, which is very sensitive to the numerical approximation. The following observations can be made:

1. In both the coarse grid and the fine grid simulation the 2^c22 formulation computes the spectrum E_{uu} with the best accuracy. The high-order schemes 5^c and 5^{nc} both exhibit a dissipative behavior in representing the spectrum. The dissipative nature of the 5^c and 5^{nc} formulations is also evidenced by the overprediction of the core velocity and the overprediction and right-shift of the maximum turbulent kinetic energy profiles for the coarse grid simulations.

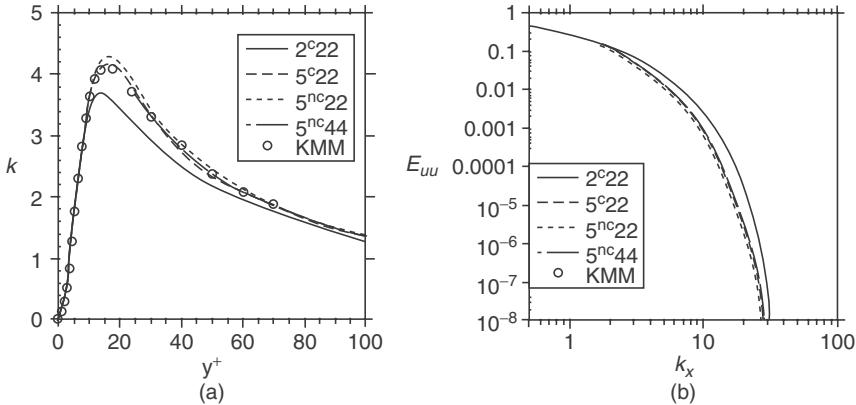


Figure 6.8. Comparison of different convection and pressure formulations in turbulent channel flow at $\text{Re}_\tau = 180$ at a resolution of $128 \times 96 \times 128$.
(a) Turbulent kinetic energy distribution and (b) one-dimensional streamwise velocity power spectrum in z -direction.

2. There are very small differences observed in the prediction capability of $5^{nc}44$ and $5^{nc}22$, in spite of the former not satisfying the integrability constraint (equations (48) and (49)). All of them predict the third moment or skewness with much better fidelity than the second-order scheme.
3. As the grid resolution increases to $128 \times 96 \times 128$, the high-order upwind schemes exhibit better accuracy in predicting the turbulent kinetic energy distribution compared to the 2^c22 formulation. However, the 2^c22 formulation is better at predicting the spectral energy distribution.

It can be surmised from these results that high-order upwind schemes in spite of their dissipative nature are still very usable and at high resolution in the DNS limit give more accurate results of turbulent statistics than the SOC difference scheme. However, they are computationally expensive because of the larger directionally dependent finite-difference stencils.

In complex CFD problems with complex grids, there is often the temptation to use low-order schemes such as second-order upwinding or the third-order upwind biased approximation (QUICK) or other monotonicity preserving schemes because of their relative stability and lower computational cost. While these schemes are useful in steady RANS calculations, they can lead to erroneous results in LES and DES. To elucidate on this, a normalized variable diagram (NVD) (Leonard [27]) is used to construct the total variation diminishing (TVD) schemes below. The NVD is constructed by defining the upstream and downstream directions based on the cell face flux direction as in Figure 6.9.

Then the normalized variable at any point is defined as:

$$\tilde{\phi} = \frac{\phi - \phi_u}{\phi_d - \phi_u}$$

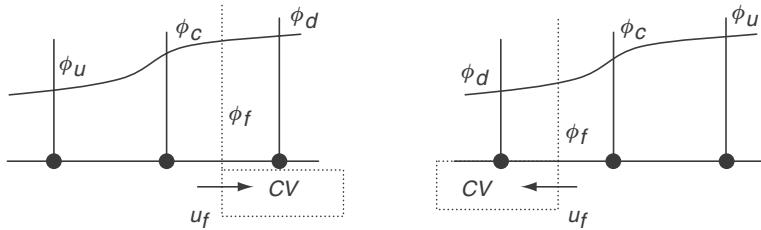


Figure 6.9. Nomenclature used to construct the NVD.

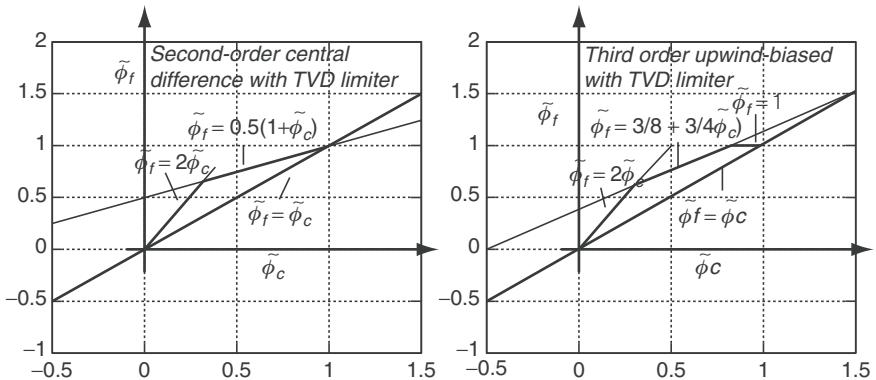


Figure 6.10. NVD diagram for TVD-limited second-order central scheme and the QUICK scheme.

where \sim denotes the normalized variable. For example, an SOC difference approximation at the cell face at time step $n\Delta t \phi_f^n = (\phi_c^n + \phi_d^n)/2$ is written in NVD coordinates as $\tilde{\phi}_f^n = (1 + \tilde{\phi}_C^n)/2$. The equivalent TVD limiter in NVD coordinates is given by:

$$\begin{aligned} \tilde{\phi}_f^n &= \tilde{\phi}_c^n && \text{for } \tilde{\phi}_c^n \leq 0 \text{ and } \tilde{\phi}_c^n \geq 1 \\ \tilde{\phi}_C^n &\leq \tilde{\phi}_f^n \leq \min[2\tilde{\phi}_C^n, 1] && \text{for } 0 \leq \tilde{\phi}_c^n \leq 1 \end{aligned} \quad (57)$$

Effectively, the first condition uses a first-order upwind approximation if ϕ_C^n is not bounded by ϕ_u^n and ϕ_d^n . Within the bounds established in the first condition, the second condition further limits the face value to lie between ϕ_C^n and the minimum of ϕ_d^n and $\phi_c^n + (\phi_c^n - \phi_u^n)$. The limiter applied together with the SOC scheme is illustrated in Figure 6.10 on the NVD. Similarly, the QUICK scheme in normalized coordinates is expressed as $\tilde{\phi}_f^n = 3/8 + 3/4\tilde{\phi}_C^n$.

In contrast to the TVD limiter, which is applied to the convected variable at $n\Delta t$ and hence can be too restrictive, Thuburn [28] developed a multidimensional flux limiter based on the less restrictive universal limiter (UL) proposed by Leonard [27].

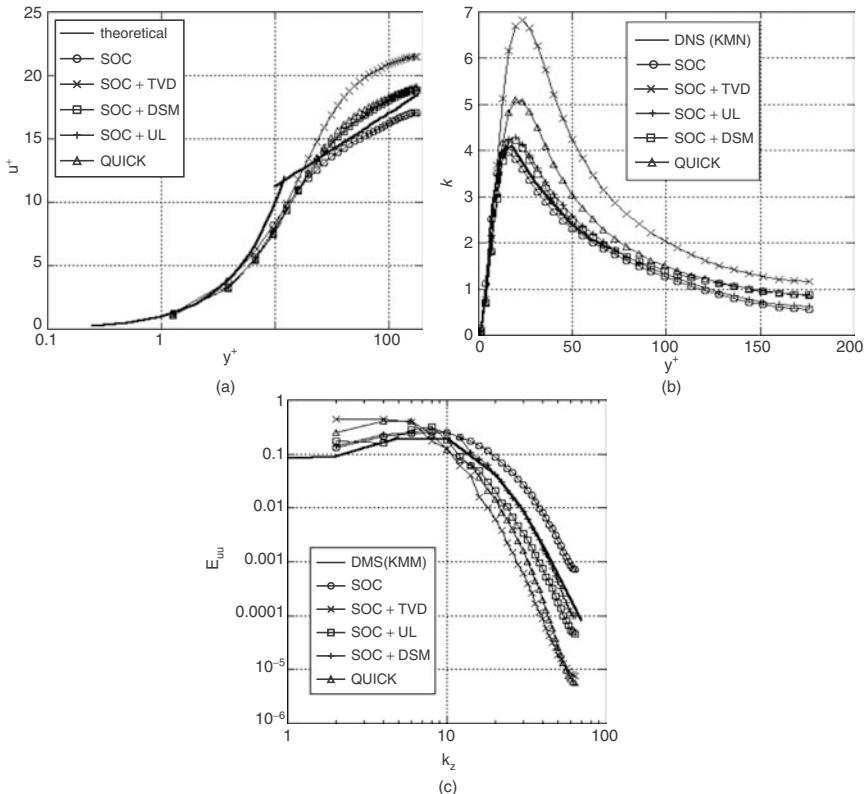


Figure 6.11. Prediction capability of different convective schemes in turbulent channel flow at $Re_\tau = 180$. (a) Mean velocity profile in wall coordinates; (b) turbulent kinetic energy distribution; and (c) one-dimensional energy spectrum of streamwise velocity in the z -direction.

In this scheme, the intermediate velocities are first calculated with the base approximations and then checked for monotonicity in a multidimensional framework. By not applying the monotonicity preserving criteria at time level n , the universal limiter is much less restrictive than the TVD limiter. The effect of different schemes is tested in turbulent channel flow at $Re_\tau = 180$ with streamwise grid spacing $\Delta x^+ = 35.34$, spanwise $\Delta z^+ = 8.84$, and cross-stream $0.63 \leq \Delta y^+ \leq 8.11$ for a domain of $4\pi \times \pi \times 2$ in x , z and y , respectively and plotted in Figure 6.11. The results are compared with DNS data of KMM. In fully developed channel flow since the flow velocity adjusts to the constant pressure gradient in reaction to the simulated wall friction, core velocities larger than theory are indicative of a dissipative scheme. This is also typically accompanied by high turbulent kinetic energy with the maximum shifted toward the core of the channel. The energy spectrum also provides additional evidence of the spectral properties of the scheme. Based

on these criteria, out of all the schemes the TVD limiter is the most dissipative even when used with the nondissipative SOC scheme. The QUICK scheme is not as dissipative as the TVD limiter, but in spite of having a higher accuracy than the SOC scheme it fails to give accurate results. On the other hand, the universal limiter behaves very much like the dynamic Smagorinsky model [29] (DSM) and seems to be a viable alternative for turbulent flow simulations. The use of minimal numerical dissipation as in the SOC-UL method to substitute for explicit subgrid modeling as a means for performing LES is referred to as implicit LES or ILES in the literature (Margolin et al. [30]; Grinstein et al. [31]).

In summary, discrete approximations used for the nonlinear convection terms have a large bearing on the accurate prediction of turbulence in time-dependent methods based on DNS, LES, and DES. The SOC method used in its conservative form is a good compromise for complex geometries between overly dissipative low-order (second- and third-order upwind scheme) and the extra cost and complexity of high-order methods.

6.7 Large-Eddy Simulations and Subgrid Modeling

Central to the large-eddy simulation of turbulence is the filtering operation and modeling of the resulting subgrid stresses. In generic terms the filtering operation is defined by a low-pass 3-D filter in physical space:

$$\bar{f}(x_i, t) = \int_{-\infty}^{+\infty} \prod_{i=1}^3 \bar{G}_i(x_i - x'_i) f(x'_i, t) dx'_i \quad (58)$$

In any discrete simulation by definition there is an implicit filter inherently applied to the calculation. In physical space with finite-difference/finite-element methods the implicit filter is referred to as the top hat filter because of its shape and is given by:

$$\begin{aligned} \bar{G}_i(x_i - x'_i) &= \frac{1}{\Delta x_i}; & |x_i - x'_i| &\leq \overline{\Delta x_i}/2 \\ &= 0; & \text{otherwise} \end{aligned} \quad (59)$$

where $\overline{\Delta x_i}$ is the grid spacing. Similarly in spectral simulations the filter width in wavenumber space is implicitly decided by the number of fourier modes used in the calculation and is referred to as a sharp cut-off filter and is given by:

$$\begin{aligned} G_i(k_i) &= 1; & k_i &\leq k_c \\ &= 0; & \text{otherwise} \end{aligned} \quad (60)$$

where k_c is the cut-off wavenumber. In contrast, the top hat filter is not sharp in wavenumber space but smooth such that it partially filters even the high wavenumbers resolved on the grid (Sagaut and Meneveau [32]). Another smooth filter that has found some use for explicit filtering is the Gaussian filter. There has been much debate and research on the properties of these filters and whether explicit filtering

with a filter width larger than that of the implicit grid spacing should be used in finite-difference/finite-volume methods. Since finite-difference methods inevitably result in some corruption of the high wavenumbers of the resolved field which are also used to model the subgrid stresses particularly in models that assume a similarity between the high wavenumber resolved field and the subgrid field, numerical inaccuracies and modeling errors become inextricably linked together. Hence the argument for explicit filtering is to break this link by using a twice as fine grid and then explicitly filtering out the numerically corrupted scales and using only the 2Δ filtered scales to model the subgrid stresses. However, the cost of the simulation more than doubles (in 1-D) and studies have shown that calculations on the twice as fine grid without explicit 2Δ filtering give better predictions of turbulent quantities.

In the past, efforts on subgrid modeling focused on calculating the Leonard terms and on modeling the individual terms in equation (21) (Leonard [33]; Clark et al. [34]). However this effort was largely abandoned in favor of modeling all the subgrid terms together. There are a number of subgrid-scale models varying in complexity from eddy viscosity to one equation models. Reviews can be found in Rogallo and Moin [35], Ferziger [36], Lesieur and Metais [37], and in the text by Saugat and Meneveau [32]. The most widely used closure model, suggested by Smagorinsky [38], is based on the eddy-viscosity approximation in which the subgrid-scale stresses are related to the strain rate tensor of the resolved field through an eddy viscosity (equation (22)). The eddy viscosity is computed from the resolved strain rate magnitude and a characteristic length scale (equation 61). The length scale is assumed to be proportional to the filter width via a Smagorinsky constant, which has assumed values anywhere from $C_s = 0.06$ to 0.23 in past studies.

$$\frac{1}{Re_t} = C_s^2 (\sqrt{g})^{2/3} |\bar{S}| \quad (61)$$

yielding

$$\tau_{ij}^a = -2C_s^2 (\sqrt{g})^{2/3} |\bar{S}| \bar{S}_{ij} = -2C \beta_{ij} \quad (62)$$

where $C = C_s^2$, $|S| = \sqrt{2\bar{S}_{ij}\bar{S}_{ij}}$ is the magnitude of the strain rate tensor that provides the nondimensional time scale of subgrid turbulence. Germano et al. [29] proposed a dynamic procedure for the computation of the Smagorinsky constant. The dynamic procedure utilizes information from the resolved high wavenumber scales by applying a test filter in addition to the grid filter to obtain the model constant.

Central to the dynamic procedure is the application of a test filter denoted by \hat{G} to the filtered governing equations (10) and (11) with the characteristic length scale of \hat{G} being larger than that of the grid filter, \bar{G} (usually $\hat{\Delta} = 2\Delta$). In computational coordinates the filter is written as:

$$\begin{aligned} \hat{G}(\xi_i - \xi'_i) &= \frac{1}{2\Delta\xi_i}; \quad |\xi_i - \xi'_i| \leq \Delta\xi_i; \quad i = 1, 2, 3 \\ &= 0; \quad \text{otherwise} \end{aligned} \quad (63)$$

and the filtered quantity as:

$$\hat{\bar{f}}(\xi_i) = \int_{-\Delta\xi_i}^{+\Delta\xi_i} \prod_{i=1}^3 \frac{1}{2\Delta\xi_i} \bar{f}(\xi'_i) d\xi'_i \quad (64)$$

Using Simpson's rule for a 1-D integration (say; $i = 2$ or η -direction)

$$\hat{\bar{f}}(\eta) = \frac{1}{6}(\bar{f}_{j-1} + 2\bar{f}_j + \bar{f}_{j+1}) \quad (65)$$

Using a five-point integration rule

$$\hat{\bar{f}}(\eta) = \frac{1}{2} \left(\frac{1}{6}(\bar{f}_{j-1} + 4\bar{f}_{j-1/2} + 2\bar{f}_j + 4\bar{f}_{j+1/2} + \bar{f}_{j+1}) \right) \quad (66)$$

Using a second-order interpolation at the cell faces in equation (66) leads to the trapezoidal top hat test filter

$$\hat{\bar{f}}(\eta) = \frac{1}{4}(\bar{f}_{j+1} + 2\bar{f}_j + \bar{f}_{j-1}) \quad (67)$$

From the development above it is straightforward to extend the filtering operation to nonuniform grids and to inhomogeneous boundaries. For example, at a wall boundary a simple extension of the method above will give:

$$\hat{\bar{f}}_2(\eta) = \frac{1}{4}(2\bar{f}_1 + \bar{f}_2 + \bar{f}_3) \quad (68)$$

where 1 denotes the boundary node and node 2 is at $\bar{\Delta}/2$ from the boundary. The one-sided filter now has a complex transfer function and introduces phase errors in the filtering operation. Najjar and Tafti [39] studied the effect of inhomogeneous filtering in turbulent channel flow and found that the local modification to the test filter had an effect on the distribution of the dynamic constant and subsequently the turbulent viscosity at the first few nodes near the boundary.

Using the same methodology a number of "high-order" filters can be constructed (Najjar and Tafti [40]). The high-order filters filter out a smaller percentage of the high wavenumber resolved scales and approach the behavior of a cut-off filter. It was determined by Najjar and Tafti [41], that the trapezoidal filter yielded more consistent results with finite-difference approximations.

Since the grid filter is applied to the conservation equations in physical space, for consistency the test filtering operations are transformed from computational to

physical space. This is accomplished by the following transformations:

$$\hat{\hat{f}}_x = \frac{1}{\Delta x} \left[x_\xi \hat{\hat{f}}(\xi) + x_\eta \hat{\hat{f}}(\eta) + x_\zeta \hat{\hat{f}}(\zeta) \right] \quad (69)$$

$$\hat{\hat{f}}_{xy} = \frac{1}{\Delta y} \left[y_\xi \hat{\hat{f}}_x(\xi) + y_\eta \hat{\hat{f}}_x(\eta) + y_\zeta \hat{\hat{f}}_x(\zeta) \right] \quad (70)$$

$$\hat{\hat{f}}_{xyz} = \frac{1}{\Delta z} \left[z_\xi \hat{\hat{f}}_{xy}(\xi) + z_\eta \hat{\hat{f}}_{xy}(\eta) + z_\zeta \hat{\hat{f}}_{xy}(\zeta) \right] \quad (71)$$

to give the filtered variable in physical space. Contrary to the formulation of the subgrid terms and consequent test filtering in physical space, Jordan [42] formulated the subgrid stresses in computational space, i.e., the grid filtering operation is applied to the transformed equations.

The test filtered subtest scales that include part of the grid-resolved scales and the subgrid scales $T_{ij} = \widehat{\bar{u}_i \bar{u}_j} - \widehat{\bar{u}_i} \widehat{\bar{u}_j}$ are modeled as:

$$T_{ij}^a = -2C_s^2(\sigma^2(\sqrt{2})^{2/3})|\hat{s}| \hat{s}_{ij} = -2C\alpha_{ij} \quad (72)$$

where σ is the ratio of the test filter width to the grid filter width. The most common value used in the literature is 2, which is consistent with a sharp cut-off filter. For the trapezoidal filter, Najjar and Tafti [41] obtained a value of $\sqrt{6}$

The resolved turbulent stresses representing the energy scales between the test and the grid filters, L_{ij} , are then calculated and related to the subtest (T_{ij}) and subgrid (τ_{ij}) stresses through the identity

$$L_{ij} = T_{ij} - \hat{\tau}_{ij}, \quad \text{where } L_{ij} = \widehat{\bar{u}_i \bar{u}_j} - \widehat{\bar{u}_i} \widehat{\bar{u}_j} \quad (73)$$

to give

$$L_{ij}^a = L_{ij} - \frac{1}{3}\sigma_{ij}L_{kk} = -2C\alpha_{ij} + 2\widehat{C}\beta_{ij} \quad (74)$$

Equation (74) corresponds to a set of five independent equations that cannot be solved explicitly for C , which appears in the test filtering operation. A simplification arises if C is assumed only to be a function of time and the direction where no explicit test filtering is applied (say the y -direction), i.e., $C = C(y, t)$, then C can be extracted from the filtering operation resulting in $\widehat{C}\beta_{ij} = C\widehat{\beta}_{ij}$. Using a least-square minimization procedure (Lilly [43]) for flows with at least one homogeneous direction, the following closed form for the dynamic subgrid-scale model constant, C , is obtained:

$$C(y, t) = -\frac{1}{2} \frac{\langle L_{ij}^a(\alpha_{ij} - \widehat{\beta}_{ij}) \rangle}{2\langle (\alpha_{mn} - \widehat{\beta}_{mn})(\alpha_{mn} - \widehat{\beta}_{mn}) \rangle} \quad (75)$$

where $\langle \rangle$ represents an averaging operation in the homogeneous directions. Equation (75) has been extensively used in LES computations of various turbulent flows.

The averaging procedure applied in equation (75) represents a smoothing operator to damp the large local fluctuations in C . In complex geometries however, there are no homogeneous directions and the test filter is often applied in all three directions and C has to be computed locally. The simplest approximation is to remove C from the filtering operation in equation (74) and then apply some smoothing to the local values of C based on zonal averaging or application of a smoothing filter not very different from the test filter. In addition to the smoothing, some bounds are usually placed on C or $1/\text{Re}_t$. The dynamic procedure is capable of predicting $C < 0$ or backscatter of energy. To permit backscatter, but to prevent negative effective viscosities and the ensuing numerical instability, $-1/\text{Re}_t$ is limited by $1/\text{Re}$. However, usage in complex geometries with finite-difference approximations has tended to eliminate backscatter by constraining $C \geq 0$. Just as negative values of $1/\text{Re}_{\text{eff}}$ can induce numerical instabilities, so can very high positive values of C by decreasing the effective Reynolds number of the flow, which could reduce the effective diffusive time scale below the time step. Hence a somewhat arbitrary upper bound on C of the order 0.04 or $C_s \leq 0.2$ is also used based on the theoretically derived value of $C_s = 0.18$ for homogeneous turbulence. More elaborate solutions for calculating C from equation (74) can be found in Ghosal et al. [44] and Meneveau et al. [45].

Table 6.1 summarizes calculations done in a fully developed ribbed duct from Tafti [46] at a nominal $\text{Re} = 20,000$. The duct is of square cross section with a square rib of dimension $e/D_h = 0.1$ and a rib pitch of $P/e = 10$, where e is the dimension of the rib and P is the streamwise pitch between ribs. Four calculations are shown at different resolutions to highlight the effect of the dynamic Smagorinsky model (DSM in Table 6.1). An SOC difference scheme is used for the convective terms. The dynamic Smagorinsky constant is localized and is constrained to be positive. The turbulent Prandtl number was fixed at 0.5 [47]. The heat transfer results were found to be quite insensitive to the turbulent Prandtl number and tests on a grid of 128^3 showed no difference when Pr_t was changed to 0.9. Two grid sizes of 96^3 and 128^3 are investigated with and without the model. Since Nusselt number augmentation ratio is a very weak function of Reynolds number in the range 20,000 to 30,000, the predictions are compared to the available experiments of Rau et al. [48] at $\text{Re} = 30,000$. The table summarizes the prediction of reattachment length of the separation bubble behind the rib, and Nusselt and friction coefficient augmentation.

All calculations reproduce the major flow structures with fidelity, namely the eddy formed at the junction between the rib and the wall, a recirculation zone formed on top of the rib and behind the rib with a corner eddy, and the lateral impingement of flow on the smooth side wall at the rib junction. Qualitatively, the bulk flow field results are indistinguishable, but quantitative differences of 10 to 15% exist between the different calculations. However, there are large differences in the predicted heat transfer and friction coefficients. The degree of underprediction of heat transfer and friction varies from 20 to 30% for the 96^3 quasi-DNS calculation to 15 to 20% for the 128^3 quasi-DNS. This is caused primarily by low turbulence intensities. The use of LES with the dynamic model increases the level

Table 6.1. Summary of heat transfer and friction data and percentage error with data of Rau et al. [48]. Experimental uncertainty is $\pm 5\%$

	Computations $e/D_h = 0.1, P/e = 10$				Rau et al. [48], $e/D_h = 0.1,$ $P/e = 10$
	96 ³	96 ³ DSM	128 ³	128 ³ DSM	
Re_τ	6,667	6,667	6,667	6,667	—
Re_b	24,000	20,000	22,000	20,000	30,000
% Form loss	92	94	92	91	85
Reattachment length (x_r/e)	4.8	4.3	4.6	4.1	4.0–4.25
$\langle Nu \rangle / Nu_0$ ($Nu_0 = 0.023 \cdot Re_b^{0.8} \cdot Pr^{0.4}$)					
Rib	2.22	2.84	2.54	2.89	—
Ribbed wall	1.78 (−26%)	2.35 (0%)	2.00 (−17%)	2.4 (0%)	2.40
Smooth sidewall	1.40 (−32%)	1.89 (−7%)	1.60 (−22%)	1.89 (−7%)	2.05
Overall with rib	1.67	2.22	1.89	2.23	—
Overall w/o rib	1.58 (−28%)	2.11 (−4.5%)	1.79 (−19%)	2.14 (−3.4%)	2.21
f/f_0 ($f_0 = 0.046 \cdot Re_b^{-0.2}$)					
Overall	6.11 (−36%)	8.53 (−10%)	7.23 (−24%)	8.6 (−9%)	9.5

of turbulence in the flow, particularly near walls, and is able to predict the heat transfer coefficient to within experimental uncertainty and friction coefficient to within 10% of experiments for both mesh resolutions. The level of turbulence augmentation provided by the dynamic model is commensurate with the mesh resolution such that the turbulent energy, heat transfer coefficient, and friction are predicted at the right levels independent of the resolution.

The dynamic Smagorinsky model has been used extensively in the literature for complex flows, and in spite of some as yet unresolved issues it does a reasonable job in calculating the eddy viscosity in complex flows. Chief among them is that it is difficult to separate out numerical errors from the subgrid model. Some new developments in subgrid modeling and LES simulations can be found in the defiltering approach to reconstruct the subgrid scales (Domaradzki and Saiki [49]; Domaradzki and Loh [50]; Stolz and Adams [51]), the use of grid hierarchies for dynamic simulations on coarse grids using information from fine grids to model

the subgrid stresses with large savings in computational effort (Dubois et al. [52, 53], Terracol et al. [54]), and the variational multiscale method of Hughes [55, 56]. Alternate methods of simulating turbulent flows without explicit subgrid modeling have also been proposed and used [31].

6.8 Detached Eddy Simulations or Hybrid RANS-LES

LES is an attractive alternative to DNS, but is still computationally expensive in wall-bounded flows, particularly when wall friction and heat transfer coefficients are sought from the calculation. Typically, to capture the energy-producing scales a fine grid resolution is required not only in the wall normal direction (which even wall integration or low Reynolds number RANS models require), but also in the wall parallel direction. In typical turbulent boundary layers the streaks and bursts, which are primarily responsible for near-wall production of turbulence, extend about 40 spanwise wall units and are spaced about 100 wall units apart. With LES it is imperative that these structures are resolved in the calculation, which puts large demands on the near-wall resolution as the Reynolds number increases. Hence it is not surprising that attempts have been made to use wall functions with LES (Cabot and Moin [57]; Piomelli and Balasas [58]).

The forbidding computational expense of LES at high Reynolds coupled with the inability of RANS models to accurately predict separation and reattachment led to the development of DES, or more generally a hybrid RANS-LES method, by Spalart et al. [1]. The motivation behind the formulation was to allow a RANS solution in the near-wall region which would default to LES away from the wall. Hence, in some sense it also acted as a wall model, wherein by using RANS very close to the wall, the strict resolution requirements of LES in the wall parallel direction could be relaxed considerably. The original formulation was introduced within the framework of the Spalart–Almaras [59] (SA) model. What differentiated the DES method from previous efforts to combine URANS and LES was the fact that a single model (in this case the SA model) acted both as the RANS model in the near-wall region and as an eddy-viscosity subgrid model in regions away from the wall. The switch in the intended use of the model occurs transparently and does not involve complicated matching of flow and turbulence quantities at the interface. In the case of the SA model, this is incorporated by comparing the distance to the wall (d) with the local cell size (Δ) (which in LES is representative of the smallest resolved scale). For $d < C_{des}\Delta$, the SA model is used in the RANS mode and when $d \geq C_{des}\Delta$, $C_{des}\Delta$ is used as the characteristic turbulence length scale for operation in LES mode. This idea was applied successfully to airfoils with massive unsteady separation (Shur et al. [60]), to external flow over cylinders (Travin et al. [61]), and to turbulent channel flow at high Reynolds numbers (Nikitin et al. [62]) and to a number of external flows with separation [63–66]. In initiating the switch from RANS to LES, while the formulation in the SA model related the turbulent length scale to the distance from the wall, Strelets [67] extended the DES concept to a two equation eddy-viscosity model (SST model) by calculating the turbulent length scale from the

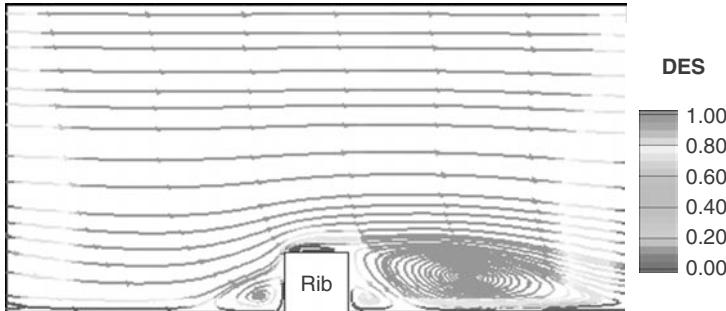


Figure 6.12. Plot of the time-averaged LES and RANS region in the DES computation for a 64^3 grid in a lateral plane. A value of 1 represents a complete LES region and a value of 0 a complete RANS region. Flow is periodic or fully developed in the streamwise direction. Reynolds number based on mean flow velocity and hydraulic diameter of duct is 20,000 [69].

model turbulent quantities. This is illustrated for the $k - \omega$ model (Wilcox [68]) in which the dissipation of turbulent energy, $-C_\mu k\omega$ is written as $k^{3/2}/l_{\text{DES}}$. The DES length scale is obtained from $l_{\text{DES}} = \min(l_{\text{RANS}}, l_{\text{LES}})$, where $l_{\text{RANS}} = \sqrt{k}/(C_\mu\omega)$ and $l_{\text{LES}} = C_{\text{DES}} \max(\Delta_x, \Delta_y, \Delta_z)$ is the turbulent length scale and the grid length scale, respectively. For a grid scale smaller than the turbulent length scale the model reverts to an LES solution. This feature facilitates the computation to be cognizant of the eddy length scales and hence behave as RANS or LES depending on the instantaneous local conditions, unlike in the case of SA model DES, where it is constrained to be a wall model. Another prominent feature of the DES version of the two equation model is that although an instantaneous discontinuity may exist between the RANS region and the LES region, in the mean however a smooth transition takes place from RANS to LES and vice versa. Figure 6.12 plots the fraction of time the model operates in LES mode at a given location in space. It can be observed from Figure 6.12 that the RANS regions transition into LES smoothly. In a DES SA model, the distance from the wall determines the switch from RANS to LES, which is not as receptive to instantaneous flow features.

Much like the SA model the near-wall region is always resolved in the RANS mode, which transitions to the LES mode as the distance from the wall increases. The flow in the vicinity of the ribs is mostly resolved in LES mode all the way to the channel center. This includes the unsteady large-scale dynamics of the separated shear layer as shown in Figure 6.12. Though the inter-rib spacing is predominantly LES, it transitions to a RANS mode at the center of inter-rib region. The grid facilitates the computation of the region in RANS and/or LES depending on the instantaneous conditions existing in the region. Downstream of the rib, in the region of separation it is observed that the computation is mostly carried out in the LES model. This sensitizes the RANS model to grid length scales, thereby

allowing the natural instabilities of the flow in this region to develop. Thus, the unsteady physics in the separated region is captured accurately.

In spite of its attractiveness and successful application, extension of DES to a wide range of flows poses several challenges [69]. DES is a strategy that depends on the grid not only for accurately resolving the turbulence in the regions of interest, but also in determining the switch between RANS and LES. Though guidelines have been prescribed for SA-based DES (Spalart [70]) in external flows, grid generation in internal flows is not trivial and requires some a priori knowledge of the flow. One probable alternative is the scale adaptive simulation (SAS) approach suggested by Menter et al. [71, 72] that compares a von Karman length scale to the turbulent length scales to make the RANS-LES switch independent of the grid.

DES still depends on the prediction capability of the base RANS model to predict phenomena such as separation, transition, and effects of secondary strain rates due to rotation and buoyancy forces. In previous investigations it has been found that while DES was successful in capturing secondary strain effects, which the base RANS model did not capture when used in RANS or URANS mode ([73, 74]), it also failed to predict shear layer transition in contrast to LES which was successful [75].

The transition from URANS to LES still remains a “grey area” in DES. The physics behind the switch from a completely modeled solution in the RANS regions to a well-resolved LES still remains to be understood [76].

6.9 Solution of Linear Systems

Solution of linear systems generated in the semi-implicit and implicit treatment of the momentum equations and the pressure equation requires careful consideration because these take up a majority of the computational time. The system matrix in generalized coordinates with second-order approximations is sparse with 18 off-diagonal bands on nonorthogonal grids and can either be symmetric or nonsymmetric. A symmetric system is only generated on orthogonal grids in the semi-implicit treatment of momentum and in the pressure equation, whereas the implicit treatment of convection always leads to a nonsymmetric system. Non-conformal block boundaries can further lead to strong asymmetries in the system matrix. There are a variety of methods for solving large sparse systems of linear equations (Meurant [77]). Out of these methods, iterative Krylov subspace methods have proven to be quite efficient for CFD applications (Barrett et al. [78]). While the method of conjugate gradients (CG) is useful for symmetric systems, methods like BiCGSTAB and GMRES(m) can be applied to nonsymmetric systems (Tafti [79]). Preconditioning can be applied using a variety of methods. One particular framework that is quite effective is based on a two-level additive/multiplicative Schwarz domain decomposition (DD) method (Smith et al. [80]; Dryja and Widlund [81]). In this method, the full system of equations is substructured into smaller sets of overlapping domains, which are then solved individually in an iterative manner, all the time updating the boundaries such that the global system is

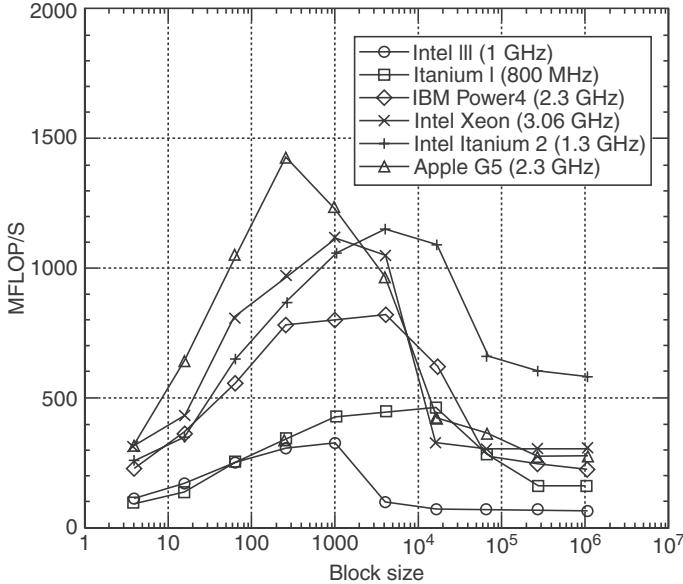


Figure 6.13. Effect of substructured block size on single-processor performance on some modern computer chips. MLOP, million floating point operations.

driven to convergence. Each subdomain is smoothed with an iterative method such as the Jacobi method or symmetric successive over-relaxation (SSOR) or incomplete LU (ILU) decomposition.

Not only does the DD method act as an effective preconditioner for the Krylov method, but also results in large performance gains on modern computer architectures. By substructuring the large system into smaller systems that are designed to fit into cache memory, main memory accesses are minimized, which result in large performance gains [82–84]. Figure 6.13 shows the effect of the sub-structured block size on performance with an additive Schwarz preconditioned conjugate gradient (ASPCG) algorithm for solving a linear system generated by a Laplace equation in Cartesian coordinates.

Figure 6.13 shows that there is an optimal block size that depends on the processor speed, cache size, speed and memory bandwidth, and latency. For very small blocks the cache lines are not utilized effectively and the overhead of getting data from memory exceeds the equivalent number of floating point operations performed on the data. As the block size increases, the overhead of memory accesses is amortized over more floating point operations on block data in cache, and hence performance increases till the block size gets so large that data required by the iterative solver for a given block does not fit in cache and has to be accessed from main memory multiple times – once again increasing the overhead of data reads and stores and decreasing performance. The asymptotic performance beyond a block

size of 10^5 is an indicator of the main memory performance for each architecture. Figure 6.13 shows one aspect of the equation that governs time to solution. The other effect that needs to be considered in designing an optimal block size is its effect on number of iterations to convergence. Typically as the block size decreases, convergence slows and the number of iterations increase. This is countered by using a two-level method in which a coarse level is implemented with a granularity of one coarse node per substructured block. A typical V-cycle between the two levels works much like in multigrid, in which the solution is relaxed individually in each block, then restricting the block-integrated residues on to the coarse level followed by an iterative smoothing of a correction, and finally a prolongation of the correction back to the blocks to be distributed across all nodes contained in the block. The additive Schwarz preconditioner provided the following benefits (Wang and Tafti [82]):

- Reduced the number of iterations for convergence in the Krylov solver by a factor of 2 to 3, hence reducing the number of global inner products and matrix-vector products.
- Increased floating point performance by a factor of 2 to 3.
- Decreased the overall CPU time by nearly an order of magnitude.

6.10 Parallelization Strategies

It would not be an overstatement to say that much of the gains in the widespread use of steady and time-dependent CFD has been facilitated by the exponential increase in computational power over the past two decades. Not only have processors become more powerful, but the ability to harness their collective power in parallel has also increased. While single-processor performance is dependent on the efficient use of fast cache, parallel performance is dependent on how efficiently a code parallelizes on a given architecture. Three distinct brands of parallel architectures have emerged – shared, distributed, and distributed-shared memory (DSM) architectures. In the shared memory architecture, multiple processors have access to the same shared memory through cache-coherent nonuniform memory accesses (cc-NUMA), whereas in the distributed memory architecture, the processors are distributed across a network with each processor only having access to its own memory. Hence, explicit messages are required to transfer data from one memory to the other. The DSM architecture combines the two architectures by having multiple shared memory nodes distributed across a network.

Barring recent research efforts in the computer science community, two distinct programming methods have evolved for parallel computing. One method takes advantage of the shared memory architecture by using compiler directives for parallelization, and the other uses explicit message passing libraries to move information between distributed memories. In the former the OpenMP API has evolved as the standard, whereas distributed memory programming has become synonymous with MPI (now Open MPI). The OpenMP API is a set of compiler

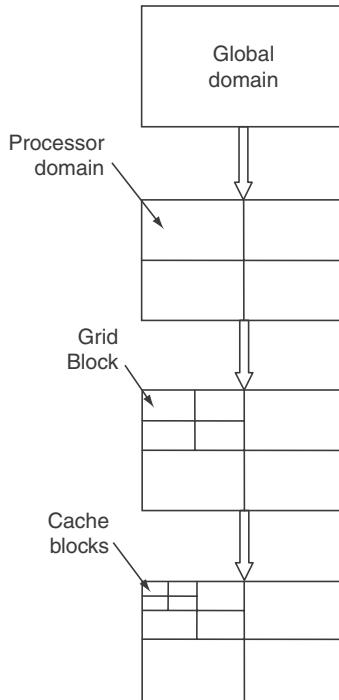


Figure 6.14. Data structure and mapping to processors and cache memory.

directives that instruct the code at runtime to perform certain operations in parallel across shared memory and can be implemented incrementally. Building parallelism into a serial code with MPI on the other hand requires that the mode of parallelism be decided a priori within a single-program multiple data (SPMD) framework such that for the most part the same operations are repeated on all processors. For CFD applications, that usually translates to a decomposition of the spatial grid into contiguous pieces mapped to different processors with explicit message passing calls through MPI for transferring data from one processor to another for global synchronicity, and requires a major rewrite of the code. Whereas OpenMP is simpler to implement and more flexible by being able to accommodate both different modes of parallelism and adaptation of parallelism [85, 86], it does not scale as well as codes written in MPI for these same reasons. There is merit in combining the two on DSM architectures to reap the benefits of both modes of parallelism (Tafti and Wang [87]).

The multiblock framework provides a natural framework for parallelization. Figure 6.14 illustrates the data structure and the multiple levels of parallelism that can be extracted. Depending on the total number of mesh blocks and the degree of parallelism sought, each processor can have multiple blocks residing on it as shown

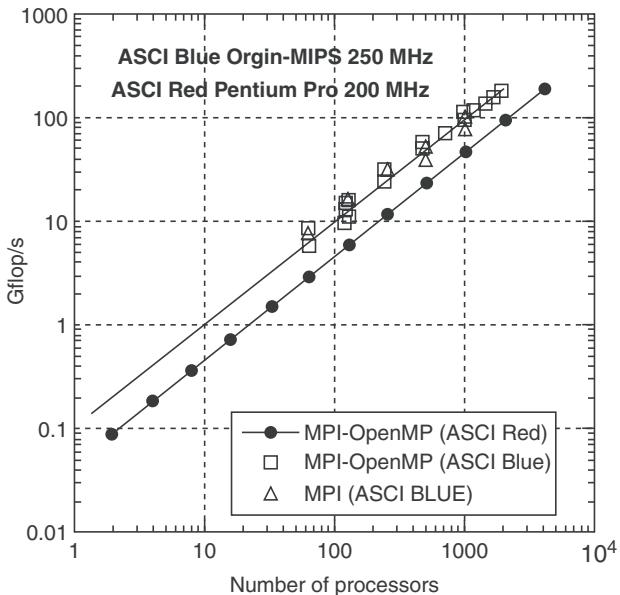


Figure 6.15. Parallel scalability of ASRPCG on two past ASCI platforms with MPI and MPI-OpenMP.

in Figure 6.14. The processor domain in Figure 6.14 is the same as an MPI process. Further within each block, “virtual cache blocks” are used for preconditioning the solution of linear systems. The “virtual” blocks are not explicitly reflected in the data structure, but are used only in the solution of linear systems. Hence embedded in each MPI process are additional levels of parallelism across mesh blocks or across cache blocks that can be exploited by OpenMP.

Figure 6.15 shows scalability studies on the ASCI Blue Origin 2000 cluster at Los Alamos and ASCI Red Intel machine at Sandia for a model problem. The domain size on each processor is maintained at $1,024 \times 1,024$ cells, with 1,024 sub-domains of size 32×32 for cache performance. The number of relaxation sweeps in the Jacobi smoother is kept at 40 in each block and 100 on the coarse level. The calculations on the ASCI Blue are run across a cluster of Origins using only MPI (in and across Origins) or a combination of MPI-OpenMP. In the hybrid MPI-OpenMP model, MPI processes are not only used across Origins but within Origins as well. For each $nproc$ (total number of processors used) in the figure, the multiple data points correspond to different combinations of MPI and OpenMP threads executed across different number of Origins. On the ASCI Red, OpenMP is used within each two-processor node. In all cases, the scaled performance exhibits a perfect linear speedup showing the efficacy of the architecture as well as the use of hybrid parallelism.

6.11 Applications

The theory and modeling techniques described in this chapter have been used extensively for LES ([88–92]) and DES ([73–75, 93–95]) of turbine internal cooling ducts with ribs with rotation and centrifugal buoyancy effects at $\text{Re} = 20,000$ to $50,000$ and LES of leading edge film cooling at $\text{Re} = 100,000$ ([96–98]). A number of investigations have also been performed in studying different fin configurations or heat transfer enhancement techniques in compact heat exchangers, which mostly operate in the transitional regime between $\text{Re} = 100$ to $2,000$ ([99–103]). A dynamic mesh capability is presently being used to investigate flapping flight for applications to Micro-Air Vehicles (MAVs) at Reynolds numbers between $10,000$ and $100,000$ ([104, 105]). A two-phase dispersed phase capability using Lagrangian particle dynamics has been used with LES to model fouling of internal cooling channels and film cooling holes ([106, 107]).

References

- [1] Spalart, P. R., Jou, W. H., Strelets. M., and Allmaras. S. R. Comments on the Feasibility of LES for Wings and a Hybrid RANS/LES Approach, First AFSOR Int. Conf. on DNS/LES, Aug. 4–8, 1997, In *Advances in DNS/LES*, C. Liu and Z. Liu Eds, Greden Press, Columbus, OH.
- [2] White, F. M. *Viscous Fluid Flow*, McGraw-Hill Series in Mechanical Engineering, second edition, McGraw-Hill, 1991.
- [3] Patankar, S. V., Liu, C. H., and Sparrow, E. M. Fully Developed Flow and Heat Transfer in Ducts Having Streamwise-Periodic Variations of Cross-Sectional Area, *ASME J. Heat Transfer*, 99, pp. 180–186, 1977.
- [4] Thompson, J. F., Warsi, Z. U. A., and Mastin, C. W. *Numerical Grid Generation Foundations and Applications*, North-Holland, Elsevier Science Publishing Co., Inc., New York, 1985.
- [5] Burden, R. L., Faires, J. D., and Reynolds, A. C. *Numerical Analysis*, Prindle, Weber and Schmidt, Boston, MA, 1980.
- [6] Kim, J., Moin, P., and Moser, R. Turbulence Statistics in Fully Developed Channel Flow at Low Reynolds Number, *J. Fluid Mech.*, 177, pp. 133–166, 1987.
- [7] Tafti, D. K., Zhang, L. W., and Wang, G. Time-Dependent Calculation Procedure for Fully-Developed and Developing Flow and Heat Transfer in Louvered Fin Geometries, *Numerical Heat Transfer A*, 35, pp. 225–249, 1999.
- [8] Patankar, S. *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing, 1980.
- [9] Versteeg, H. K., and Malalasekera, W. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, Prentice Hall, 1995.
- [10] Ferziger, J. H., and Peric, M. *Computational Methods for Fluid Dynamics*, Springer, 2002.
- [11] Chorin, A. J. Numerical solution of the Navier-Stokes equations, *Maths. Comput.*, 22, pp. 745–762, 1968.
- [12] Kim, J., and Moin, P. Application of a fractional-step method to incompressible Navier-Stokes. *J. Comput. Phys.*, 59, pp. 308–323, 1985.
- [13] Choi, H., and Moin, P. Effects of the Computational Time Step on Numerical Solutions of Turbulent Flow, *J. Comput. Phys.*, 113, pp. 1–4, 1994.

- [14] Tafti, D. K. Alternate Formulations for the Pressure Equation Laplacian on a Collocated Grid for Solving the Unsteady Incompressible Navier-Stokes Equations, *J. Comput. Phys.*, 116, pp. 143–153, 1995.
- [15] Zang, Y, Street, R. L., and Koseff, J. R. A Non-Staggered Grid, Fractional Step Method for Time-Dependent Incompressible Navier-Stokes Equations in Curvilinear Coordinates, *J. Comput. Phys.*, 114(1), pp. 18–33, 1994.
- [16] Sotiropoulos, F., and Abdallah, S. The Discrete Continuity Equation in Primitive-Variable Solutions of Incompressible Flow, *J. Comput. Phys.*, 95, pp. 212–227, 1991.
- [17] Lele, S. K. Compact finite difference schemes with spectral-like resolution. *J. Comput. Phys.*, 103(1), pp. 16–42 (1992).
- [18] Beaudan, P., and Moin, P. Numerical experiments on the flow past a circular cylinder at sub-critical Reynolds number. Rep. No. TF-62, Thermosciences Division, Department of Mechanical Engineering, Stanford University, December 1994.
- [19] Kravchenko, A. G., and Moin, P. On the Effect of Numerical Errors in Large Eddy Simulations of Turbulent Flows, *J. Comp. Phys.*, 131, pp. 310–322, 1997.
- [20] Phillips, N. A. An example of non-linear computational instability. In *The Atmosphere and Sea in Motion*, Rockfeller Institute Press in association with Oxford University Press, New York, pp. 501–504, 1959.
- [21] Lilly, D. K. On the computational stability of numerical solutions of time-dependent non-linear geophysical fluid dynamics problems, *Monthly Weather Rev.*, 93(1), pp. 11–25, 1965.
- [22] Arakawa, A. Computational Design for Long-Term Numerical Integration of the Equations of Fluid Motion: Two-Dimensional Incompressible Flow. Part 1. *J. Comput. Phys.*, 1, pp. 119–143, 1966.
- [23] Morinishi, Y., Lund, T. S., Vasilyev O. V., and Moin, P. Fully Conservative Higher Order Finite Difference Schemes for Incompressible Flow, *J. Comp. Phys.*, 143(1), pp. 90–124, 1998.
- [24] Rai, M. M., and Moin, P. Direct simulations of turbulent flow using finite-difference schemes. *J. Comput. Phys.*, 96, pp. 15–53, 1991.
- [25] Tafti, D. K. Comparison of Some Upwind-Biased High-Order Formulations With a Second-Order Central Difference Scheme For Time integration of the Incompressible Navier-Stokes Equations, *Comput. & Fluids*, 25(7), pp. 547–655, 1996.
- [26] Kreplin, H. P., and Eckelmann, H. Behavior of the Three Fluctuating Velocity Components in the Wall Region of a Turbulent Channel Flow, *Phys. Fluids*, 22(7), pp. 1233–1239, 1979.
- [27] Leonard, B. P. The Ultimate Conservative Difference Scheme Applied to Unsteady One-Dimensional Advection, *Comp. Methods App. Mech. Engg.*, 88, pp. 17–74, 1991.
- [28] Thuburn, J. Multidimensional Flux-Limited Advection Schemes, *J. Comp. Phys.*, 123, pp. 74–83, 1996.
- [29] Germano, M., Piomelli, U., Moin, P., and Cabot, W. H. A Dynamic Subgrid-Scale Eddy Viscosity Model, *Phys. Fluids A*, 3, pp. 1760–1765, 1991.
- [30] Margolin, L. G., Rider, W. J., and Grinstein, F. F. Modeling turbulent flow with implicit LES, *J. Turbulence*, 7(15), pp. 1–27, 2006.
- [31] Grinstein, F. F., Len, G., Margolin, L. G., and Rider, W. J. *Implicit Large Eddy Simulation: Computing Turbulent Fluid Dynamics*, Cambridge University Press, 2007.
- [32] Sagaut, P., and Meneveau, C. *Large Eddy Simulation for Incompressible Flows An Introduction*, Scientific Computation, Springer-Verlag, Berlin, 2006.

- [33] Leonard, A. Energy Cascade in Large-Eddy Simulations of Turbulent Fluid Flows, *A. Geophys.*, 18(1), pp. 237–248, 1975.
- [34] Clark, R. A., Ferziger, J. H., and Reynolds, W. C. Evaluation of Subgrid-scale Models Using an Accurately Simulated Turbulent Flow, *J. Fluid Mech.*, 9(1), pp. 1–16, 1979.
- [35] Rogallo, R. S., and Moin, P. Numerical Simulation of Turbulent Flows, *Ann. Rev. Fluid Mech.*, 16, pp. 99–137, 1984.
- [36] Ferziger, J. H. Large Eddy Simulation, Simulation and Modelling of Turbulent Flows – Part III. In T. B. Gatski, M. Y. Hussaini & J. L. Lumley, eds., *ICASE/LaRC Series in Computational Science and Engineering*, Oxford University Press, 1996.
- [37] Lesieur, M., and Metais, O. New Trends in Large-Eddy Simulations of Turbulence, *Ann. Rev. Fluid Mech.*, 28, pp. 45–82, 1996.
- [38] Smagorinsky, J. General Circulation Experiments with the Primitive Equations. I. The Basic Experiment, *Monthly Weather Rev.*, 91, pp. 99–164, 1963.
- [39] Najjar, F. M., and Tafti, D. K. Evaluation of the Dynamic Subgrid-Scale Stress Model in Finite-Differenced LES: Effects of Grid Resolution and Inhomogeneous Test Filtering, Symposium on LES and DNS of Turbulent Flow, *Proceedings of the ASME-IMEC&E Conference*, Atlanta, GA, Nov. 1996.
- [40] Najjar, F. M., and Tafti, D. K. Finite-Differenced Large-Eddy Simulations of Incompressible Turbulent Flows Using the Dynamic Subgrid-Scale Stress Model: *Effects of Test Filters and Numerical Approximations, Tech. Rep. 031, National Center for Supercomputing Applications*, University of Illinois at Urbana-Champaign, 1995.
- [41] Najjar, F. M., and Tafti, D. K. Study of Discrete Test Filters and Finite-Difference Approximations for the Dynamic Subgrid-Scale Stress Model, *Phys. Fluids*, 8(4), pp. 1076–1088, 1996.
- [42] Jordon, S. A. Large-Eddy Simulation Methodology in Generalized Curvilinear Coordinates, *J. Comput. Phys.*, 148, pp. 322–340, 1999.
- [43] Lilly, D. K. A Proposed Modification of the Germano Subgrid-Scale Closure Method, *Phys. Fluids A*, 4, pp. 633–635, 1992.
- [44] Ghosal, S., Lund, T. S., Moin, P., and Akselvoll, K. A Dynamic Localization Model for Large-Eddy Simulation of Turbulent Flows, *J. Fluid Mech.*, 286, pp. 229–255, 1995.
- [45] Meneveau, C., Lund, T. S., and Cabot, W. H. A Lagrangian Dynamic Subgrid-Scale Model of Turbulence, *J. Fluid Mech.*, 319, pp. 353–385, 1996.
- [46] Tafti, D. K. Evaluating the Role of Subgrid Stress Modeling in a Ribbed Duct for the Internal Cooling of Turbine Blades, *Int. J. Heat Fluid Flow*, 26(1), pp. 92–104, 2005.
- [47] Moin, P., Squires, K., Cabot, W., and Lee, S. A Dynamic Sub-Grid-Scale Model for Compressible Turbulence and Scalar Transport, *Phys. Fluids A*, 3–11, pp. 2746–2757, 1991.
- [48] Rau, G., Cakan, M., Moeller, D., and Arts, T. The Effect of Periodic Ribs on the Local Aerodynamic and Heat Transfer Performance of a Straight Cooling Channel, *ASME J. Turbomach.*, 120, pp. 368–375, 1998.
- [49] Domaradzki, J. A., and Saiki, E. M. A Subgrid-Scale Model Based on the Estimation of Unresolved Scales of Turbulence, *Phys. Fluids*, 9(7), pp. 2148–2164, 1997.
- [50] Domaradzki, J. A., and Loh K.-H. The Subgrid-Scale Estimation Model in the Physical Space Representation, *Phys. Fluids*, 11(8), pp. 2330–2342, 1999.
- [51] Stolz, S., and Adams, N. A. An Approximate Deconvolution Procedure for Large-Eddy Simulation, *Phys. Fluids*, 11(7), pp. 1699–1701, 1999.
- [52] Dubois, T., Jauberteau, F., and Temam, R. Dynamic multilevel methods and the Numerical Simulation of Turbulence, Cambridge University Press, Cambridge, 1999.

- [53] Dubois, T., Jauberteau, F., and Temam, R. Incremental Unknowns, Multilevel Methods and the Numerical Simulation of Turbulence, *Comput. Methods Appl. Mech. Engg.*, 159, pp. 123–189, 1998.
- [54] Terracol, M., Sagaut, P., and Basdevant, C. A Time Self-adaptive Multilevel Algorithm for Large-Eddy Simulation, *J. Comput. Phys.*, 184, pp. 339–365, 2003.
- [55] Hughes, J. R. T., Mazzei, L., and Oberai, A. A. The Multiscale Formulation of Large Eddy Simulation: Decay of Homogeneous Isotropic Turbulence, *Phys. Fluids*, 13(2), pp. 505–511, 2001.
- [56] Hughes, J. R. T., Oberai, A. A., and Mazzei, L. Large Eddy Simulation of Turbulent Channel Flows by the Variational Multiscale Method, *Phys. Fluids*, 13(6), pp. 1784–1798, 2001.
- [57] Cabot, W., and Moin, P. Approximate Wall Boundary Conditions in the Large-Eddy Simulation of High Reynolds Number Flow, *Flow, Turbulence Combustion* 63, pp. 269–291, 1999.
- [58] Piomelli, U., and Balaras, E. Wall-Layer Models for Large-Eddy Simulations, *Annu. Rev. Fluid Mech.*, 34, pp. 349–374, 2002.
- [59] Spalart, P., and Allmaras, S. R. A One-Equation Turbulence Model for Aerodynamic Flows, *La Recherche Aerospatiale*, 1, pp. 5–21, 1994.
- [60] Shur, M., Spalart, P. R., Strelets, M., and Travin, A. Detached-Eddy Simulation of an Airfoil at High Angle of Attack, *4th Int. Symposium on Engineering Turbulence Modeling and Measurements*, Corsica, France, 1999.
- [61] Travin, A., Shur, M., Strelets, M., and Spalart, P. Detached-Eddy Simulations Past a Circular Cylinder, *Flow, Turbulence Combustion*, 63, pp. 293–313, 1999.
- [62] Nikitin, N. V., Nicoud, F., Wasistho, B., Squires, K. D., and Spalart, P. R. An Approach to Wall Modeling in Large-Eddy Simulations, *Phys. Fluids*, 12(7), pp. 1629–1632, 2000.
- [63] Morton, S., Forsythe, J., Mitchell, A., and Hajek, D. Detached-Eddy Simulation and Reynolds-Averaged Navier-Stokes Simulations of Delta Wing Vortical Flowfields, *ASME J. Fluids Engg.*, 124, pp. 924–932, 2002.
- [64] Forsythe, J. R., Hoffmann, K. A., Cummings, R. M., and Squires, K. D. Detached-Eddy Simulation With Compressibility Corrections Applied to a Supersonic Axisymmetric Base Flow, *ASME J. Fluids Engg.*, 124, pp. 911–923, 2002.
- [65] Squires, K. D., Forsythe, J. R., Morton, S. A., Strang, W. Z., Wurtzler, K. E., Tomaro, R. F., Grismer, M. J., and Spalart, P. R. Progress on Detached-Eddy Simulation of Massively Separated Flow, *AIAA Paper 2002-1021*, 2002.
- [66] Forsythe, J. R., Squires, K. D., Wurtzler, K. E., and Spalart, P. R. Detached-Eddy Simulation of fighter aircraft at high alpha, *AIAA Paper 2002-0591*, 2002.
- [67] Strelets, M. Detached Eddy Simulation of Massively Separated Flows, *AIAA 2001-0879, 39th AIAA Aerospace Sciences Meeting and Exhibit*, 2001.
- [68] Wilcox, D. C. Reassessment of the Scale-Determining Equation for Advanced Turbulence Models, *AIAA J.*, 26(11), pp. 1299–1310, Nov. 1988.
- [69] Viswanathan, A. K. Detached Eddy Simulation of Turbulent Flow and Heat Transfer in Turbine Blade Internal Cooling Ducts, Ph.D. thesis, Mechanical Engineering Department, Virginia Polytechnic Institute and State University, April 2006.
- [70] Spalart, P. R. Young-Person's Guide to Detached Eddy Simulations, NASA/CR-2001-211032, 2001.
- [71] Menter, F., Kuntz, M., and Bender, R., A Scalable Adaptive Simulation Model for Turbulent Flow Predictions, *AIAA paper 2003-0767*, 2003.
- [72] Menter, F., and Egorov, V. A Scale Adaptive Simulation Model using Two-Equation Models, *AIAA paper 2005-1095*, 2005.

- [73] Viswanathan, A. K., and Tafti, D. K. Investigation of Detached Eddy Simulation in Capturing the Effects of Coriolis Forces and Centrifugal Buoyancy in Ribbed Ducts, *ASME J. Heat Transfer*, 129(7), pp. 778–789, July 2007.
- [74] Viswanathan, A. K., and Tafti, D. K. Capturing the Effects of Rotation in Sudden Expansion Ducts using Detached Eddy Simulation, *AIAA J.*, 45(8), pp. 2100–2102, August 2007.
- [75] Viswanathan, A. K., and Tafti, D. K. Detached Eddy Simulation of Turbulent Flow and Heat Transfer in a Two-Pass Internal Cooling Duct, *Int. J. Heat Fluid Flow*, 27(1), pp. 1–20, 2006.
- [76] Davidson, L., and Billson, M. Hybrid LES-RANS Using Synthesized Turbulent Fluctuations for Forcing in the Interface Region. *Int. J. Heat Fluid Flow*, 27, pp. 1028–1042, 2006.
- [77] Meurant, G. *Computer Solution of Large Linear Systems* (Studies in Mathematics and its Applications 28), Elsevier Oxford UK, 1999.
- [78] Barrett, R., Berry, M., Chan, T. F., Demmel, J., Donato, J., Dongarra, J., Eijkhout, V., Pozo, R., Romine, C. and van der Vorst, H., *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods* (Miscellaneous Titles in Applied Mathematics Series No 43), Philadelphia, PA, SIAM, 1994.
- [79] Tafti, D. K. A Study of Krylov Methods for the Solution of the Pressure-Poisson Equation on the CM-5, FED-Vol. 215, pp. 1–8, *ASME /JSME Fluids Engineering and Laser Anemometry Conference and Exhibition*, Hilton Head, CA, August 13–18, 1995.
- [80] Smith, B., Bjorstad, P., and Gropp, W. *Domain Decomposition, Parallel Multilevel Methods for Elliptic Partial Differential Equations*, Cambridge University Press, New York, 1996.
- [81] Dryja, M., and Widlund, O. B. An Additive Variant of the Schwarz Alternating Method for the Case of Many Subregions, Tech. Rep. #339, Courant Institute, New York University, 1987.
- [82] Wang, G., and Tafti, D. K. Performance Enhancement on Microprocessors with Hierarchical Memory Systems for Solving Large Sparse Linear Systems, *Int. J. Supercomp. Appl. High Perform. Compu.*, 13(1), pp. 63–79, Spring 1999.
- [83] Wang, G., and Tafti, D. K. Uniprocessor Performance Enhancement by Additive Schwarz Preconditioners on Origin 2000, *Adv. Eng. Softw.*, 29(3–6), pp. 425–431, 1998.
- [84] Wang, G., and Tafti, D. K. Parallel Performance of Additive Schwarz Preconditioners on Origin 2000, *Adv. Eng. Softw.*, 29(3–6), pp. 433–439, 1998.
- [85] Huang, W., and Tafti D. K. A Parallel Computing Framework for Dynamic Power Balancing in Adaptive Mesh Refinement Applications, In *Parallel Computational Fluid Dynamics, Towards Teraflops, Optimization and Novel Formulations*, D. Keyes, A. Ecer, J. Periaux, N. Satofuka and P. Fox eds., pp. 249–256, Elsevier Science B. V., 2000.
- [86] Huang, W., and Tafti, D. K. A Parallel AMR Algorithm for Solving Nonlinear Dynamical Systems, *Int. J. High Perform. Compl. Appl.*, 18(12), May 2004.
- [87] Tafti, D. K., and Wang, G. Application of Embedded Parallelism to Large Scale Computations of Complex Industrial Flows, *Proceedings of the ASME Fluids Engineering Division*, FED 247, pp. 123–130, 1998.
- [88] Abdel-Wahab, S., and Tafti, D. K. Large Eddy Simulations of Flow and Heat Transfer in a 90° Ribbed Duct with Rotation – Effect of Coriolis and Centrifugal Buoyancy Forces, *ASME J. Turbomach.*, 126(4), pp. 627–636, 2004.

- [89] Sewall, E. A., Tafti, D. K., Thole, K.A., and Graham, A. Experimental Validation of Large Eddy Simulations of Flow and Heat Transfer in a Stationary Ribbed Duct, *Int. J. Heat Fluid Flow*, 27(2), pp. 243–258, 2006.
- [90] Sewall, E. A., and Tafti, D. K. Large Eddy Simulation of Flow and Heat Transfer in the 180° Bend Region of a Stationary Ribed Gas Turbine Blade Internal Cooling Duct, *ASME J. Turbomach.*, 128, pp. 763–771, 2006.
- [91] Sewall, E. A., and Tafti, D. K. Large Eddy Simulation of Flow and Heat Transfer in the Developing Flow Region of a Rotating Gas Turbine Blade Internal Cooling Duct with Coriolis and Buoyancy Forces, *ASME J. Turbomach.*, 113(1), Jan. 2008.
- [92] Elyyan, M. A., and Tafti, D. K. LES Investigation of Flow and Heat Transfer in a Channel with Dimples and Protrusions, *ASME J. Turbomach.*, 130(4), 2008.
- [93] Viswanathan, A. K., and Tafti, D. K. Detached Eddy Simulation of Turbulent Flow and Heat Transfer in a Ribbed Duct, *ASME J. Fluids Eng.*, 127(5), pp. 888–896, 2005.
- [94] Viswanathan, A. K., and Tafti, D. K. A Comparative Study of DES and URANS for Flow Prediction in a Two-Pass Internal Cooling Duct, *ASME J. Fluids Eng.*, 128(6), pp. 1336–1345, Nov. 2006.
- [95] Viswanathan, A. K., and Tafti, D. K. Detached Eddy Simulation of Flow and Heat Transfer in Fully-Developed Rotating Internal Cooling Channel with Normal Ribs, *Int. J. Heat Fluid Flow*, 27(3), pp. 351–370, 2006.
- [96] Rozati, A., and Tafti, D. K. Large Eddy Simulation of Leading Edge Film Cooling Part-II: Heat Transfer and Effect of Blowing Ratio, *ASME J. Turbomach.*, 130(4), 2008.
- [97] Rozati, A., and Tafti, D. K. Large-Eddy Simulation of Leading Edge Film Cooling, *Int. J. Heat Fluid Flow*, 29, pp. 1–17, 2008.
- [98] Rozati, A., and Tafti, D. K. Effect of Coolant-Mainstream Blowing Ratio on Leading Edge Film Cooling Flow and Heat Transfer – LES Investigation, *Int. J. Heat Fluid Flow*, 29, pp. 857–873, 2008.
- [99] Tafti D. K., and Cui, J. Fin-Tube Junction Effects on Flow and Heat Transfer in Flat Tube Corrugated Multilouvered Heat Exchangers, *Int. J. Heat Mass Transfer*, 46(11), pp. 2027–2038, 2003.
- [100] Elyyan, M. A., Rozati, A., and Tafti, D. K. Investigation of Dimpled Fins for Heat Transfer Enhancement in Compact Heat Exchangers, *Int. J. Heat Mass Transfer*, 51, pp. 2950–2966, 2008.
- [101] Rozati, A., Tafti, D. K., and Blackwell, N. E. Effect of Pin Tip Clearance on Flow and Heat Transfer at Low Reynolds Numbers, *ASME J. Heat Transfer*, 130, July 2008.
- [102] Srinivas, S., Tafti, D. K., Rozati, A., and Blackwell, N. E. Heat-Mass Transfer and Friction Characteristics of Profiled Pins at Low Reynolds Numbers, *Numerical Heat Transfer A*, 54(2), pp. 130–150, 2008.
- [103] Elyyan, M., and Tafti, D. K. A Novel Split Dimple Interrupted Fin Configuration for Heat Transfer Augmentation, *Int. J. Heat Mass Transfer*, 52, pp. 1561–1572, 2009.
- [104] Gopalakrishnan, P., and Tafti, D. K. A Parallel Boundary Fitted Dynamic Mesh Solver for Applications to Flapping Flight, *Comp. Fluids*, 38(8), pp. 1592–1607, 2009.
- [105] Gopalakrishnan, P., and Tafti, D. K. Effect of Rotation and Angle of Attack on Force Production of Flapping Flights, *AIAA J.*, 47(11), pp. 2505–2518, 2009.
- [106] Shah, A., and Tafti, D. K. Transport of Particulates in an Internal Cooling Ribbed Duct, *ASME J. Turbomach.*, 129(4), pp. 816–825, October 2007.
- [107] Rozati, A., Tafti, D. K., and Shreedharan, S. Effects of Syngas Ash Particle size on Deposition and Erosion of a Film-Cooled Leading Edge, Paper No. 56155, 2008 *ASME Summer Heat Transfer Conference*, Aug. 10–14, Jacksonville, FL.

7 On large eddy simulation of turbulent flow and heat transfer in ribbed ducts

Bengt Sundén and Rongguang Jia
Lund University, Lund, Sweden

Abstract

This chapter presents some work on large eddy simulation (LES) for heat transfer studies. Firstly, the numerical accuracy is validated with the simulation of the fully developed pipe flow with the direct numerical simulation (DNS) data. Secondly, the solver is used to study the heat transfer and fluid flow in rib-roughened ducts, where the ribs could be straight or V-shaped. This is primarily to clarify a contradiction on which pointing direction is the better for V-shaped ribs. Lastly, current status and prospects of LES are presented.

Keywords: LES, Heat transfer, Ribs, Duct flow

7.1 Introduction

In order to allow the gas turbine designer to increase the turbine inlet temperature while maintaining an acceptable material temperature, sophisticated cooling methods are essential. For the internal cooling of the blades, ribbed ducts are usually used. The presence of ribs, also called roughness or turbulators, enhances the heat transfer coefficients by redevelopment of the boundary layer after flow reattachment between the ribs, and because of induced secondary flows. The rib layouts and configurations can be varied and include 90-degree, 60-degree, and 45-degree parallel ribs and 45-degree and 60-degree V-shaped ribs.

It is essential to be able to accurately predict the enhancement of heat transfer generated by the roughness elements to ensure good design decisions. Accordingly, the heat transfer and fluid flow in ribbed ducts have been extensively studied both experimentally and numerically. A summary is presented in Jia et al. [1], where contradictions on experimental studies of V-shaped ribs have been identified. The experimental studies of both Han et al. [2] and Olsson and Sundén [3] show that square ducts roughened with upstream pointing V-shaped ribs perform better than those pointing downstream. However, results on the contrary were obtained by Taslim et al. [4] and Gao and Sundén [5]. The contradiction might be due to the

differences in the experimental conditions, which need further clarification. In a previous study [1] by the RANS method, side-wall heating was found to be the main reason for the discrepancy. However, the RANS method filtered out too many details, so that the explanation of the reason becomes difficult. In addition, the uncertainties of the RANS models may become large for complex geometries. Therefore, a more universal modeling method, large eddy simulation (LES), is employed to further confirm and explain the findings from RANS.

LES is a computational method in which the large eddies are computed and the smallest, subgrid-scale (SGS) eddies are modeled. The underlying premise is that the largest eddies are directly affected by the boundary conditions, carrying most of the anisotropic Reynolds stresses, and must be computed. The small-scale turbulence is weaker, contributing less to the turbulent Reynolds stresses, is nearly isotropic, and has nearly-universal characteristics. Therefore, it is more amenable to model.

In LES, the transport equations are filtered from the 3D, unsteady, Navier–Stokes equations by a filter function in space:

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial(\bar{\rho} \bar{u}_i)}{\partial x_i} = 0 \quad (1)$$

$$\frac{\partial(\bar{\rho} \bar{u}_i)}{\partial t} + \frac{\partial(\bar{\rho} \bar{u}_j \bar{u}_i)}{\partial x_j} = - \left(\frac{\partial p^*}{\partial x_i} - \beta \right) + \frac{\partial}{\partial x_j} \left[\bar{\mu} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) + \bar{\rho} \tau_{ij} \right] \quad (2)$$

$$\frac{\partial(\bar{\rho} \bar{\phi}^*)}{\partial t} + \frac{\partial(\bar{\rho} \bar{u}_j \bar{\phi}^*)}{\partial x_j} = \frac{\partial}{\partial x_j} \left(\frac{\bar{\mu}}{\bar{P}_r} \frac{\partial \bar{\phi}^*}{\partial x_j} + \bar{\rho} \varphi_j \right) - \delta_{1j} \bar{\rho} \bar{u}_j \gamma \quad (3)$$

where $\tau_{ij} = \bar{u}_i \bar{u}_j - \bar{\bar{u}_i \bar{u}_j}$ are called the subgrid-scale (SGS) stresses, through which the small scales influence the large (resolved) scales. Accordingly, $\varphi_j = \bar{\phi} \bar{u}_j - \bar{\bar{\phi} u_j}$ are called the SGS heat fluxes. They are unknown and must be modeled by a SGS model.

The implicit model, proposed by Boris et al. [6], is the simplest SGS model without introducing neither the filtering operation nor the SGS stress tensor. It is based on the observation that truncation errors in such discretizations of Navier–Stokes equations introduce numerical dissipation with the implicit effects of the discretization qualitatively similar to the effects of the explicit SGS models.

However, in the presence of the inviscid unsteadiness, the turbulence becomes much stronger, such as the separation zone downstream a rib. Consequently, the numerical diffusion is not enough to dissipate the turbulent energy, especially for a fine grid. Some additional viscosity is necessary to prevent the buildup of turbulent energy at high frequency regions. Smagorinsky [7] was the first to postulate an explicit model for the SGS stresses. His model assumes the SGS stresses follow a gradient-diffusion process, similar to molecular motion. Consequently, τ_{ij} is given by:

$$\tau_{ij} = 2\nu_T S_{ij} \quad (4)$$

where $S_{ij} = \frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right)$ is called the “resolved strain rate,” and ν_T is the Smagorinsky eddy viscosity given by:

$$\nu_T = (C_S \Delta)^2 \sqrt{S_{ij} S_{ij}} \quad (5)$$

where Δ and C_S are the filter width (grid spacing) and the Smagorinsky constant, respectively, which must be specified prior to a simulation, varying from flow to flow. The Smagorinsky constant is set as $C_S = 0.1$ in this simulation.

7.2 Numerical Method

The computations are carried out in an in-house parallel multi-block computer code CALC-MP [8, 9], based on the finite volume technique. The code uses a collocated body-fitted grid system and employs the improved Rhie and Chow interpolation procedure method to calculate the velocities at the control volume faces. The SIMPLEC algorithm couples the pressure and velocity. An algorithm based on SIP (Strongly Implicit Procedure) is used for solving the algebraic equations. Both the diffusion and convection terms are discretized by the second order central difference. The validation of the code has been presented by Jia [10].

7.3 Results And Discussion

7.3.1 Fully developed pipe flow

To illustrate the numerical accuracy of the present code, a fully developed pipe flow is calculated with LES, for $Re_\tau = 460$. Implicit model is used for modeling the SGS stresses, and a central scheme is used for the convection terms. A butterfly-type multi-block grid is generated, and the approximate number of grid points is around one million.

Although flows such as a turbulent pipe flow are geometrically simple, their accurate simulation using techniques such as LES is in some way more difficult than flows in complicated geometries. This is because no inviscid instability is available, its accurate simulation is an indication of the reliability of the method.

Figure 7.1a shows the spectrum at $y+ = 230$ (i.e., a point at the centerline of the pipe), and the cutoff of the high-frequency turbulence energy is clearly illustrated.

Figure 7.1b shows the mean main stream velocity and turbulent Reynolds stresses by LES, in comparison with the DNS data of Eggels et al. [11]. They are in very good agreement. At the region close to the wall, the fluctuations in the main flow direction are much larger than those in the other two directions. From any RSTM model (Reynolds stress turbulence model), one can see that the production term for the fluctuations of the two cross directions is zero. The fluctuations of these two directions obtain the turbulence energy from pressure strain terms. The pressure strain terms redistribute energy from the high-fluctuation directions

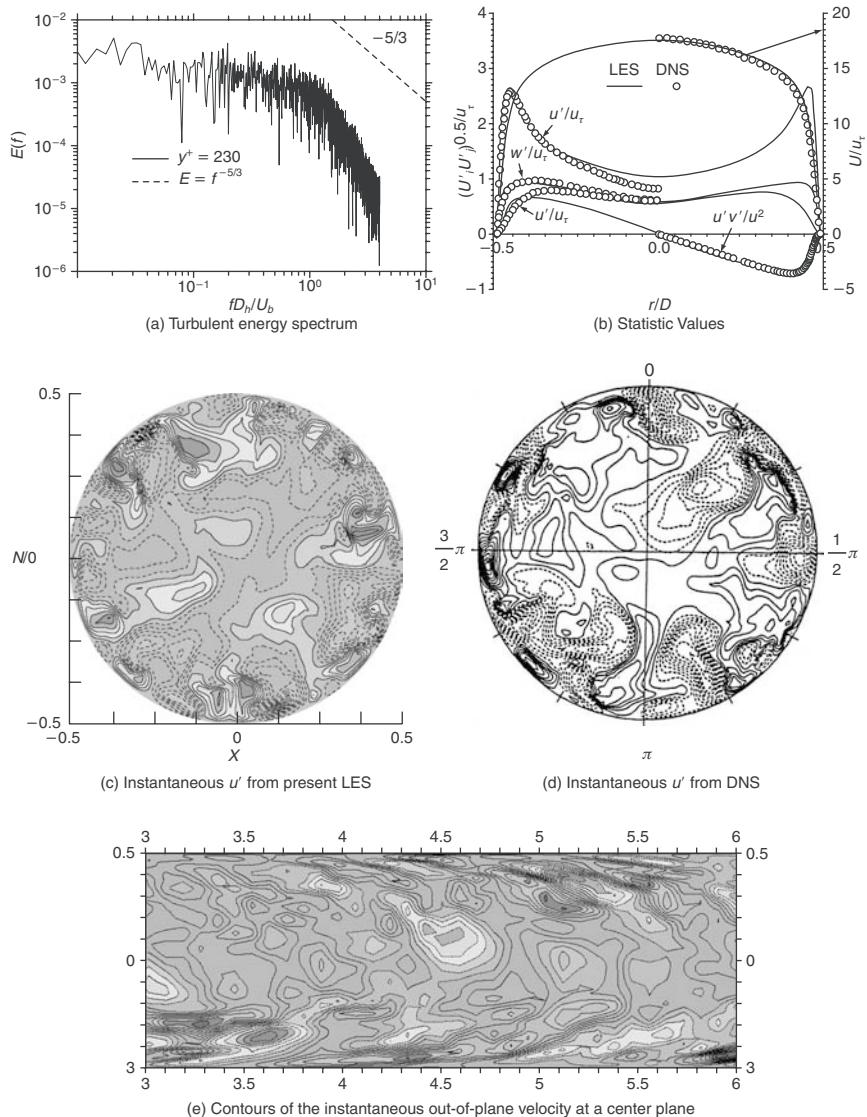


Figure 7.1. The fully developed pipe flow case $\text{Re}_\tau = 460$.

to the low-fluctuation directions to reduce the difference. However, it can not take away the difference totally. Therefore, the mainstream direction fluctuation is much larger than those in the other two directions.

Figure 7.1c shows the main-flow direction instantaneous velocity fluctuations in a cross section of the pipe. One can clearly observe the ejection events from the wall toward the pipe center, and low-speed streaks will form downstream these ejection

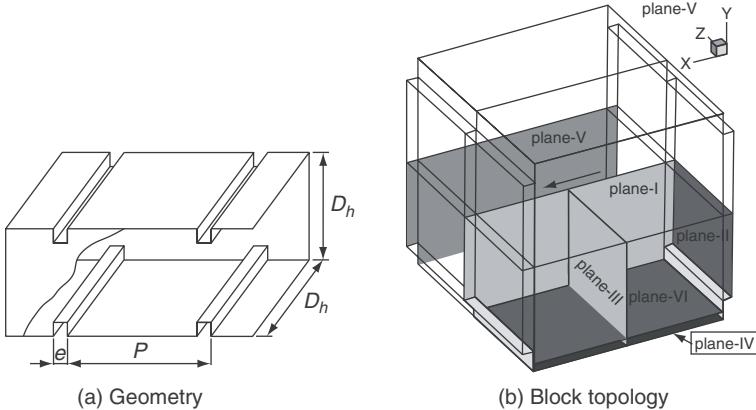


Figure 7.2. The computation geometry.

events. These ejection events and streaks produced the majority of the turbulence. Approximately, there are around six ejection events in the circumferential direction. Normally the spanwise distance for the ejection events are around $s+ = 100$. Figure 7.1d shows a similar picture from the DNS of Eggels et al. [11].

Figure 7.1e shows the out-of-plane instantaneous velocity in a plane through the pipe centerline. Superimposed on top of the small-scale turbulence in the core region of the flow, large scale structure, on the scale of the pipe diameter, is clearly seen. Near the pipe walls, the ejection events described earlier are seen as the dark and light regions that leave the pipe walls at an angle of approximately 20 degrees.

In general, we observed good agreement with the DNS data. This is true for both mean and rms results.

7.3.2 Transverse ribbed duct flow

The computational geometry is schematically shown in Figure 7.2a, and the computational block topology is illustrated in Figure 7.2b. The square duct with hydraulic diameter D_h is roughened by periodically mounted ribs of size e , $e/D_h = 0.1$, and the pitch is $10e$. The V-shape incline angle may be 90-degree, 60-degree, or 45-degree. When it is 90-degree, transverse ribs prevail. The simulated Reynolds number is small, around 4,000, turbulent flow is ensured. The selection of a low Re is mainly because a relatively small cell number can be used, and the main features of the turbulence field are not different from the high-Re cases. In addition, the contradiction in the experiments is not Re-dependent.

Flow field

Figure 7.3a shows the mean streamlines in the symmetry plane. The streamlines are characterized by four recirculation zones, accompanying flow separation and

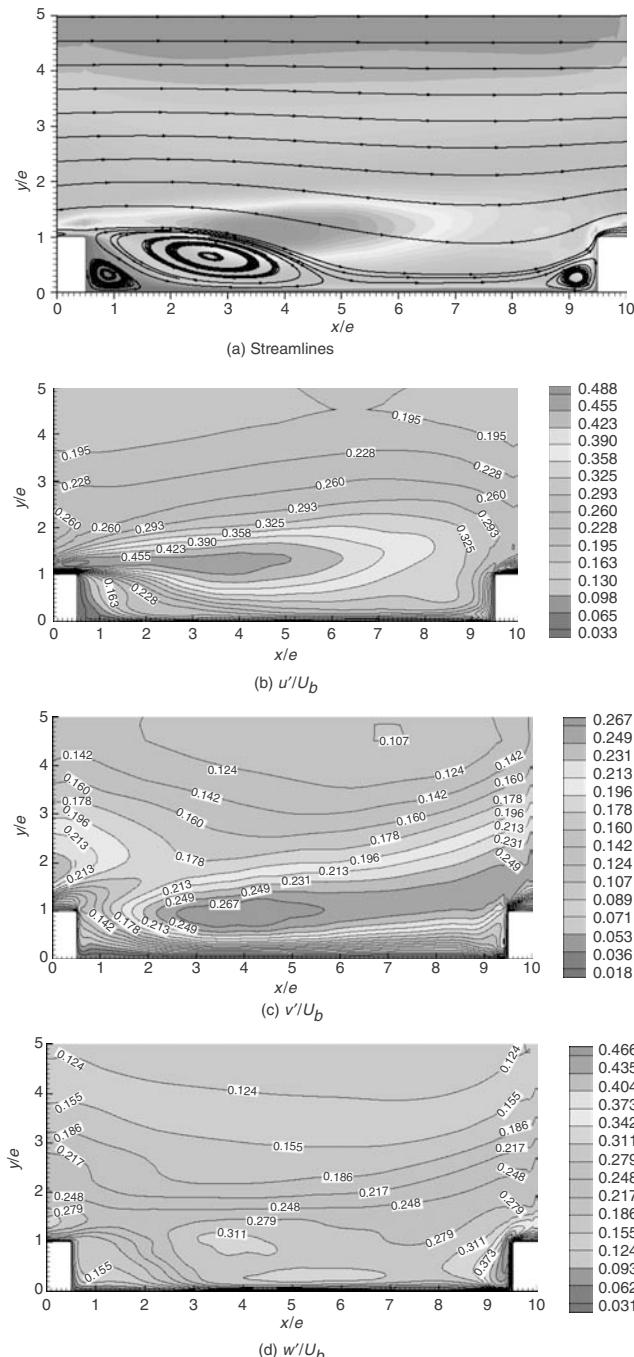


Figure 7.3. Streamlines and contours of the turbulent Reynolds stresses in the symmetry plane (plane-I).

reattachment. The sudden expansion downstream of a rib leads to the generation of a wide recirculation vortex associated with a strong shear layer. Together with this main recirculation, a second smaller separation is also identified, located in the corner between the downstream face of the rib and the bottom wall. After reattachment, a new boundary layer grows; the flow is accelerated by the main stream until it impinges upon the next rib. Finally, a fourth vortex is identified on top of the rib.

Figure 7.3b shows contours of streamwise turbulence intensity u'/U_b . The peak values of u'/U_b are located at the rib top. This is due to acceleration and strong shear at the rib top. Some local maxima of u'/U_b values are found at the top of the main separation zone behind the upstream rib. They are related to the shear layer originating from the rib edge and located above the separation region. Contours with high u'/U_b values are closer to the ribbed wall.

Figure 7.3c shows the y -direction velocity fluctuations. High v'/U_b values are found above the main separation zone inside the cavity between two successive ribs. v'/U_b is small near the bottom wall in the cavity, indicating weak vertical turbulent motions.

Figure 7.3d shows that the spanwise turbulence intensity, w'/U_b , is enhanced by the ribs. The peak value lies at the lower corner in front of the rib, where stream wise flow is blocked by the rib. A local maximum is also found in the main separation zone behind the upstream rib. The w'/U_b maximum is similar to the u'/U_b peak. High w'/U_b values are found upstream the rib, near the reattachment point in the cavity, and close to the rib front edge.

In general, very strong anisotropy is observed in the distribution of Reynolds stresses in three directions. This makes it very difficult to predict it by using linear-eddy-viscosity models.

7.3.3 V-shaped ribbed duct flow

The solver was also employed to simulate the duct flow roughened with 60-degree-inclined V-shape ribs. The heat transfer coefficients are calculated locally, and compared with the average value. The deviations are displayed with contours to help explain the discrepancy between different experimental results.

Figure 7.4 shows the deviation of the local Nu from the average Nu. The reason for the experimental contradiction can be explained from this picture. If we use one thermocouple (TC) to measure the temperature representing the surface; ideally, we should put the TC at the places, where the averaged value is located, i.e., the 0%-deviation contour lines. For the downstream pointing V-shaped ribs, if one puts the TC at the center of the surface, one will obtain a reasonably accurate average value (Figure 7.4a), because the value at the center of the surface is very close (around -5% underprediction) to the average value. For the upstream pointing V-shaped ribs, however, if one puts the TC at the center of the surface, one will obtain an over predicted averaged value (Figure 7.4b), because the value at the center of the surface is far (around 30% over prediction) from the averaged value.

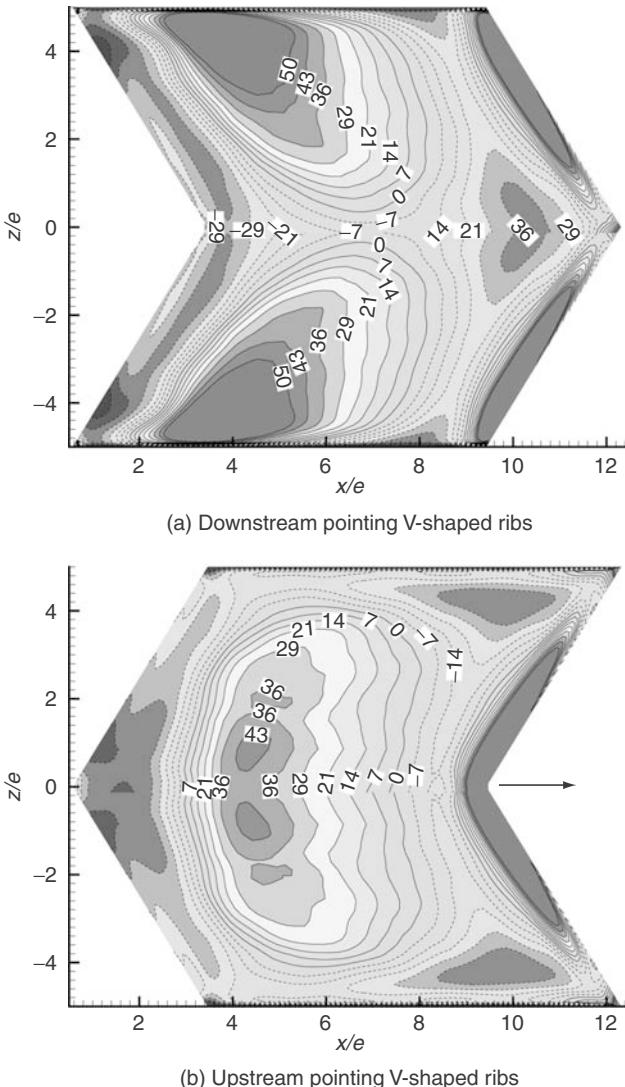


Figure 7.4. Deviation (%) from the average Nu.

With the liquid crystal (LC) method, the numbers of the sampling points for heat transfer coefficients are doing much better to correctly represent anisotropic heat transfer behavior. Therefore, correct conclusions are more likely obtained.

In summary, we can conclude that the LC experimental data in [4] and [5] are more reliable than the thermocouple data in [2] and [3]. The main reason is that the precise temperature distribution is important. The downstream pointing V-shaped ribs are providing better heat transfer efficiency.

7.4 Conclusions

The LES method implemented in the present solver has good accuracy for predictions of heat transfer and fluid flow.

Based on the simulation for the V-shaped ribbed duct flow, we have found that the LC experimental data are more reliable than those of TC data, which is the reason for the discrepancy of the experimental conclusions. It further confirmed that the V-shaped ribs pointing downstream are performing better in heat transfer enhancement.

However, for general application of LES in heat transfer simulations, there are remaining difficulties:

1. Treatment of near-wall turbulence: The near-wall turbulence is highly turbulent and small scaled (both in time and space), and plays a too significant role in heat transfer predictions to be neglected. These small scales normally cannot be captured by the simulations oriented for large scales. Therefore, subgrid models are vital here. Consequently, SGS models would be very important for heat transfer problems. Some numerical error may play a role here, but may only have limited success. Some SGS models seem promising, such as WALE (wall-adapting local eddy-viscosity) and the dynamic Smagorinsky model (DSM).
2. Computational cost: It takes long time to obtain enough LES statistic samples, which imposes a heavy burden on the computational speed and capacity. Some RANS calculations may be finished in minutes, but it may take days for a comparable LES calculation.

With the development of even faster and larger computer and quicker communication hardware, at the same time dipping prices, these computational costs will become less significant, but the software level parallelization is still somewhat left behind.

In general, we can foresee advancement of computational power and consequently the SGS model will become less important at the end, since mesh can be finer and time steps can be smaller, and gradually come close to the DNS level, although the better understanding of near wall turbulence will speedup this procedure. Robust and accurate schemes and algorithms will always be an important part of research.

As a summary, on today's parallel computer clusters built with commodity components, realistic heat transfer problems can be resolved with LES at an acceptable cost and decent accuracy, while the advancement of the computational technologies will make the solution faster and more accurate.

Acknowledgment

The Swedish Energy Agency financially supported this research work.

References

- [1] Jia, R., Saidi, A., and Sundén, B. Heat transfer enhancement in square ducts with V-shaped ribs, *ASME J. Turbomachinery*, 125(4), pp. 788–791, 2003.
- [2] Han, J. C., Zhang, Y. M., and Lee, C. P. Augmented heat transfer in square channels with parallel, crossed, and V-shaped angled ribs, *ASME J. Heat Transfer*, 113, pp. 590–596, 1991.
- [3] Olsson, C. O., and Sundén, B. Experimental study of flow and heat transfer in rib-roughened rectangular channels, *Exp. Therm. Fluid Sci.*, 16(4), pp. 349–365, 1998.
- [4] Taslim, M. E., Liu, T., and Kercher, D. M. Experimental heat transfer and friction in channels roughened with angled, V-shaped, and discrete ribs on two opposite walls, *ASME J. Turbomachinery*, 118(1), pp. 20–28, 1996.
- [5] Gao, X., and Sundén, B. Heat transfer and pressure drop measurements in rib-roughened rectangular ducts, *Exp. Therm. Fluid Sci.*, 24(1–2), pp. 25–34, 2001.
- [6] Boris, J. P., Grinstein, F. F., Oran, E. S., and Kolbe, R. L. New insights into large eddy simulation, *Fluid Dyn. Res.*, 10, pp. 199–208, 1992.
- [7] Smagorinsky, J. General circulation experiments with the primitive equations. I. The basic experiment, *Mon. Weather Rev.*, 91, pp. 99–164, 1963.
- [8] Jia, R., and Sundén, B. A Multi-block implementation strategy for a 3D pressure-based flow and heat transfer solver, *Numer. Heat Transfer B*, 44(5), pp. 457–472, 2003.
- [9] Jia, R., and Sundén, B. Parallelization of a multi-blocked CFD code via three strategies for fluid flow and heat transfer analysis, *Comput. Fluids*, 33, pp. 57–80, 2004.
- [10] Jia, R., Turbulence modelling and parallel solver development relevant for investigation of gas turbine cooling processes, Doctoral thesis, Department of Energy Sciences, Lund Institute of Technology, 2004.
- [11] Eggels, J. G. M., Unger, F., Weiss, M. H., Westerweel, J., Adrian, R. J., Friedrich, R., and Nieuwstadt, F. T. M. Fully developed turbulent pipe flow: A comparison between direct numerical simulation and experiment, *J. Fluid Mech.*, 268, pp. 175–209, 1994.

8 Recent developments in DNS and modeling of turbulent flows and heat transfer

Yasutaka Nagano and Hiroyuki Hattori

Department of Mechanical Engineering, Nagoya Institute of Technology, Japan

Abstract

The purpose of this chapter is to review recent studies on direct numerical simulation (DNS) and turbulence models from the standpoint of Computational Fluid Dynamics (CFD) and Computational Heat Transfer (CHT). Trends in recent DNS research and its role in turbulence modeling are discussed in detail. First, the instantaneous and the Reynolds-averaged governing equations for flow and energy are written in rotation and nonrotation frames. Secondly, as a typical recent DNS using the high-accuracy finite-difference method, the detailed DNS results of velocity and thermal fields in turbulent channel flow with transverse-rib roughness are presented. The problem is very important for elucidating the structure, which affects heat transfer enhancement. Thirdly, our recent DNS of rotating channel flows with various rotating axes and heat transfer is described. Rotating flows with heat transfer are encountered in many applications relevant to engineering, such as in gas turbines. This DNS was conducted using the spectral method. Therefore, two significant types of DNSs are presented in this chapter. Finally, on the basis of DNSs, much emphasis is placed on evaluating the performance of various kinds of turbulence models used in the velocity and thermal fields. Then, a new turbulence model is proposed for the accurate prediction of rotating turbulent flow with heat transfer. We should, however, bear in mind that a universal turbulence model can never be established without further development of DNS.

Keywords: Turbulence, Heat Transfer, DNS, Turbulence Model, Wall Roughness, Rotation

8.1 Introduction

Various attempts to analyze turbulence phenomena using available computer techniques have been made by many researchers. During the past three decades, direct numerical simulation (DNS) of three-dimensional time-dependent turbulent flows

has become a powerful tool for in-depth investigation of the structure of turbulent shear flows. This progress is mainly due to the considerable improvement in computational power of supercomputers and the great efficiency of numerical methods. With the advent of new massively parallel computers, this computational power is still continuously increasing and DNS of moderately high Reynolds number flows in complex geometries has been becoming a reality. One distinct advantage of the DNS technique is ready and accurate information such as the pressure fluctuation p and dissipation rate ε of turbulent kinetic energy k , which can never be obtained from the measurements that have played the central role in traditional turbulence research.

DNS has also contributed to the development of turbulence models. Especially, two-equation (i.e., $k - \varepsilon$ for a velocity field and $k_\theta - \varepsilon_\theta$ for a thermal field, where k and ε are the turbulent kinetic energy and its dissipation rate, and k_θ and ε_θ the intensity of fluctuating temperature and its dissipation rate, respectively) and second-moment closure (i.e., Reynolds stress and turbulent heat flux) models have made rapid progress with the aid of DNS. It is well known that the standard two-equation and second-moment closure models provide comparatively satisfactory results for simple flow fields, whereas in more complicated flow results are not as satisfactory as initially expected. Thus, constructing a more sophisticated turbulence model applicable to various types of flows is the central issue.

In this chapter, we first discuss the trend in recent DNS research and its applicability to turbulence model construction or evaluation. Then, we show two types of DNSs, one is conducted using the highly accurate finite-difference method and the other performed using the spectral method. Both DNSs mimic complex turbulent flows with a high degree of accuracy. And finally, we intend to introduce our latest turbulence models used in the velocity and thermal fields of technological interest. In addition, we introduce some new methods of evaluation and construction of turbulence models and attempt to comment on future research.

8.2 Present State of Direct Numerical Simulations

At present, DNS is widely regarded as a new method in place of experiments using apparatus. Since various information that can never be obtained from experiments is immediately and accurately supplied by DNS, DNS is expected to help establish new turbulence theories, analyze turbulence phenomena, and construct universal turbulence models. To establish turbulence theories and analyze turbulence phenomena, Robinson [1] conducted thorough investigations using the DNS database on boundary layer flow. Previous DNS studies were compiled and reviewed by Kasagi [2], especially the relation between velocity and thermal fields.

From the viewpoint of turbulence modelers, there is much more interest in the role of DNS in constructing a universal turbulence model. For example, detailed information on fundamental turbulence quantities such as the pressure fluctuation p and the dissipation rate ε have brought to light many problems inherent to the existing turbulence models and paved the way for the construction of a universal turbulence model. Also, information on budget profiles of various transport equations have encouraged active modeling of the elemental processes of turbulence [3].

As a result, DNS has emerged as the most important tool for the development of turbulence model.

On the other hand, some new attempts to construct the turbulence model with the aid of the DNS databases have been suggested. Nagano and Shimada [4], for example, have devised a method for evaluating a modeled-transport equation itself using DNS data. This method also has the advantage of individually estimating elemental processes of turbulence.

Early DNS research mainly aimed to consolidate the foundation of numerical techniques under simple flow conditions, e.g., homogeneous shear flow [5], two-dimensional channel flow [6, 7], and turbulent boundary layer flow [8]. Recent DNS research focuses on the analyses of turbulence phenomena (e.g., mechanisms of turbulence generation and destruction, the interaction between inner and outer layers, and the interaction between velocity and thermal fields). Moreover, DNS has made it possible to construct databases under more complicated flow conditions. Examples include adverse-pressure-gradient boundary layer flow [9, 10], backward-facing step flow [11], channel flow with riblets [12, 13], rotating channel flow [14]–[18], square duct flow [19], channel flow with injection and suction [20], channel flow under stable and unstable stratifications [21], impinging jet [22], and stable and unstable turbulent thermal boundary layer [23], all of which introduce highly advanced DNS techniques. DNS databases presented above are now available for assessing various kinds of turbulence models.

8.3 Instantaneous and Reynolds-Averaged Governing Equations for Flow and Heat Transfer

In DNS of turbulent motion, the three-dimensional unsteady Navier–Stokes equations for an incompressible fluid are solved directly for the instantaneous values of the velocity components and pressure. Here, we consider a turbulent heated channel flow with rotation at a constant angular velocity as shown in Figure 8.1. The governing equations for an incompressible rotating channel flow in reference frame rotational coordinates can be described in the following dimensionless forms:

$$\frac{\partial u_i^+}{\partial x_i^+} = 0 \quad (1)$$

$$\frac{Du_i^+}{Dt^+} = -\frac{\partial p_{\text{eff}}^+}{\partial x_i^+} + \frac{1}{\text{Re}_\tau} \frac{\partial^2 u_i^+}{\partial x_j^+ \partial x_j^+} - \varepsilon_{ik\ell} \text{Ro}_{\tau k} u_\ell^+ + \Pi_1^* \quad (2)$$

$$\frac{D\theta^+}{Dt^+} = \frac{1}{\text{Pr Re}_\tau} \frac{\partial^2 \theta^+}{\partial x_j^+ \partial x_j^+} \quad (3)$$

where $t^+ (= tu_\tau / \delta)$ is the time, $u_i^+ (= u_i / u_\tau)$ is the instantaneous velocity in $x_i^+ (= x_i / \delta)$ direction, $\theta^+ (= \theta / \Delta\Theta)$ is the instantaneous temperature. These are normalized by the friction velocity, u_τ , the channel half width, δ , and the temperature difference, $\Delta\Theta (= \Theta_H - \Theta_c)$, between upper (Θ_c) and lower (Θ_H) walls.

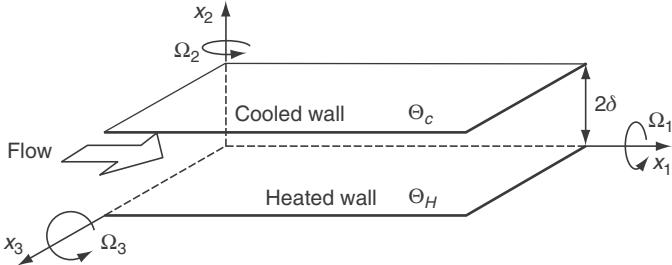


Figure 8.1. Heated rotating channel with arbitrary rotating axes.

The Reynolds-, the rotation-, and Prandtl numbers are respectively defined as $\text{Re}_\tau = u_\tau \delta / \nu$, $\text{Ro}_{\tau k} = 2\Omega_k \delta / u_\tau$, and $\text{Pr} = \nu / \alpha$, where ν is the kinematic diffusivity for momentum, Ω_k is the angular velocity with x_k axis and α is the thermal diffusivity for heat. The centrifugal force can be included in the effective pressure $p_{\text{eff}}^+ (= p_s^+ - \text{Ro}^2 r_c^2 / 8)$, if fluid properties are constant, where p_s^+ is the normalized static pressure, $\text{Ro} = (\text{Ro}_j \text{Ro}_j)^{1/2}$ is the absolute rotation number, and r_c is the dimensionless distance from the rotating axis [17]. Π_1^* is the nondimensional constant mean pressure gradient to maintain a flow, which is implemented only for a flow with asymmetrically roughened walls at $\text{Ro} = 0$ to make calculations more stable. As described later, the spectral method [16] is used for the DNSs of rotating channel flows. The computational conditions are listed in Table 8.1. The boundary conditions are non-slip conditions for the velocity field and different constant temperatures for the thermal field on the walls, $\Delta\Theta = \text{constant}$, as well as periodic conditions in the streamwise and spanwise directions.

In turbulence modeling, the so-called Reynolds-averaged equations are usually employed, and turbulence models based on Reynolds-averaged Navier–Stokes equation are often referred to as RANS models. The governing equations for RANS turbulence model can be written as follows [24]:

$$\frac{\partial \bar{U}_i}{\partial x_i} = 0 \quad (4)$$

$$\frac{D \bar{U}_i}{Dt} = -\frac{1}{\rho} \frac{\partial \bar{P}_{\text{eff}}}{\partial x_i} + \frac{\partial}{\partial x_j} \left(\nu \frac{\partial \bar{U}_i}{\partial x_j} - \bar{u}_i \bar{u}_j \right) - 2\varepsilon_{ik\ell} \Omega_k \bar{U}_\ell \quad (5)$$

$$\frac{D \bar{\Theta}}{Dt} = \frac{\partial}{\partial x_j} \left(\alpha \frac{\partial \bar{\Theta}}{\partial x_j} - \bar{u}_j \bar{\theta} \right) \quad (6)$$

$$\bar{u}_i \bar{u}_j = \frac{2}{3} k \delta_{ij} - 2C_0 \nu_t S_{ij} + \text{Nonlinear terms} \quad (7)$$

$$\bar{u}_j \bar{\theta} = -\alpha'_{jk} \frac{\partial \bar{\Theta}}{\partial x_k} + \text{Nonlinear terms} \quad (8)$$

where C_0 is a model constant, $k (= \bar{u}_i \bar{u}_i / 2)$ is turbulent kinetic energy, $S_{ij} [= (\partial \bar{U}_i / \partial x_j + \partial \bar{U}_j / \partial x_i) / 2]$ is the strain tensor, \bar{U}_i is the mean velocity in x_i

Table 8.1. Methods for direct numerical simulation

Grid	Regular grid
Time advancement	
Viscosity term	Crank–Nicolson method
Other terms	Adams–Bashforth method
Spatial scheme	Spectral method
Grid points ($x_1 \times x_2 \times x_3$)	$64 \times 65 \times 64$
Computational volume	$4\pi\delta \times 2\delta \times 2\pi\delta$

direction, $\alpha_{jk}^t = -C_{t0}\overline{u_j u_k}\tau_m$ is the anisotropy eddy diffusivity for heat, $\overline{\Theta}$ is the mean temperature, $\nu_t = C_\mu f_\mu(k^2/\varepsilon)$ is the eddy diffusivity for momentum, ρ is density, and τ_m is the hybrid/mixed time scale.

The transport equations of turbulence quantities, i.e., turbulent kinetic energy, k , dissipation rate of turbulent kinetic energy, ε , temperature variance, $k_\theta (= \overline{\theta}/2)$, and dissipation rate of temperature variance, ε_θ , which compose the eddy diffusivities are given as follows [16]:

$$\frac{Dk}{Dt} = \nu \frac{\partial^2 k}{\partial x_j \partial x_j} + T_k + \Pi_k + P_k - \varepsilon \quad (9)$$

$$\frac{D\varepsilon}{Dt} = \nu \frac{\partial^2 \varepsilon}{\partial x_j \partial x_j} + T_\varepsilon + \Pi_\varepsilon + \frac{\varepsilon}{k} (C_{\varepsilon 1} P_k - C_{\varepsilon 2} f_\varepsilon \varepsilon) + E + R \quad (10)$$

$$\frac{Dk_\theta}{Dt} = \alpha \frac{\partial^2 k_\theta}{\partial x_j \partial x_j} + T_{k_\theta} + P_{k_\theta} - \varepsilon_\theta \quad (11)$$

$$\begin{aligned} \frac{D\varepsilon_\theta}{Dt} = & \alpha \frac{\partial^2 \varepsilon_\theta}{\partial x_j \partial x_j} + T_{\varepsilon_\theta} + \frac{\varepsilon_\theta}{k_\theta} (C_{P1} f_{P1} P_{k_\theta} - C_{D1} f_{D1} \varepsilon_\theta) \\ & + \frac{\varepsilon_\theta}{k} (C_{P2} f_{P2} P_k - C_{D2} f_{D2} \varepsilon) + E_\theta + R_\theta \end{aligned} \quad (12)$$

where T_k , T_ε , T_{k_θ} , and T_{ε_θ} are turbulent diffusion terms, Π_k and Π_ε are pressure diffusion terms, and $P_k (= -\overline{u_i u_j} \partial \overline{U}_i / \partial x_j)$ and $P_{k_\theta} (= -\overline{u_j \theta} \partial \overline{\Theta} / \partial x_j)$ are production terms, E and E_θ are extra terms to correct the near-wall behaviors of ε and ε_θ , and R and R_θ are rotation-influenced additional terms. Note that, in the rotational coordinate system, the vorticity tensor, $\Omega_{ij} [= (\partial \overline{U}_i / \partial x_j - \partial \overline{U}_j / \partial x_i)/2]$, should be replaced with the absolute vorticity tensor in equations (7) and (8), i.e., $W_{ij} = \Omega_{ij} + \varepsilon_{mji} \Omega_m$ to satisfy the material frame indifference (MFI) [25].

8.4 Numerical Procedures of DNS

In general, the DNSs are performed using the two methods described briefly in the following. Each method has its own special feature. Thus, between the two approaches one should decide which to choose after careful consideration of problems to be solved.

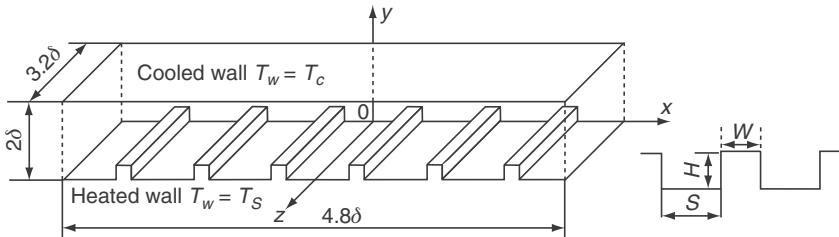


Figure 8.2. Channel with transverse-rib roughness and coordinate system.

8.4.1 DNS using high-accuracy finite-difference method

To solve a flow problem in complex geometries, a high-accuracy finite-difference method is the best choice. Figure 8.2 shows a schematic of the channel with transverse-rib roughness and the coordinate system used in the present study [13]. This type of flow is definitely complicated. The origin of the coordinate axes is located in the middle of the enclosure between the ribs. Table 8.2 summarizes some details of the high-accuracy finite-difference method. DNS was carried out with a constant mean pressure gradient to balance the wall shear stress on both walls. Note that a fully consistent and conservative finite-difference method is used for the convective term of the Navier–Stokes equation [26]. In order to solve the Poisson equation for the pressure, the standard SOR method is applied and the over-relaxation coefficient is set to 1.5. The adequate numbers of grid points are arranged in each direction for DNS so as to exactly capture the turbulent heat transfer as listed in Table 8.2. Especially, the grid resolution of spanwise direction, Δz^* , must be arranged smaller than 10.0. The time advancement is $\Delta t^* = 1 \times 10^{-4}$, and the total time steps needed for the statistical quantities to converge reasonably are 300,000. The numerical scheme used in this study is validated by comparing the statistical quantities, including the budget of the turbulent kinetic energy, in plane channel flow with those calculated by Nagano and Hattori [16], employing a spectral method as described next.

8.4.2 DNS using spectral method

Referring to Kim et al. [6], a fourth-order partial differential equation for v (wall-normal velocity component), a second-order partial differential equation for the wall-normal component of vorticity, and the continuity equation are solved to obtain the instantaneous flow field. The numerical method is fully described by Kim et al. [6]. That is, a spectral method is adopted with Fourier series in the streamwise and spanwise directions and Chebyshev polynominal expansion in the wall-normal direction. The collocation grid used to compute the nonlinear terms in physical space has 1.5 times finer resolution in each direction to remove aliasing errors (sometimes referred to as padding or 3/2-rule). For time integration,

Table 8.2. Computational methods

Channel with transverse-rib roughness	
Grid	Staggered grid
Coupling algorithm	Fractional step method
Time advancement	
Conductive term	Crank–Nicolson method
Other terms	Adams–Bashforth method
Spatial scheme	2nd-order central difference
Computational volume	$4.8\delta \times 2\delta \times 3.2\delta$
Grid points	$192 \times (96 + 36 \text{ or } 18) \times 96$
($x \times y$ (rib outside + rib inside) $\times z$)	
Grid resolution	$\Delta x^* = 3.75$ $\Delta y^* = 0.30 \sim 6.50$ $\Delta z^* = 5.0$

the second-order Adams–Bashforth and Crank–Nicolson schemes are used for the nonlinear and viscous terms, respectively (see Section 4.1).

8.5 DNS of Turbulent heat Transfer in Channel Flow with Transverse-Rib Roughness: Finite-Difference Method

DNS of turbulent heat transfer in channel flow with transverse-rib roughness was carried out using the high-accuracy finite-difference method [13], in which flow parameter and five types of wall roughness are arranged as listed in Table 8.3. The number of grid points in the y -direction in the enclosure is 36 for Cases 1, b, and c, and 18 for Cases 2 and 3 as indicated in Table 8.2. According to the classification of roughness [27, 28], Case 1 in Table 8.3 belongs to k -type roughness, and is similar shape to the experiment of Hanjalić and Launder [29]. In Case 2, the ribs are set to half the height of Case 1, and in Case 3, the height is set to half the height of Case 2. In the k -type roughness, the effect of the roughness is expressed in terms of the roughness Reynolds number, $H^+ = u_\tau H / v$. In Cases b and c, the height of the ribs is set the same as in Case 1, though the spacing of the ribs is varied systematically. Case c belongs to d -type roughness, in which the effect of the roughness cannot be expressed by H^+ . On the other hand, Case b is not regarded as the exact d -type, but d -like type roughness. Note that the two kinds of roughness are extreme versions and intermediate forms can exist [28].

8.5.1 Heat transfer and skin friction coefficients

Figures 8.3 and 8.4, respectively, show the skin friction coefficient and the Nusselt number against the Reynolds number. By taking into account the asymmetry [30]

Table 8.3. Flow and rib parameters

Re_τ	150			
Pr	0.71			
	H	W	S	Roughness type
Case 1	0.2 δ	0.2 δ	0.6 δ	k type
Case 2	0.1 δ	0.2 δ	0.6 δ	
Case 3	0.05 δ	0.2 δ	0.6 δ	
Case b	0.2 δ	0.4 δ	0.4 δ	d -like type
Case c	0.2 δ	0.2 δ	0.2 δ	

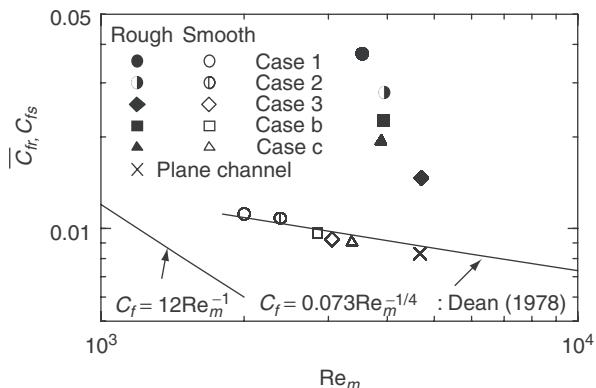


Figure 8.3. Distributions of skin friction coefficients.

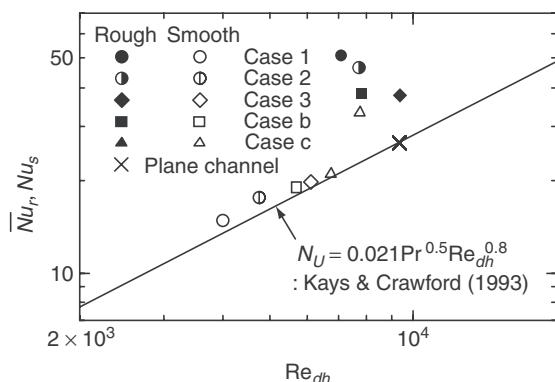


Figure 8.4. Distributions of Nusselt numbers.

in the flow field disturbed by the ribs, the skin friction coefficient and the Nusselt number at the rough wall are defined as follows:

$$\overline{C_{fr}} = \frac{2\overline{\tau_{wr}}}{\rho\langle U \rangle_r^2}, \quad \overline{Nu}_r = \frac{2\overline{q_{wr}}d}{(T_h - \langle T \rangle_r)\lambda} \quad (13)$$

where d indicates the distance between the wall and the location where the mean velocity becomes maximum, and $\langle \rangle$ denotes the bulk mean over the distance d . At the smooth wall, they are similarly defined as follows:

$$C_{fs} = \frac{2\tau_{ws}}{\rho\langle U \rangle_s^2}, \quad Nu_s = \frac{2q_{ws}(2\delta - d)}{(\langle T \rangle_s - T_c)\lambda} \quad (14)$$

The wall shear stress, τ_{ws} , at the (upper) smooth wall is estimated directly by the mean velocity gradient on the wall and then, $\overline{\tau_{wr}}$ at the (lower) rough wall is calculated from the balance of the imposed pressure gradient:

$$\overline{\tau_{wr}} = -2\delta \frac{dp}{dx} - \tau_{ws} = 2\tau_{w0} - \tau_{ws} \quad (15)$$

The above-defined wall shear stress at the rough wall, $\overline{\tau_{wr}}$, includes the pressure drag (form drag) as well as the viscous drag:

$$\overline{\tau_{wr}} = \tau_{wp} + \tau_{vv} \quad (16)$$

On the other hand, because the fully developed state is assumed for the thermal condition in the present study, there is no increase in the enthalpy of the flowing fluid in the streamwise direction. Thus, the overall heat flux at the rough wall, $\overline{q_{wr}}$, is equal to the absolute value of that at the smooth wall, q_{ws} , which can be estimated directly by the mean temperature gradient on the wall. In order to compare the results with the correlation curve by Kays and Crawford [31] for the pipe flow, the length scales of the Reynolds and Nusselt numbers are doubled in Figure 8.3. In those figures, the DNS results in the plane channel flow ($Re_\tau = 150$) calculated by the spectral method [16] are also included to compare with the results of the present DNS using the high-accuracy finite-difference method.

From Figures 8.3 and 8.4, it can be seen that the skin friction coefficient and the Nusselt number at the rough wall become very large in comparison with those at the corresponding smooth wall, where these quantities are well represented by the correlation curves. Moreover, from Figure 8.3, even though the imposed pressure gradient is the same as in the smooth wall, the rib decreases the bulk mean velocity, so the Reynolds number heavily depends on the rib configuration; in Case 1, where the rib is highest in the k -type roughness walls, the Reynolds number is the smallest among them. In the present study, Case 3 is found to be the lowest in drag. On the other hand, in the d -like type roughness (Cases b and c), the heat transfer augmentation is smaller than that in the k -type roughness with the same roughness height (Case 1).

Table 8.4. Coefficients for heat transfer rate at rough wall

	$(\overline{St}_r/\overline{C}_{fr})/(St_0/C_{f0})$	\overline{C}_{fr}/C_{f0}	\overline{Nu}_r/Nu_0
Case 1	0.61	3.93	2.41
Case 2	0.71	2.87	2.04
Case 3	0.85	1.66	1.42
Case b	0.69	2.44	1.67
Case c	0.69	2.10	1.46

Table 8.4 shows the Stanton number, the skin friction coefficient, and the Nusselt number, which are divided by the estimated smooth-wall values from the respective correlation curve [31, 32] for the same Reynolds number. In the *k*-type classification, Case 1 enhances heat transfer most. However, from the viewpoint of the heat transfer characteristic including the drag, if we compare the Stanton number divided by the skin friction coefficient $\overline{St}_r/\overline{C}_{fr}$, from Table 8.4, Case 1 enhances the heat transfer more than the smooth wall, but the overall characteristics of heat transfer do not improve because of the large drag. However, Case 3, the *k*-type with the lowest ribs, promotes the heat transfer with very low drag. This case is the most efficient from the standpoint of overall heat transfer performance including the drag. In the *d*-like type roughness, the heat transfer characteristic cannot be improved regardless of the rib spacing. The stagnation region in the enclosure becomes larger than that in the *k*-type roughness (not shown here), and the deterioration in heat transfer has been profound there. Thus, in the following, we discuss the effects of the height of the rib on the statistical characteristics of thermal property in the *k*-type roughness, which includes a rib configuration which promotes the heat transfer.

8.5.2 Velocity and thermal fields around the rib

To visualize the velocity and thermal fields around the rib, Figures 8.5 and 8.6 show the mean streamlines estimated from the mean velocity profiles and contour lines of the mean temperature distributions. The averages are taken only with respect to the spanwise direction in these figures. Two-dimensional vortices exist in the enclosures between the ribs in Cases 1 and 2, and their shapes are different in each case. In Case 1, the center of the two-dimensional vortex is biased to the upstream side of the rib and becomes asymmetric, whereas in Case 2, the center of the vortex is biased to the downstream side of the rib. Moreover, in Case 3, the small two-dimensional vortices are located on both sides of the rib, and the flow reattachment is seen in the enclosure.

On the other hand, from the contour lines of the mean temperature, around the front corner of the rib, the spacing between the lines becomes smaller, making for very active heat transfer there. In the enclosure between the ribs, the contour lines are distorted corresponding to the streamlines and are not parallel to the bottom wall. Especially, in Case 3, it is confirmed that the mean temperature contour lines

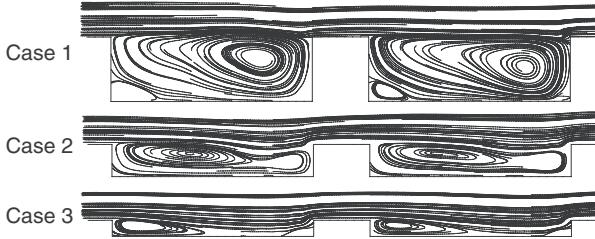


Figure 8.5. Streamlines of mean velocity averaged in the spanwise direction. Flow is left to right.

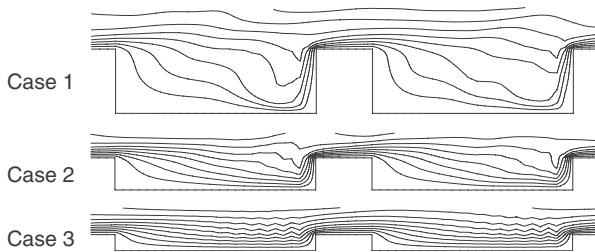


Figure 8.6. Contour lines of mean temperature averaged in the spanwise direction. The interval between successive contour level is 0.02. Flow is left to right.

are densely distributed, causing more enhanced heat transfer. On the upstream side of the ribs, the wavy patterns of temperature can be seen. This phenomenon is a characteristic specific to a temperature field near the upstream side of the rib.

Figures 8.7 and 8.8 show the local skin friction coefficient, $C_{fr}(x)[=2\tau_{wv}(x)/\rho\langle U \rangle_r^2]$ and the local Nusselt number, $Nu_r(x)[=2q_{wr}(x)d/(T_h - \langle T \rangle_r)\lambda]$ along the bottom of the enclosure and the crest on the rough wall, divided by the values on the smooth wall. These mean quantities are obtained by averaging with respect to time, the spanwise direction and the streamwise period of the rib roughness. The local wall shear stress, $\tau_{wv}(x)$, and wall heat flux, $q_{wr}(x)$, are estimated directly from the gradients of mean velocity and temperature, respectively. Thus, the local shear stress shown in Figure 8.7 does not include the pressure drag see equation (16). By averaging the local wall shear stress along the bottom of the enclosure and the crest in the x -direction, the contributions of the viscous drag to the total drag of the rough-wall side are estimated at 4.5%, 1.1% and, 20.8%, in Cases 1, 2, and 3, respectively. Ashrafiyan and Andersson [33] reported the contribution of the viscous drag to the total drag was 2.5%, in their DNS calculation of the rib-roughened channel flow (the pitch to the height ratio, $(S + W)/H$, is 8, $Re_{\tau_0} = 400$). Leonardi et al. [34] have systematically investigated the effects of the pitch to the height ratio on the contributions from the viscous and pressure drags. They reported that

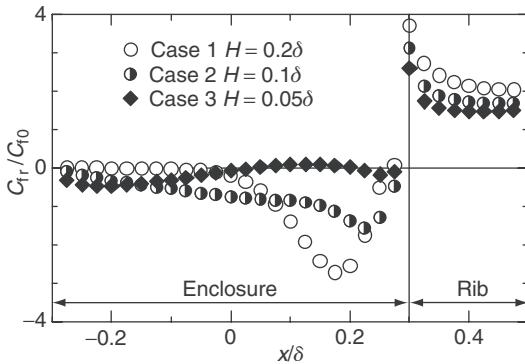


Figure 8.7. Profiles of local skin friction coefficients around rib along the horizontal wall with $x = 0$ at the middle of the enclosure.

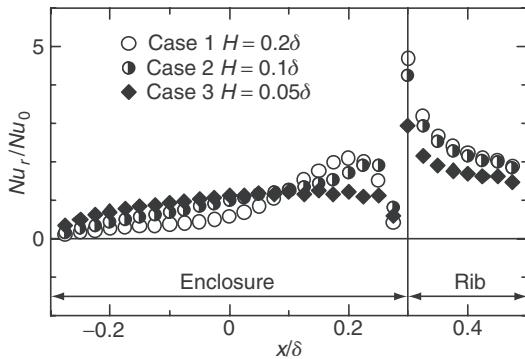


Figure 8.8. Profiles of local Nusselt number around rib along the horizontal wall with $x = 0$ at the middle of the enclosure.

the total drag is determined almost entirely by the pressure drag when the ratio is within the range from 6 to 20.

From Figure 8.7, the local skin friction coefficient becomes minimum in the enclosure and maximum at the front corner of the rib in Cases 1 and 2. On the other hand, in Case 3, although the local skin friction coefficient takes the maximum at the front corner of the rib, both a large overall decrease and a local peak in the enclosure cannot be seen; but it changes sign in the enclosure. The reattachment point is located at $x/\delta = 0.04$; thus the forward flow region is observed over $0.04 < x/\delta < 0.21$ until separation occurs in the enclosure between the ribs.

The local Nusselt numbers in Cases 1 and 2 distribute similarly to the *absolute value* of the local skin friction coefficient. However, in Case 3, the Nusselt number increases over the entire region in the enclosure. This situation is also observed near the reattachment region of the backward-facing step flows [35], and it is reported that the heat transfer coefficient reaches maximum there. Similarly,

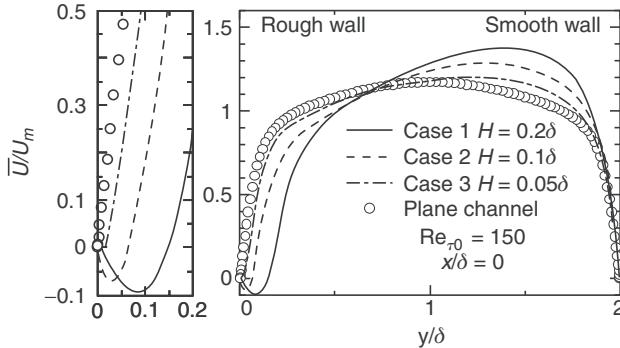


Figure 8.9. Profiles of mean velocity.

the local Nusselt number increases in approaching the reattachment point $x/\delta = 0.04$ and the following forward flow region ($0.04 < x/\delta < 0.21$). From the above results, it is confirmed that in Case 3, the heat transfer is promoted, yet with only a relatively small increase in the drag.

8.5.3 Statistical characteristics of velocity field and turbulent structures

In a velocity field of a channel flow with a rib surface, the flow motions in the enclosure between the ribs strongly affect the flow field above the ribs. We examined the various ways of averaging, e.g., average over the x - z plane, spanwise average at the middle of the enclosure ($x/\delta = 0$) and at the rib crest ($x/\delta = 0.4$). The differences were not seen along the streamwise direction except near the roughness element. Thus, in the following, the statistical quantities of turbulence in the middle of the enclosure ($x/\delta = 0$) are discussed.

Figure 8.9 shows the mean velocity normalized by the bulk velocity U_m . Because of the effects of the rib, the velocity decreases on the rough-wall side. The corresponding distributions of the Reynolds shear stress and turbulent kinetic energy are shown in Figures 8.10 and 8.11. As the rib height increases, turbulence is promoted, and both the Reynolds shear stress and the turbulent kinetic energy increase near the wall. This affects the region over the center of the channel. However, in the near-wall region of the opposite wall, there is only a small effect, in comparison with the results for the plane channel flow. Despite the negative value of the velocity gradient near the wall in the enclosure as seen in Figure 8.9, the Reynolds shear stress takes a positive value, thus confirming the occurrence of counter gradient diffusion. Because the production of the Reynolds shear stress is mainly maintained with the pressure-strain correlation and the pressure diffusion, no production from the mean shear is observed. An experimental study [29] indicated that the place where the Reynolds shear stress becomes zero and the mean velocity becomes maximum is different. However, the difference between them is very small; 2.0%, 2.8%, and 0.8% of a channel width for Cases 1, 2, and 3, respectively.

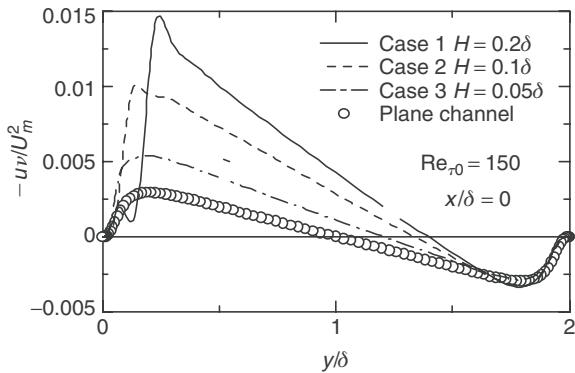


Figure 8.10. Profiles of Reynolds shear stress.

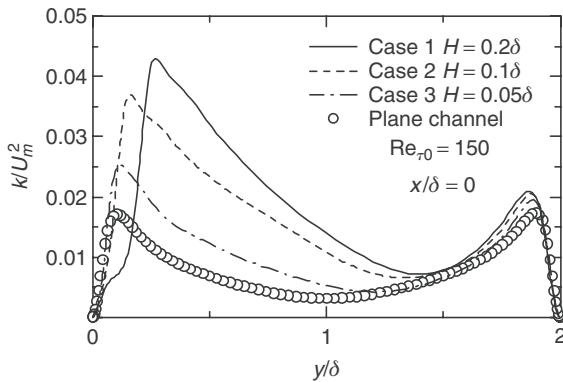


Figure 8.11. Profiles of turbulent kinetic energy.

Figure 8.12 shows the turbulence intensities normalized by the friction velocity on the rough wall. In all cases, the statistical quantities get together well in the upper region of the ribs. However, from Figure 8.11, the turbulence intensities increase relative to the bulk mean velocity. The budget of the turbulent kinetic energy is shown in Figure 8.13. In Case 1, the convection term is enhanced in comparison with the plane channel flow. This is because the mean vertical velocity component near the rib is increased, and also there is a variation in the turbulence in the streamwise direction. Moreover, the contributions from the turbulent and pressure diffusion terms are very large; in the enclosure between the ribs, there is no turbulence production from the mean shear, but turbulent transport maintains the turbulence there. In Case 3, the distribution of the budget is similar to the result in the plane channel, in comparison with Case 1, but the contributions from the turbulent transport (turbulent and pressure diffusions) are large.

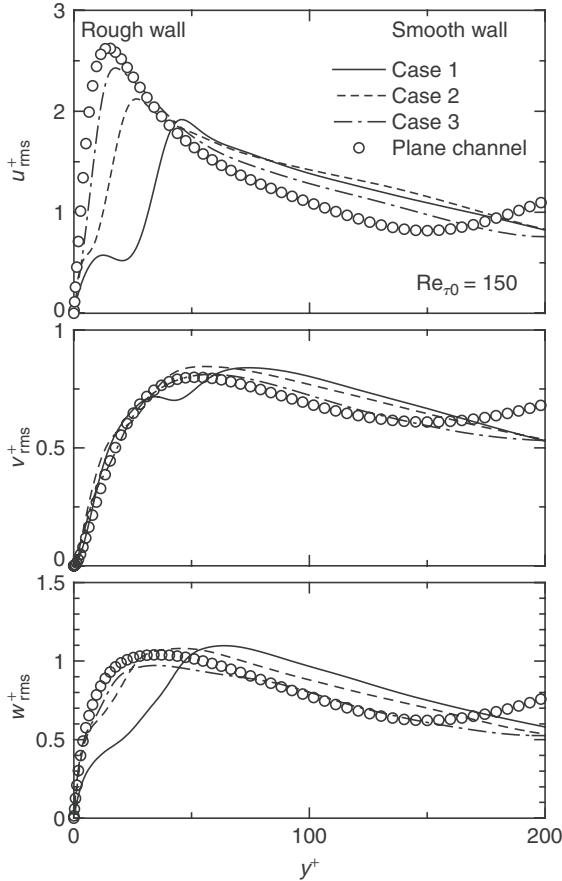


Figure 8.12. Distributions of turbulence intensities.

Next, to investigate the anisotropy of turbulence, Figure 8.14 shows the pressure-strain correlation terms see equation (23) which are important in the near-wall region. The pressure-strain correlation is affected because the flow impinges on the upstream side of the rib wall, and the pressure fluctuations increase over a wide region on the rough-wall side in comparison with the smooth-wall side. If we compare Case 1, where the rib is highest, with the plane channel flow, the respective components of the pressure-strain term become maximum in the region between the ribs, and a very active energy exchange occurs. In Case 1, the sign of Φ_{11} becomes positive at $y^+ \simeq 15$. In ordinary wall turbulence without external force, it is observed that in the near-wall region, the splatting phenomenon caused by v^2 gives the energy to $\overline{w^2}$ as seen in the DNS results [16]. However, in the enclosure between ribs, the energy does not redistribute to the $\overline{w^2}$ component. This can be attributed to the two-dimensional vortices existing in the enclosure, which promote

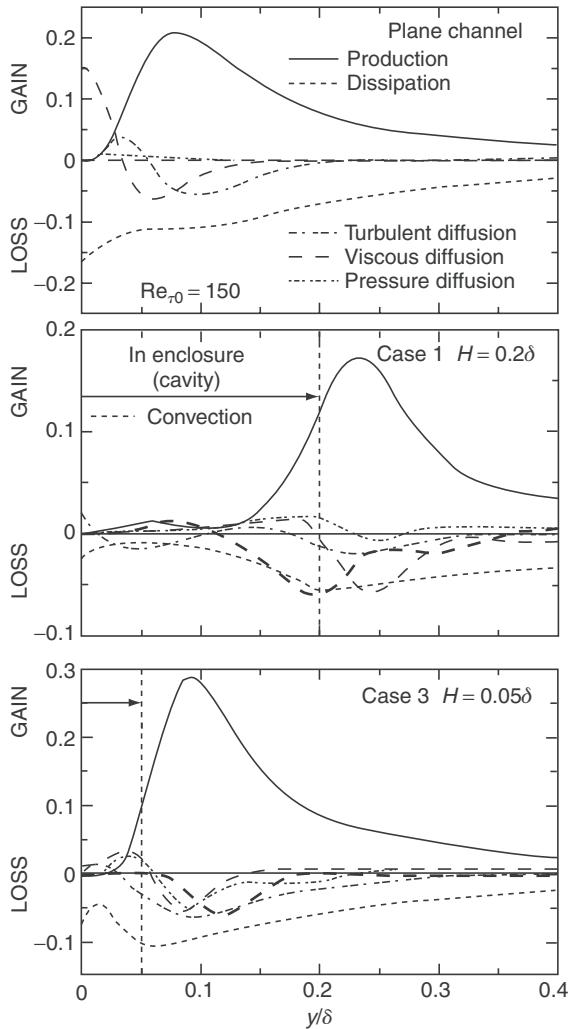


Figure 8.13. Budgets of turbulent kinetic energy.

the redistribution from the \bar{v}^2 to \bar{u}^2 components. As for Case 3, the effects from the two-dimensional vortex and the reattachment are combined and they affect the redistribution mechanism. As a result, as shown in Figure 8.12, in spite of the existing rough wall, the anisotropy of turbulence shows behavior similar to that in the smooth wall, and the roughness promotes the heat transfer with a relatively small drag.

Finally, Figure 8.15 shows the streamwise vortices educed with the second invariant tensors of the velocity gradient II [36, 37]. In each case, streamwise vortices are produced in the region above the ribs on the rough-wall side, and the structures spread over the region above the center of the channel. Moreover,

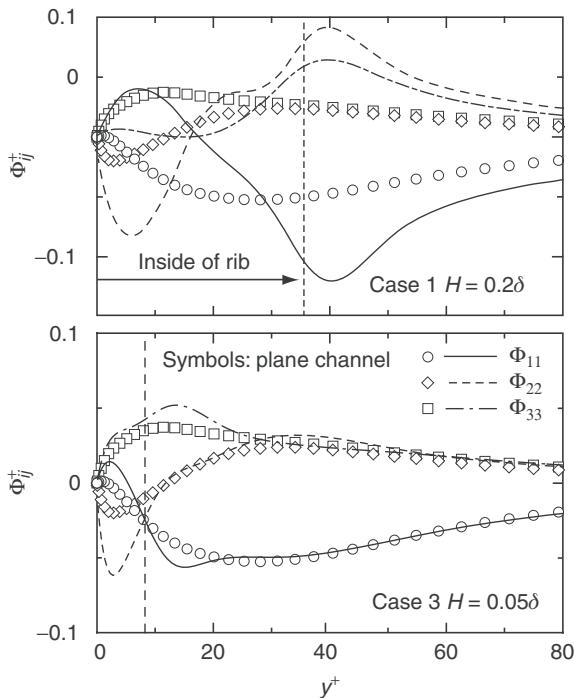


Figure 8.14. Distributions of pressure strain terms in transport equations for Reynolds stress.

if we compare Case 1 with Case 3, many more streamwise vortices are produced in Case 1, where the rib effects are strong. In the enclosure between the ribs, the streamwise vortices are seen in the whole region of the enclosure in Case 3. However, in Case 1, because the flow stagnates downstream of the rib surface, not many streamwise vortices are produced. This corresponds to the results of the distributions of the skin friction coefficient and the Nusselt number as seen in Figures 8.7 and 8.8.

8.5.4 Statistical characteristics of thermal field and related turbulent structures

For the thermal field, the turbulence statistics at the middle of the cross-sectional enclosure ($x/\delta = 0$) are discussed below. Figure 8.16 shows the mean temperature distribution normalized by the temperature difference ΔT_w . The effect of the rib causes the mean temperature to become asymmetric in all cases as for the mean velocity profiles, and the temperature increases in the major part of the channel. On the other hand, because of the rough-wall side effects, the region of the large temperature gradient extends to the smooth-wall side. Figure 8.17 shows the turbulent heat flux in the wall-normal direction. The heat transfer is activated from

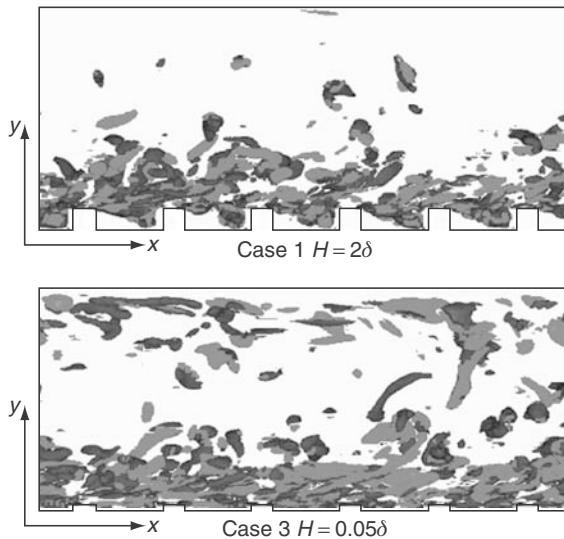


Figure 8.15. Vortex structures visualized by second invariant: $I\!I^* < -0.5$.

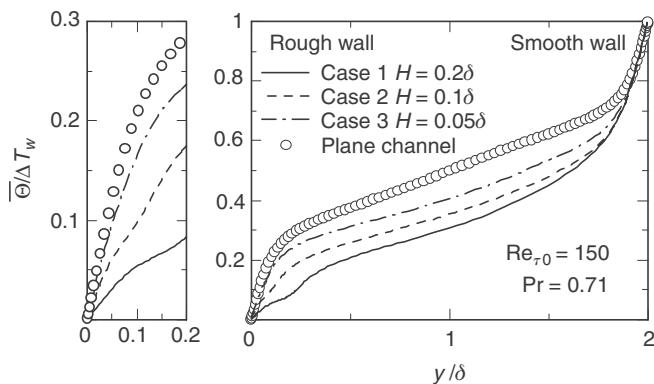


Figure 8.16. Profiles of mean temperature.

the enclosure with the ribs, but in the thermal field also there are counter gradient diffusions in Case 1.

Next, Figure 8.18 shows the rms intensities of temperature fluctuation, which decrease on the rough-wall side, where the velocity fluctuations are promoted. However, the temperature intensities increase on the smooth-wall side, where the turbulent velocity fluctuation is suppressed. Apparently, the contrary situations occur between the velocity and thermal fields on each side. To investigate the cause and effect, we calculated the production and turbulent diffusion terms of the transport equation for $\overline{\theta^2}/2$ see equation (11) in each case (Figure 8.19).

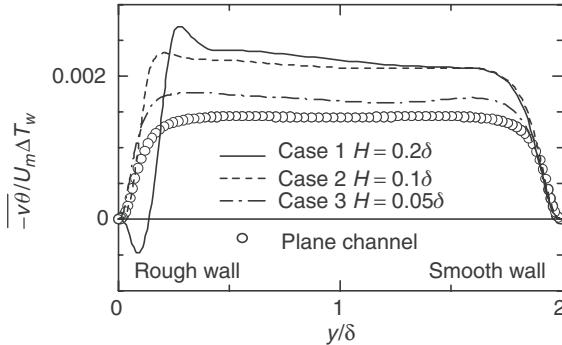


Figure 8.17. Profiles of wall-normal turbulent heat flux.

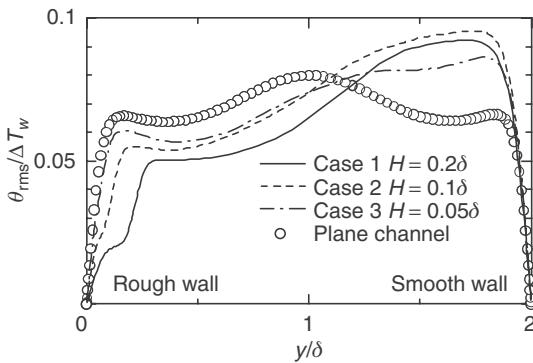


Figure 8.18. Profiles of rms intensities of temperature fluctuation.

The production terms on the smooth-wall side, where the temperature gradient is large and the wall-normal heat fluxes increase, contributes greatly in the region around the peak to the center of the channel. This is why the temperature intensities become large on the smooth-wall side. On the other hand, on the rough-wall side, because the contribution from the production decreases and the turbulence is maintained by the turbulent diffusion, the intensities of temperature fluctuations decrease, especially in Case 1.

In order to examine the relationship between the velocity and thermal fields, the distributions of the turbulent Prandtl number, Pr_t , are shown in Figure 8.20. The spanwise averaged Pr_t at the middle of the enclosure and of the rib crest are compared with that in the plane channel flow. Near the region above the rib, Pr_t becomes larger than that on the rib crest. However, it becomes smaller and then becomes larger again in the enclosure. Thus, the turbulent Prandtl number is not constant and the analogy between heat and momentum transfer cannot be expected near the rib roughness. On the other hand, on the smooth-wall side, the Pr_t distribution well corresponds to that in the plane channel flow.

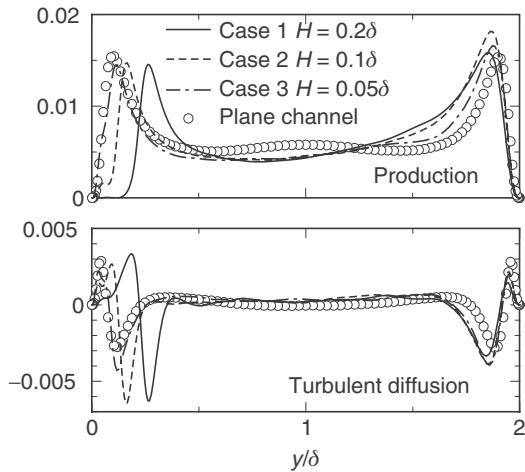


Figure 8.19. Production and turbulent diffusion terms in temperature variance budget.

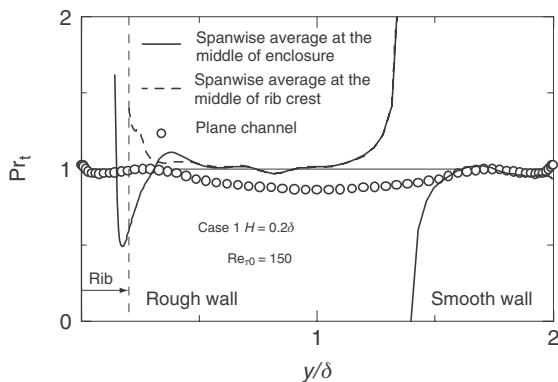


Figure 8.20. Turbulent Prandtl number.

Finally, to observe the spatial structures that contribute to the transport of the passive scalar, Figure 8.21 shows the streamlines, which are spatially averaged along the streamwise direction, and the temperature fluctuations in the $y-z$ cross section in the plane channel flow and Case 1. In the plane channel flow, the streamlines show the streamwise vortices in the near-wall region; the sharp variations in the temperature fluctuations associate with the vortices. On the other hand, in Case 1, large-scale vortex structures appear, which extend to the center of the channel from the enclosure between the ribs. With this vortex structure, the turbulent mixing becomes larger in the center of the channel, and the turbulence transport is promoted. This behavior is consistent with the above-mentioned statistical results, i.e., the increase in the heat transfer coefficient, the pronounced change in the mean

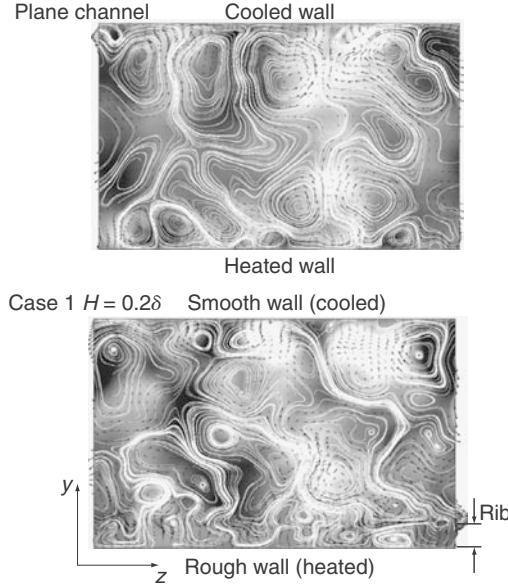


Figure 8.21. Streamwise-averaged streamlines and temperature fluctuations in $y-z$ plane: $\theta/\Delta T_w = -0.05$ (white) 0.05 (black).

temperature distributions, and the increase in the turbulent diffusion terms in the budget of temperature variance.

8.6 DNS of Turbulent heat Transfer in Channel Flow with Arbitrary Rotating Axes: Spectral Method

In order to determine improvements in the modeled expression of rotating, wall-bounded turbulent shear flows, DNSs of fully developed channel flows with streamwise, wall-normal, and spanwise rotation are carried out using the spectral method for the 15 cases indicated in Table 8.5 [18]. DNS results of cases for WNR1~WNR5 and STR1~STR5 are shown in Figures 8.22 and 8.23, respectively (Case 3 is not shown here). In these cases, the spanwise mean velocity appears to be caused by a rotational effect, which increases with the increase in rotation number in both cases. In particular, with the increase in the spanwise mean velocity of Case 1, streamwise mean velocity decreases due to the exchange momentum between the streamwise and spanwise velocities as in the following equations obtained from equation (5):

$$0 = -\frac{1}{\rho} \frac{\partial \bar{P}}{\partial x} + \frac{\partial}{\partial y} \left(\nu \frac{\partial \bar{U}}{\partial y} - \bar{u}\bar{v} \right) - 2\Omega \bar{W} \quad (17)$$

$$0 = \frac{\partial}{\partial y} \left(\nu \frac{\partial \bar{W}}{\partial y} - \bar{v}\bar{w} \right) + 2\Omega \bar{U} \quad (18)$$

Table 8.5. Computational conditions

Re_τ	150				
Pr	0.71				
Case 1: Wall-normal rotation					
	WNR1	WNR2	WNR3	WNR4	WNR5
$Ro_{\tau 2}$	0.01	0.02	0.05	0.1	0.3
Case 2: Streamwise rotation					
	STR1	STR2	STR3	STR4	STR5
$Ro_{\tau 1}$	1.0	2.5	7.5	10.0	15.0
Case 3: Spanwise rotation					
	SPR1	SPR2	SPR3	SPR4	SPR5
$Ro_{\tau 3}$	0.0	0.75	1.5	3.0	5.0

Therefore, to predict the flow of Case 1 exactly, the spanwise mean velocity should be reproduced by a turbulence model. Note that the Reynolds shear stress, \bar{uv} , tends to decrease remarkably with the increase in rotational number as shown in Figure 8.22b.

On the other hand, it is found from DNS that cases of streamwise rotating channel flow (Case 2) involve the counter gradient turbulent diffusion in the spanwise direction shown in Figure 8.23. In view of turbulence modeling, this fact clearly demonstrates that the linear “eddy diffusivity turbulence model” (EDM) and the quadratic “nonlinear eddy diffusivity turbulence model” (NLED) cannot be applied to calculate the case of a streamwise rotating channel flow. Thus, the following modeled equation 19 employed in the linear EDM and the quadratic NLED cannot clearly express a counter gradient turbulent diffusion:

$$-\bar{vw} = \nu_t \frac{\partial \bar{W}}{\partial y} \quad (19)$$

where the Reynolds shear stresses of a quadratic NLED are expressed identical to a linear model. Also, noted that the rotational term does not appear in the momentum equation of a fully developed streamwise rotating channel flow indicated as equation (22), in which the rotational effect is included implicitly in the Reynolds shear stress in the same manner as the spanwise rotational flow [38]. Therefore, a cubic NLED or Reynolds stress equation model (RSM) should be used for the calculation of streamwise rotating channel flows as follows:

$$0 = -\frac{1}{\rho} \frac{\partial \bar{P}}{\partial x} + \frac{\partial}{\partial y} \left(\nu \frac{\partial \bar{U}}{\partial y} - \bar{uv} \right) \quad (20)$$

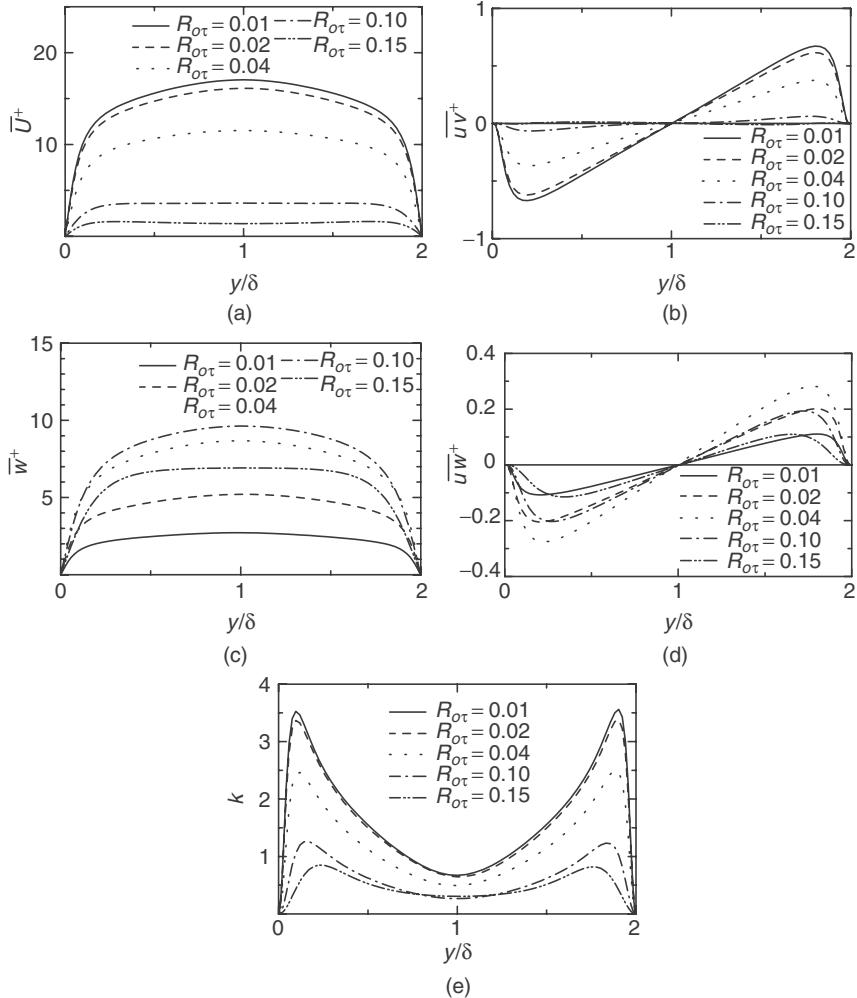


Figure 8.22. DNS results of wall-normal rotating flows (Case 1): (a) streamwise mean velocities, (b) Reynolds shear stresses, $\bar{u}\bar{v}$, (c) spanwise mean velocities, (d) Reynolds shear stresses, $\bar{v}\bar{w}$, and (e) turbulent kinetic energy.

$$0 = -\frac{1}{\rho} \frac{\partial \bar{P}}{\partial y} - \frac{\partial \bar{v}^2}{\partial y} + 2\Omega \bar{W} \quad (21)$$

$$0 = \frac{\partial}{\partial y} \left(\nu \frac{\partial \bar{W}}{\partial y} - \bar{v}\bar{w} \right) \quad (22)$$

As previously stated, one of the main objectives in conducting DNS is to obtain the detailed data for the evaluation of the turbulence models. Thus, this section

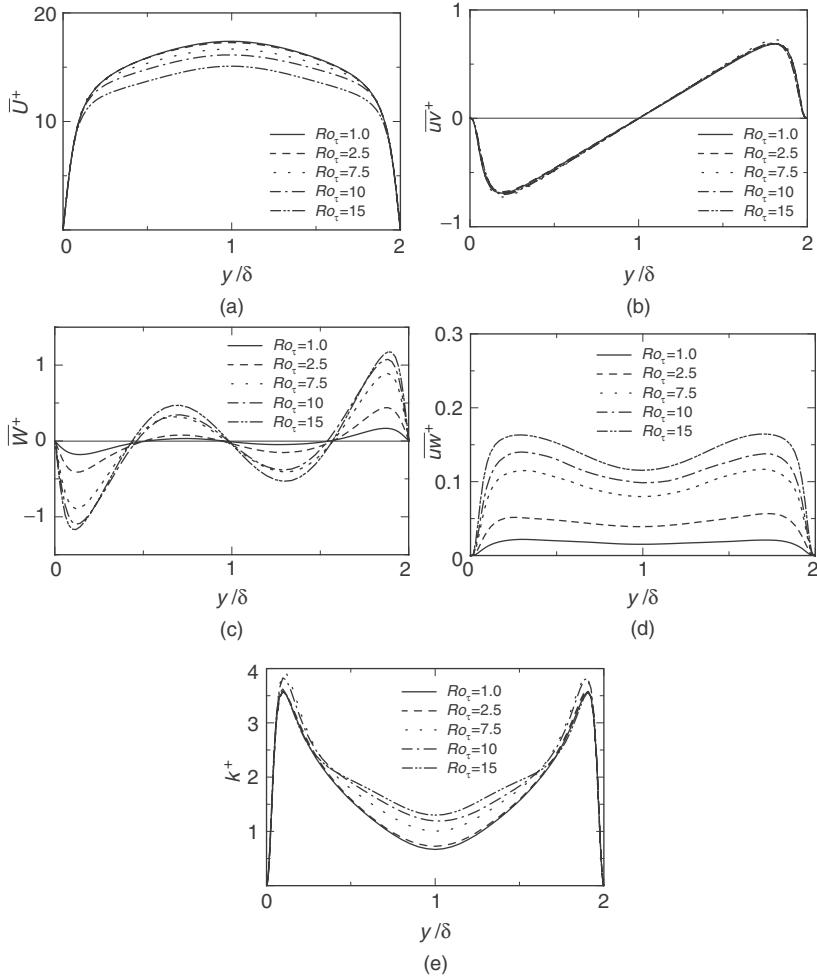


Figure 8.23. DNS results of streamwise rotating flows (Case 2): (a) streamwise mean velocities, (b) Reynolds shear stresses, \bar{uv} , (c) spanwise mean velocities, (d) Reynolds shear stresses, \bar{vw} , and (e) turbulent kinetic energy.

briefly explains the characteristics of channel flows with heat transfer, and with wall-normal or streamwise rotation [39]. DNSs of fully developed nonisothermal channel flows with wall-normal (Case 1) or streamwise (Case 2) rotation are carried out for the 10 cases indicated in Table 8.5. DNS results are shown in Figures 8.24–8.27. In these cases, the spanwise mean velocity appears to be caused by a rotational effect, which increases with the increase in rotation number in both cases. Also, Reynolds shear stress, \bar{vw} , and the spanwise turbulent heat flux, $\bar{w\theta}$, are yielded in both velocity and thermal fields as indicated in Figures 8.25 and 8.27 (note that

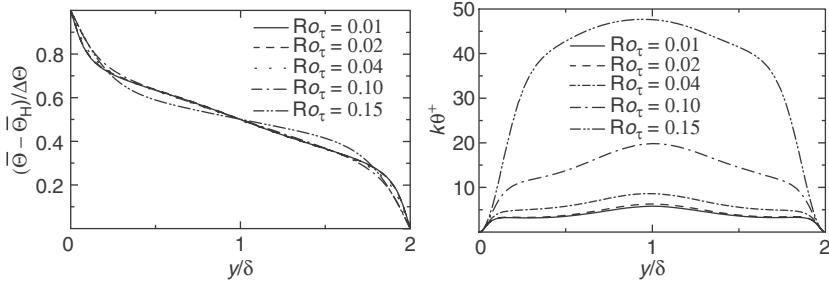


Figure 8.24. Distributions of mean temperature $\bar{\Theta}$ and temperature fluctuation k_θ (Case 1).

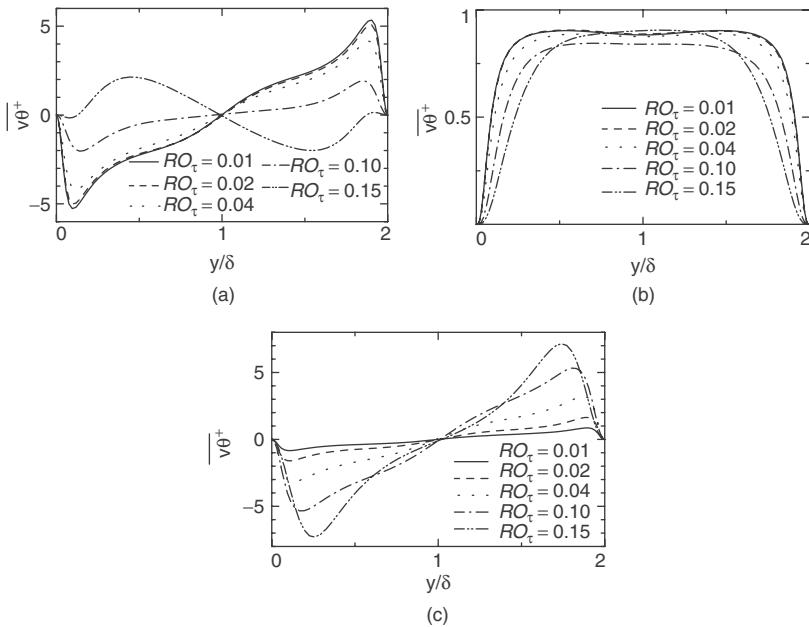


Figure 8.25. Turbulent heat fluxes (Case 1): (a) streamwise, (b) wall-normal, (c) spanwise.

turbulent quantities of velocity field are omitted here; for details, see Ref. [18]). It is well known that Reynolds shear stress, \bar{vw} , remarkably relates to the distribution of spanwise mean velocity, but the spanwise turbulent heat flux, $\bar{w\theta}$, hardly affects the distribution of mean temperature due to the absence of mean temperature gradient in the spanwise direction.

In Case 1, it can be seen that the profiles of mean temperature scarcely vary, but the temperature fluctuations increase distinctly with the increase in rotation number. As for the turbulent heat fluxes, decrease of the streamwise turbulent heat

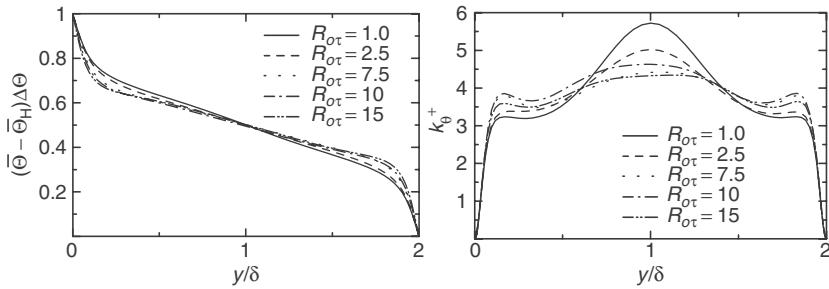


Figure 8.26. Distributions of mean temperature $\bar{\Theta}$ and temperature fluctuation k_θ (Case 2).

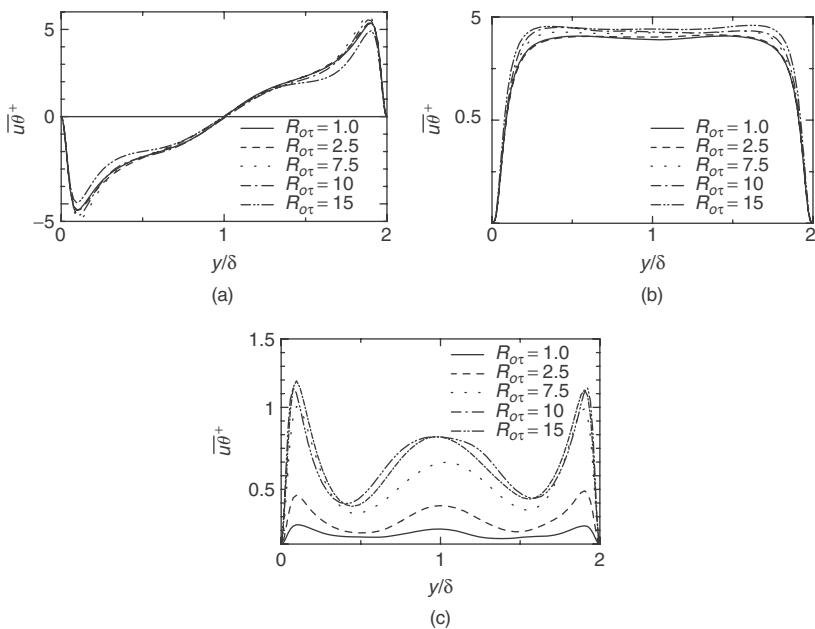


Figure 8.27. Turbulent heat fluxes (Case 2): (a) streamwise, (b) wall-normal, (c) spanwise.

flux, and increase in the spanwise turbulent heat flux with large rotation number are observed as shown in Figure 8.25. The streamwise turbulent heat flux ultimately indicates an inverse sign at the largest rotation number. On the other hand, the wall-normal turbulent heat flux, which affects mean temperature distribution is kept in the region of the channel center, while the decrease is found near the wall with the increase in rotation number.

As for Case 2, since the streamwise mean velocity decreases little and the spanwise mean velocity fluctuates significantly and increases with the increase in

rotation number [18], remarkable increases in the spanwise turbulent heat flux, $\bar{w}\theta$, are also observed near the wall and center of the channel. However, the mean temperature and the other turbulent heat fluxes of this case are hardly affected by the rotation as indicated in Figures 8.26 and 8.27. Note that the counter-gradient diffusion phenomenon was found between the spanwise mean velocity and Reynolds shear stress, \bar{vw} in Case 2 [18]. Therefore, the linear eddy diffusivity model cannot accurately predict in this case [18].

In view of turbulence modeling, this fact clearly demonstrates that the linear eddy diffusivity models cannot be applied to calculate the thermal field of rotating channel flow, because it is difficult to include the rotational effect in the linear eddy diffusivity model. On the other hand, it is well known that it is difficult for a linear eddy diffusivity model to predict the streamwise turbulent heat flux. Although the spanwise heat flux also appears in these cases, the model might not predict it by reason of the modeled hypothesis, i.e., gradient diffusion modeling. Therefore, in order to accurately predict the heat transfer of rotating channel flows, the turbulent heat-flux equation model, the algebraic turbulent heat-flux model, or the nonlinear eddy diffusivity model should be used. In this study, the nonlinear eddy diffusivity for heat model (NLEDHM) [16] is developed to predict adequately these heat transfer phenomena of rotating channel flows.

8.7 Nonlinear Eddy Diffusivity Model for Wall-Bounded Turbulent Flow

8.7.1 Evaluations of existing turbulence models in rotating wall-bounded flows

The evaluated models are a cubic model by Craft et al. [40] (hereinafter referred to as NLCLS) and a quadratic model by Nagano and Hattori [38] (NLHN).

Evaluations of wall-normal rotating flows predicted by the NLCLS and the NLHN models are indicated in Figure 8.28. In this case, it can be seen that the NLHN accurately reproduces turbulent quantities. Thus, one may consider that streamwise rotating flows can be predicted using a quadratic nonlinear model. However, the wall-limiting behavior of the Reynolds stress component in the wall-normal direction is not predicted in the vicinity of the wall shown in Figure 8.28f.

Figure 8.29 shows the assessment result using the present DNS of two-equation models with NLEDHM in the case of streamwise rotating channel flow. Obviously, the quadratic model cannot reproduce this flow as mentioned in the previous section. The cubic model indicates overprediction of mean streamwise velocity, \bar{U} , and underpredicts the Reynolds shear stress, \bar{vw} , in the case of a higher rotation number. The wall-limiting behavior of Reynolds stress components is shown in Figure 8.29f. Although the NLHN model is modeled to satisfy the wall-limiting behaviour of the Reynolds stress component in the wall-normal direction in spanwise rotational flow, it can be seen that the model does not reproduce the wall-limiting behavior.

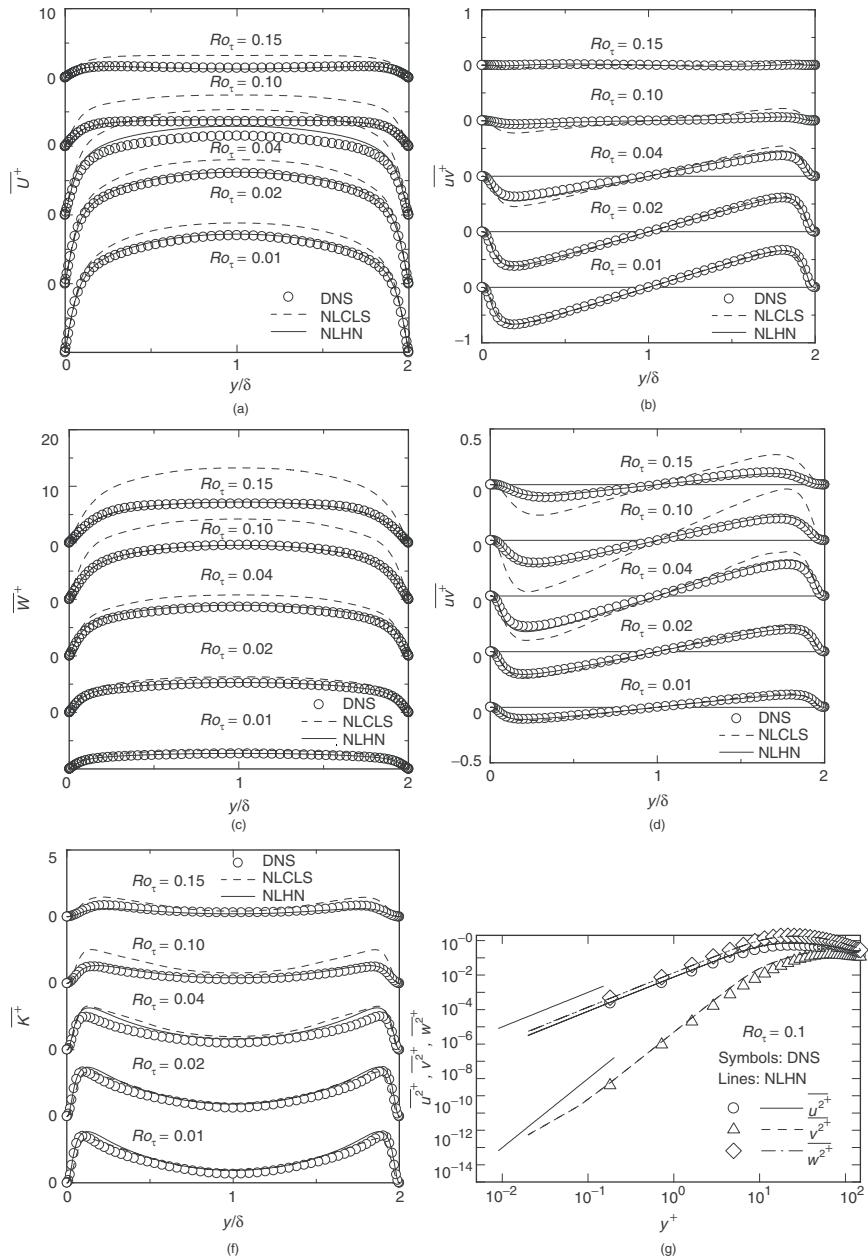


Figure 8.28. Evaluations of predicted wall-normal rotating flows (Case 1):
(a) streamwise mean velocities, (b) Reynolds shear stresses, \overline{uv} , (c) spanwise mean velocities, (d) Reynolds shear stresses, \overline{vw} , (e) turbulent kinetic energy, and (f) wall-limiting behavior of Reynolds stress components.

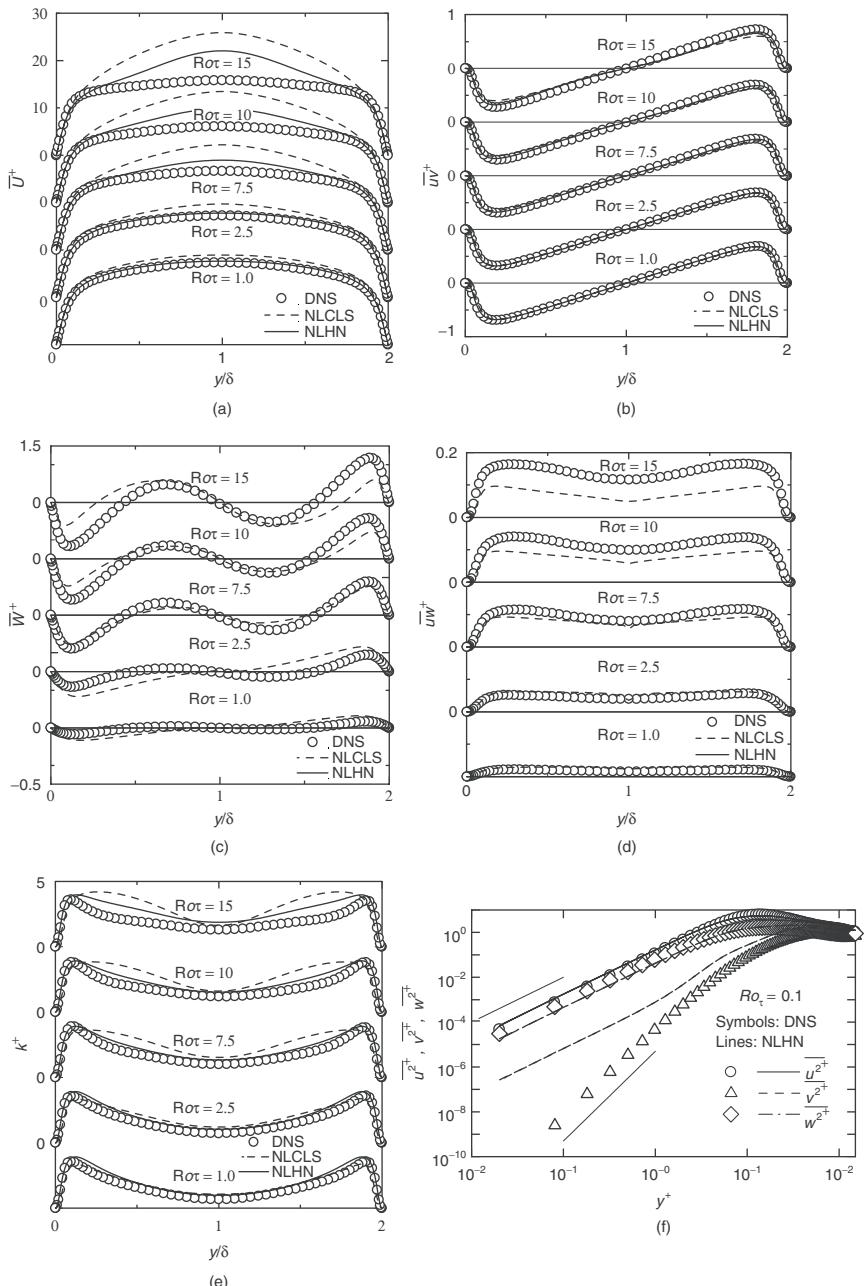


Figure 8.29. Evaluations of predicted streamwise rotating flows (Case 2): (a) streamwise mean velocities, (b) Reynolds shear stresses, $\bar{u}\bar{v}$, (c) spanwise mean velocities, (d) Reynolds shear stresses, $\bar{v}\bar{w}$, (e) turbulent kinetic energy, and (f) wall-limiting behavior of Reynolds stress components.

8.7.2 Proposal of nonlinear eddy diffusivity model for wall-bounded flow

From these results, we propose a cubic NLEDM in a two-equation turbulence model which can adequately predict rotational channel flows with arbitrary rotating axes, in which a modeling of wall-limiting behavior of Reynolds stress components is also considered.

The transport equation of Reynolds stress with the Coriolis term is given as follows:

$$\frac{D\bar{u}_i\bar{u}_j}{Dt} = D_{ij} + T_{ij} + P_{ij} + C_{ij} + \Phi_{ij} - \varepsilon_{ij} \quad (23)$$

where D_{ij} is a molecular diffusion term, T_{ij} is a turbulent and pressure diffusion term, $P_{ij} = -\bar{u}_i\bar{u}_k(\partial\bar{U}_j/\partial x_k) - \bar{u}_j\bar{u}_k(\partial\bar{U}_i/\partial x_k)$ is a production term, $C_{ij} = -2\Omega_m(\epsilon_{mkj}\bar{u}_i\bar{u}_k + \epsilon_{mki}\bar{u}_j\bar{u}_k)$ is a Coriolis term, Φ_{ij} is a pressure-stain correlation term and ε_{ij} is a dissipation term, respectively.

Introducing the Reynolds stress anisotropy tensor $b_{ij} = \bar{u}_i\bar{u}_j/2k - \delta_{ij}/3$ and neglecting the diffusive effect, the following relation is derived from equations (9) and (23):

$$\frac{Db_{ij}}{Dt} = \frac{1}{2k}(P_{ij} + C_{ij} + \Phi_{ij} - \varepsilon_{ij}) - \frac{b_{ij} + \delta_{ij}/3}{k}(P_k - \varepsilon) \quad (24)$$

In the local equilibrium state, since the relation $Db_{ij}/Dt = 0$ holds, equation (24) yields the following relation:

$$(P_{ij} + C_{ij} + \Phi_{ij} - \varepsilon_{ij}) = 2\left(b_{ij} + \frac{\delta_{ij}}{3}\right)(P_k - \varepsilon) \quad (25)$$

Using the form $\varepsilon_{ij} = \frac{2}{3}\varepsilon\delta_{ij} + {}_D\varepsilon_{ij}$ of the dissipation term [41] for equation (25), we can obtain:

$$\begin{aligned} (P_k - \varepsilon)b_{ij} &= -\frac{2}{3}kS_{ij} - k\left(b_{ik}S_{jk} + b_{jk}S_{ik} - \frac{2}{3}b_{mn}S_{mn}\delta_{ij}\right) \\ &\quad - k[b_{ik}(W_{jk} + 2\varepsilon_{mkj}\Omega_m) + b_{jk}(W_{ik} + 2\varepsilon_{mki}\Omega_m)] + \frac{1}{2}\Pi_{ij} \end{aligned} \quad (26)$$

where $\Pi_{ij} = \Phi_{ij} - {}_D\varepsilon_{ij}$, and the modeled Π_{ij} is employed as the following general linear model:

$$\begin{aligned} \Pi_{ij} &= -C_1\varepsilon b_{ij} + C_2kS_{ij} + C_3k\left(b_{ik}S_{jk} + b_{jk}S_{ik} - \frac{2}{3}b_{mn}S_{mn}\delta_{ij}\right) \\ &\quad + C_4k(b_{ik}W_{jk} + b_{jk}W_{ik}) \end{aligned} \quad (27)$$

where $C_1 \sim C_5$ are model constants.

Substituting equation (27) into (26) and introducing nondimensional quantities, we can obtain the following relation:

$$b_{ij}^* = -S_{ij}^* - \left(b_{ik}^* S_{jk}^* + b_{jk}^* S_{ik}^* - \frac{2}{3} b_{k\ell}^* S_{k\ell}^* \delta_{ij} \right) + b_{ik}^* W_{kj}^* + b_{jk}^* W_{ki}^* \quad (28)$$

where

$$\left. \begin{aligned} S_{ij}^* &= \frac{1}{2} g \tau (2 - C_3) S_{ij}, & W_{ij}^* &= \frac{1}{2} g \tau (2 - C_4) \left[\Omega_{ij} + \left(\frac{C_4 - 4}{C_4 - 2} \right) \varepsilon_{mji} \Omega_m \right] \\ b_{ij}^* &= \left(\frac{C_3 - 2}{C_2 - \frac{4}{3}} \right) b_{ij}, & \tau &= \frac{k}{\varepsilon}, & g &= \left(\frac{1}{2} C_1 + \frac{P_k}{\varepsilon} + \frac{G_k}{\varepsilon} - 1 \right)^{-1} \end{aligned} \right\} \quad (29)$$

Equation (28) can be written in the matrix form as:

$$\mathbf{b}^* = -\mathbf{S}^* - \left(\mathbf{b}^* \mathbf{S}^* + \mathbf{S}^* \mathbf{b}^* - \frac{2}{3} \{ \mathbf{b}^* \mathbf{S}^* \} \mathbf{I} \right) + \mathbf{b}^* \mathbf{W}^* - \mathbf{W}^* \mathbf{b}^* \quad (30)$$

In order to derive an explicit form of \mathbf{b}^* from equation (30), the integrity basis, $\mathbf{b}^* = \sum_{\lambda} Q^{(\lambda)} \mathbf{T}^{(\lambda)}$ first proposed by Pope [42], is used with the following 10 basis tensors:

$$\left. \begin{aligned} \mathbf{T}^{(1)} &= \mathbf{S}^*, & \mathbf{T}^{(6)} &= \mathbf{W}^{*2} \mathbf{S}^* + \mathbf{S}^* \mathbf{W}^{*2} - \frac{2}{3} \{ \mathbf{S}^* \mathbf{W}^{*2} \} \mathbf{I} \\ \mathbf{T}^{(2)} &= \mathbf{S}^* \mathbf{W}^* - \mathbf{W}^* \mathbf{S}^*, & \mathbf{T}^{(7)} &= \mathbf{W}^* \mathbf{S}^* \mathbf{W}^{*2} - \mathbf{W}^{*2} \mathbf{S}^* \mathbf{W}^* \\ \mathbf{T}^{(3)} &= \mathbf{S}^{*2} - \frac{1}{3} \{ \mathbf{S}^{*2} \} \mathbf{I}, & \mathbf{T}^{(8)} &= \mathbf{S}^* \mathbf{W}^* \mathbf{S}^{*2} - \mathbf{S}^{*2} \mathbf{W}^* \mathbf{S}^* \\ \mathbf{T}^{(4)} &= \mathbf{W}^{*2} - \frac{1}{3} \{ \mathbf{W}^{*2} \} \mathbf{I}, & \mathbf{T}^{(9)} &= \mathbf{W}^{*2} \mathbf{S}^{*2} + \mathbf{S}^{*2} \mathbf{W}^{*2} - \frac{2}{3} \{ \mathbf{S}^{*2} \mathbf{W}^{*2} \} \mathbf{I} \\ \mathbf{T}^{(5)} &= \mathbf{W}^* \mathbf{S}^{*2} - \mathbf{S}^{*2} \mathbf{W}^*, & \mathbf{T}^{(10)} &= \mathbf{W}^* \mathbf{S}^{*2} \mathbf{W}^{*2} - \mathbf{W}^{*2} \mathbf{S}^{*2} \mathbf{W}^* \end{aligned} \right\} \quad (31)$$

Substituting equation (31) into $\mathbf{b}^* = \sum_{\lambda} Q^{(\lambda)} \mathbf{T}^{(\lambda)}$ gives:

$$\begin{aligned} \mathbf{b}^* &= Q^{(1)} \mathbf{S}^* + Q^{(2)} (\mathbf{S}^* \mathbf{W}^* - \mathbf{W}^* \mathbf{S}^*) + Q^{(3)} \left(\mathbf{S}^{*2} - \frac{1}{3} \{ \mathbf{S}^{*2} \} \mathbf{I} \right) \\ &\quad + Q^{(4)} \left(\mathbf{W}^{*2} - \frac{1}{3} \{ \mathbf{W}^{*2} \} \mathbf{I} \right) + Q^{(5)} (\mathbf{W}^* \mathbf{S}^{*2} - \mathbf{S}^{*2} \mathbf{W}^*) \\ &\quad + Q^{(6)} \left(\mathbf{W}^{*2} \mathbf{S}^* + \mathbf{S}^* \mathbf{W}^{*2} - \frac{2}{3} \{ \mathbf{S}^* \mathbf{W}^{*2} \} \mathbf{I} \right) \\ &\quad + Q^{(7)} (\mathbf{W}^* \mathbf{S}^* \mathbf{W}^{*2} - \mathbf{W}^{*2} \mathbf{S}^* \mathbf{W}^*) + Q^{(8)} (\mathbf{S}^* \mathbf{W}^* \mathbf{S}^{*2} - \mathbf{S}^{*2} \mathbf{W}^* \mathbf{S}^*) \\ &\quad + Q^{(9)} \left(\mathbf{W}^{*2} \mathbf{S}^{*2} + \mathbf{S}^{*2} \mathbf{W}^{*2} - \frac{2}{3} \{ \mathbf{S}^{*2} \mathbf{W}^{*2} \} \mathbf{I} \right) \\ &\quad + Q^{(10)} (\mathbf{W}^* \mathbf{S}^{*2} \mathbf{W}^{*2} - \mathbf{W}^{*2} \mathbf{S}^{*2} \mathbf{W}^*) \end{aligned} \quad (32)$$

On the other hand, substituting the integrity basis and the theoretical expressions [42]:

$$\mathbf{T}^{(\lambda)} \mathbf{S}^* + \mathbf{S}^* \mathbf{T}^{(\lambda)} - \frac{2}{3} \{ \mathbf{T}^{(\lambda)} \mathbf{S}^* \} \mathbf{I} = \sum_{\gamma} H_{\lambda\gamma} \mathbf{T}^{(\gamma)}$$

and

$$\mathbf{T}^{(\lambda)} \mathbf{W}^* - \mathbf{W}^* \mathbf{T}^{(\lambda)} = \sum_{\gamma} J_{\lambda\gamma} \mathbf{T}^{(\gamma)}$$

where H and J are the scalar functions, into equation 30, gives the following equation for $\mathbf{T}^{(\lambda)}$:

$$\sum_{\lambda} Q^{(\lambda)} \mathbf{T}^{(\lambda)} = - \sum_{\lambda} \delta_{1\lambda} \mathbf{T}^{(1)} - \sum_{\lambda} Q^{(\lambda)} \left[\left(\sum_{\lambda} H_{\lambda\gamma} \mathbf{T}^{(\lambda)} \right) - \left(\sum_{\lambda} J_{\lambda\gamma} \mathbf{T}^{(\lambda)} \right) \right] \quad (33)$$

Equation (33) can be written as $Q^{(\lambda)} = A_{\gamma\lambda}^{-1} B_{\lambda}$, where $A_{\gamma\lambda} = -\delta_{\lambda\gamma} - H_{\lambda\gamma} + J_{\lambda\gamma}$, and $B_{\lambda} = \delta_{1\lambda}$. By using Cayley–Hamilton identities, the matrices $H_{\lambda\gamma}$ and $J_{\lambda\gamma}$ can be determined. Thus, we can obtain $Q^{(\lambda)}$ using Mathematica as follows:

$$\left. \begin{aligned} Q^{(1)} &= -\frac{1}{2}(6 - 3\eta_1 - 21\eta_2 - 2\eta_3 + 30\eta_4)/D & Q^{(6)} &= -9/D \\ Q^{(2)} &= -(3 + 3\eta_1 - 6\eta_2 + 2\eta_3 + 6\eta_4)/D & Q^{(7)} &= 9/D \\ Q^{(3)} &= (6 - 3\eta_1 - 12\eta_2 - 2\eta_3 - 6\eta_4)/D & Q^{(8)} &= 9/D \\ Q^{(4)} &= -3(3\eta_1 + 2\eta_3 + 6\eta_4)/D & Q^{(9)} &= 18D \\ Q^{(5)} &= -9/D & Q^{(10)} &= 0 \end{aligned} \right\} \quad (34)$$

where

$$D = 3 - \frac{7}{2} + \eta_1^2 - \frac{15}{2}\eta_2 - 8\eta_1\eta_2 + 3\eta_2^2 - \eta_3 + \frac{2}{3}\eta_1\eta_3 - 2\eta_2\eta_3 + 21\eta_4 + 24\eta_5 + 2\eta_1\eta_4 - 6\eta_2\eta_4 \quad (35)$$

with $\eta_1 = \{\mathbf{S}^{*2}\}$, $\eta_2 = \{\mathbf{W}^{*2}\}$, $\eta_3 = \{\mathbf{S}^{*3}\}$, $\eta_4 = \{\mathbf{S}^* \mathbf{W}^{*2}\}$, and $\eta_5 = \{\mathbf{S}^{*2} \mathbf{W}^{*2}\}$.

In order to obtain reasonable forms for $Q^{(1)} \sim Q^{(9)}$, we assume $S_{13} = W_{13} = 0$. Thus, η_3 and η_4 become 0 and $\eta_5 = \frac{1}{2}\eta_1\eta_2$. Therefore, the following reasonable forms for $Q^{(1)} \sim Q^{(6)}$ can be derived:

$$\left. \begin{aligned} Q^{(1)} &= -\frac{3}{3 - 2\eta_1 - 6\eta_2}, & Q^{(2)} &= -\frac{3}{3 - 2\eta_1 - 6\eta_2}, & Q^{(3)} &= \frac{6}{3 - 2\eta_1 - 6\eta_2} \\ Q^{(4)} &= \frac{-18\eta_1}{(3 - 2\eta_1 - 6\eta_2)(2 - \eta_1 - \eta_2)}, & Q^{(5)} &= \frac{-18}{(3 - 2\eta_1 - 6\eta_2)(2 - \eta_1 - \eta_2)} \\ Q^{(6)} &= \frac{-18}{(3 - 2\eta_1 - 6\eta_2)(2 - \eta_1 - \eta_2)} \end{aligned} \right\} \quad (36)$$

Consequently, we can obtain the cubic NLEDM as follows:

$$\begin{aligned} b_{ij}^* = & -\frac{3}{3-2\eta^2+6\zeta^2} \left\{ S_{ij}^* + (S_{ik}^* W_{kj}^* - W_{ik}^* S_{kj}^*) - 2 \left(S_{ik}^* S_{kj}^* - \frac{1}{3} S_{mn}^* S_{nm}^* \delta_{ij} \right) \right. \\ & + \frac{6}{2-\eta^2+\zeta^2} \left[\eta^2 (W_{ik}^* W_{kj}^* - \frac{1}{3} \delta_{ij} W_{nn}^* W_{nn}^*) + (W_{ik}^* S_{k\ell}^* S_{\ell j}^* - S_{ik}^* S_{k\ell}^* W_{\ell j}^*) \right. \\ & \left. \left. + \left(W_{ik}^* W_{k\ell}^* S_{\ell j}^* + S_{ik}^* W_{k\ell}^* W_{\ell j}^* - \frac{2}{3} S_{\ell m}^* W_{mn}^* W_{n\ell}^* \delta_{ij} \right) \right] \right\} \end{aligned} \quad (37)$$

where $\eta = (S_{ij}^* S_{ij}^*)^{\frac{1}{2}}$, $\zeta = (W_{ij}^* W_{ij}^*)^{\frac{1}{2}}$, and b_{ij}^* , S_{ij}^* , and W_{ij}^* are nondimensional quantities, respectively, redefined as follows:

$$b_{ij}^* = C_D b_{ij}, \quad S_{ij}^* = C_D \tau_{R_o} S_{ij}, \quad W_{ij}^* = 2C_D \tau_{R_o} W_{ij} \quad (38)$$

Since the proposed model in equation (37) can be written as the following equation for the Reynolds shear stress, \bar{w} (or b_{23}), in streamwise rotating flow, in which $S_{12}(=S_{21})$, $W_{12}(=-W_{21})$, $S_{23}(=S_{32})$, and $W_{23}(=-W_{32})$ exist, the Reynolds shear stress, \bar{w} , can be reproduced by the present model:

$$\begin{aligned} b_{23}^* = & \frac{3}{3-2\eta^2+6\zeta^2} \left\{ S_{23}^* + \frac{6}{2-\eta^2+\zeta^2} [(W_{21}^* S_{12}^* S_{23}^* - S_{21}^* S_{12}^* W_{23}^*) \right. \\ & \left. + (W_{21}^* W_{12}^* S_{23}^* + W_{23}^* W_{32}^* S_{23}^* + S_{21}^* W_{12}^* W_{23}^* + S_{23}^* W_{32}^* W_{23}^*)] \right\} \end{aligned} \quad (39)$$

In equation (37), the functions $3/(3-2\eta^2+6\zeta^2)$ and $6/(2-\eta^2+\zeta^2)$ may be taken to be a negative value or 0 with the increase in η . Thus, in order to avoid taking a negative value or 0, we model these functions taking into account the rotational effect as follows:

$$\frac{3}{3-2\eta^2+6\zeta^2} \simeq \frac{1}{\left[1 + \frac{22}{3} \left(\frac{W^{*2}}{4} \right) + \frac{2}{3} \left(\frac{W^{*2}}{4} - S^{*2} - f_\omega^{*2} \right) f_B + \frac{2}{3} f_\omega^{*2} \right]} \quad (40)$$

$$\frac{6}{2-\eta^2+\zeta^2} \simeq \frac{3}{\left[1 + \frac{3}{2} \left(\frac{W^{*2}}{4} \right) + \frac{1}{2} \left(\frac{W^{*2}}{4} - S^{*2} - f_\omega^{*2} \right) f_B + \frac{1}{2} f_\omega^{*2} \right]} \quad (41)$$

where

$$f_B = 1 + C_\eta \left(\frac{W^{*2}}{4} - S^{*2} - f_\omega^{*2} \right) \quad (42)$$

$$f_\omega^{*2} = (C_D \tau_{R_o})^2 [\Omega_m (2\varepsilon_{mji} W_{ij} - \Omega_m)]^2 \quad (43)$$

The model function, f_ω^{*2} , in equation (43) is introduced to avoid an inappropriate value of $(W^{*2}/4 - S^{*2})$ with the increase in a rotation number. Since $f_\omega^{*2} = 0$ is consistently kept in non rotational flows, inadequate Reynolds stresses predicted by the proposed model are not given in non rotational flows.

Finally, in order to construct a precise model for rotational flow, a cubic term introduced in an NLCLS model [40] is added in the proposed model. Consequently, the final form of the present cubic NLEDM is given as follows:

$$\begin{aligned}\overline{u_i u_j} = & \frac{2}{3} k \delta_{ij} - c_1 v_t S_{ij} + c_2 k (\tau_{R_o}^2 + \tau_{R_w}^2) (S_{ik} W_{kj} - W_{ik} S_{kj}) \\ & + c_3 k (\tau_{R_o}^2 + \tau_{R_w}^2) \left(S_{ik} S_{kj} - \frac{1}{3} S_{mn} S_{mn} \delta_{ij} \right) \\ & + c_4 k \tau_{Ro}^2 \left(W_{ik} W_{kj} - \frac{1}{3} W_{mn} W_{mn} \delta_{ij} \right) \\ & + c_5 k \tau_{Ro}^3 (W_{ik} S_{k\ell} S_{\ell j} - S_{ik} S_{k\ell} W_{\ell j}) + c_6 k \tau_{Ro}^3 \left(W_{ik} W_{k\ell} S_{\ell j} + S_{ik} W_{k\ell} W_{\ell j} \right. \\ & \left. - \frac{2}{3} S_{\ell m} W_{mn} W_{n\ell} \delta_{ij} \right) + c_7 k \tau_{Ro}^3 (S_{ij} S_{k\ell} S_{k\ell} - S_{ij} W_{k\ell} W_{k\ell})\end{aligned}\quad (44)$$

where $v_t = C_\mu f_\mu k^2 / \varepsilon$ and $C_\mu = 0.12$, and the model functions and constants are given as follows:

$$\left. \begin{aligned}c_1 &= -2/f_{R1}, \quad c_2 = -4C_D/f_{R1}, \quad c_3 = 4C_D/f_{R1} \\c_4 &= -8C_D^3 S^2/(f_{R1} f_{R2}), \quad c_5 = 4C_D^2 f_{R\Omega}/(f_{R1} f_{R2}), \\c_6 &= 2C_D^2/(f_{R1} f_{R2}), \quad c_7 = 10C_\mu^2 C_D^2, \quad C_D = 0.8 \\f_{R1} &= 1 + (C_D \tau_{Ro})^2 [(22/3)W^2 + (2/3)(W^2 - S^2 - f_\omega^2)f_B] \\f_{R2} &= 1 + (C_D \tau_{Ro})^2 [(3/2)W^2 + (1/2)(W^2 - S^2 - f_\omega^2)f_B] \\f_{R\Omega} &= 12 [1 - \exp(f_{SW}^{3/4}/26)] \\f_\mu &= [1 - f_w(32)] \{1 + (40/R_t^{3/4}) \exp[-(R_{tm}/35)^{3/4}]\} \\f_B &= 1 + C_\eta (C_D \tau_{Ro})^2 (W^2 - S^2 - f_\omega^2) \\f_\omega^2 &= \Omega_m (2\varepsilon_{mji} W_{ij} - \Omega_m), \quad C_\eta = 5.0\end{aligned}\right\} \quad (45)$$

The characteristic time-scale τ_{R_w} is introduced in an NLHN model [38] for an anisotropy and wall-limiting behavior of turbulent intensity defined as:

$$\tau_{R_w} = \sqrt{\frac{1}{6} \frac{f_{R1}/C_D}{f_{SW}}} \left(1 - \frac{3C_{v1}f_{v2}}{8} \right) f_{v1}^2 \quad (46)$$

Table 8.6. Model constants and functions of transport equations for k and ε

$C_{\varepsilon 1}$	$C_{\varepsilon 2}$	$C_{\varepsilon 3}$	$C_{\varepsilon 4}$	$C_{\varepsilon 5}$	C_s	C_ε	C_Ω	$C_{f\Omega}$
1.45	1.9	0.02	0.5	0.015	1.4	1.4	-0.045	6.0

f_{t1}	f_{t2}	f_ε
$\frac{1 + 9f_w(8)}{[1 - f_w(32)]^{1/2}}$	$\frac{1 + 5f_w(8)}{[1 - f_w(32)]^{1/2}}$	$\left\{ 1 - 0.3 \exp \left[- \left(\frac{R_t}{6.5} \right)^2 \right] \right\} [1 - f_w^2(3.7)]$

f_Ω	R_Ω
$C_{f\Omega} \exp \left[- \left(\frac{R_\Omega}{10} \right)^{0.2} \right]$	$\sqrt{\frac{\nu}{\varepsilon}} \sqrt{f_{SW}^\Omega}$

$$f_{SW} = \frac{W^2}{2} + \frac{S^2}{3} - f_{SW}^\Omega \quad (47)$$

$$f_{SW}^\Omega = \left[\left(\sqrt{\frac{S^2}{2}} - \sqrt{\frac{W^2}{2}} \right) f_w(1) \right]^2 \quad (48)$$

where $C_{v1} = 0.4$, $C_{v2} = 2.0 \times 10^3$, $f_{v1} = \exp[-(R_{tm}/45)^2]$ and $f_{v2} = 1 - \exp(-\sqrt{R_t/C_{v2}})$. The wall reflection function is defined by $f_w(\xi) = \exp[-(R_{tm}/\xi)^2]$, and the corrected Reynolds number for the rotating flow is given as $R_{tm} = (C_{tm} n^* R_t^{1/4}) / (C_{tm} R_t^{1/4} + n^*)$ with $C_{tm} = 1.3 \times 10^2$ [38]. Moreover, it is found from the evaluation shown in Figures 8.28f and 8.29f that the wall-limiting behavior of Reynolds stress component is not satisfied for the tested rotating flows. Thus, the model function in equation (48) of characteristic time-scale τ_{R_w} is slightly modified as follows:

$$f_{SW}^\Omega = \left| \sqrt{\frac{S^*{}^2}{2}} - \sqrt{\frac{W^*{}^2}{2}} \right| f_w(1)^2 \quad (49)$$

where $S^* = \varepsilon_{\ell m n} S_{m n} \Omega_\ell$, $W^* = \varepsilon_{\ell m n} W_{m n} \Omega_\ell$.

For the modeled transport equations of k and ε as indicated in equations (9) and (10), the turbulent diffusion and the pressure diffusion terms are modeled as follows [38]:

$$T_k = \frac{\partial}{\partial x_j} \left(C_s f_{t1} \frac{\nu_t}{k} \bar{u}_j \bar{u}_\ell \frac{\partial k}{\partial x_\ell} \right) \quad (50)$$

$$T_\varepsilon = \frac{\partial}{\partial x_j} \left(C_\varepsilon f_{t2} \frac{\nu_t}{k} \bar{u}_j \bar{u}_\ell \frac{\partial \varepsilon}{\partial x_\ell} \right) \quad (51)$$

$$\Pi_k = \max \left\{ -0.5\nu \frac{\partial}{\partial x_j} \left[\frac{k}{\varepsilon} \frac{\partial \varepsilon}{\partial x_j} f_w(1) \right], 0 \right\} \quad (52)$$

$$\Pi_\varepsilon = C_{\varepsilon 4} \frac{\partial}{\partial x_j} \left\{ [1 - f_w(5)] \frac{\varepsilon}{k} \frac{\partial \varepsilon}{\partial x_j} f_w(5) \right\} \quad (53)$$

The extra term in the ε -equation is adopted as the identical to the NLHN model [38]:

$$E = C_{\varepsilon 3} \nu \frac{k}{\varepsilon} \overline{u_j u_\ell} \frac{\partial^2 \overline{U}_i}{\partial x_\ell \partial x_k} \frac{\partial^2 \overline{U}_j}{\partial x_\ell \partial x_k} + C_{\varepsilon 5} \nu \frac{k}{\varepsilon} \frac{\partial \overline{u_j u_k}}{\partial x_j} \frac{\partial \overline{U}_i}{\partial x_k} \frac{\partial^2 \overline{U}_i}{\partial x_j \partial x_k} \quad (54)$$

Finally, a rotation-influenced addition term, R , in the ε -equation (10) is generalized as follows:

$$R = C_\Omega f_\Omega k \epsilon_{ij\ell} W_{ij} d_\ell \Omega_\ell \quad (55)$$

where d_ℓ is the unit vector in the spanwise direction. Model constants and functions in the k - and ε -equations are indicated in Table 8.6. In what follows, we call the cubic NLEDMM, thus obtained, a nonlinear eddy diffusivity for momentum model (NLEDMM).

8.8 Nonlinear Eddy Diffusivity Model for Wall-Bounded Turbulent Heat Transfer

8.8.1 Evaluations of turbulent heat transfer models in rotating channel flows

To improve the nonlinear eddy diffusivity for heat model (NLEDHM), the following nonlinear turbulence models are evaluated using DNS data (refer to the detailed model functions and constants of each models in Refs. [38], [43], and [44]).

- Nagano and Hattori [38] (hereinafter referred to as NLHN)

$$\overline{u_j \theta} = -\alpha'_{jk} \frac{\partial \overline{\Theta}}{\partial x_k} + \frac{C_{\theta 1}}{f_{RT}} \tau_{mo}^2 \overline{u_\ell u_k} (C_{\theta 2} S_{j\ell} + C_{\theta 3} W_{j\ell} + 2C_{\theta 1} \varepsilon_{jml} \Omega_m) \frac{\partial \overline{\Theta}}{\partial x_k} \quad (56)$$

where α'_{jk} is an anisotropic eddy diffusivity tensor for heat as follows:

$$\alpha'_{jk} = \left(\frac{C_{\theta 1}^*}{f_{RT}} \right) \overline{u_j u_k} \tau_m \quad (57)$$

- Suga and Abe [43] (hereinafter referred to as NLSA)

$$\overline{u_i \theta} = -C_\theta k \tau (\sigma_{ij} + \alpha_{ij}) \frac{\partial \overline{\Theta}}{\partial x_j} \quad (58)$$

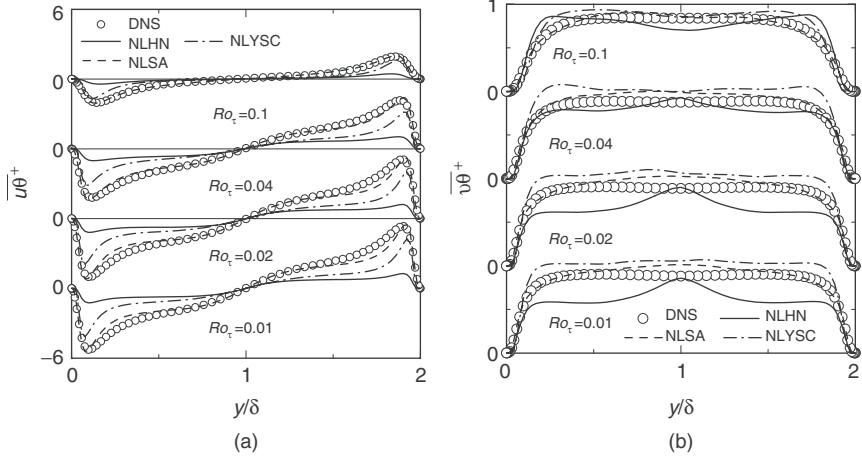


Figure 8.30. Evaluations of turbulent heat fluxes (Case 1); (a) streamwise ($\bar{u}\theta$), (b) wall-normal ($\bar{v}\theta$).

where

$$\begin{aligned} \sigma_{ij} &= C_{\sigma 0} \delta_{ij} + C_{\sigma 1} \frac{\bar{u}_i \bar{u}_j}{k} + C_{\sigma 2} \bar{u}_i \bar{u}_\ell \frac{\bar{u}_\ell \bar{u}_j}{k}, \\ \alpha_{ij} &= C_{\alpha 0} \tau W_{ij} + C_{\alpha 1} \tau \left(W_{i\ell} \frac{\bar{u}_\ell \bar{u}_j}{k} + W_{\ell j} \frac{\bar{u}_i \bar{u}_\ell}{k} \right) \end{aligned}$$

Note that the original model is described in the non rotational form [43]. In order to evaluate the NLSA model in the rotational flow, the vorticity tensor is replaced with the absolute vorticity tensor in the model.

- Younis et al. [44] (hereinafter referred to as NLYSC)

$$\begin{aligned} \bar{u}_i \bar{\theta} &= -C_1 \frac{k^2}{\varepsilon} \frac{\partial \bar{\Theta}}{\partial x_i} - C_2 \frac{k}{\varepsilon} \frac{\bar{u}_i \bar{u}_j}{\varepsilon} \frac{\partial \bar{\Theta}}{\partial x_j} - C_3 \frac{k^3}{\varepsilon^2} \frac{\partial \bar{U}_i}{\partial x_j} \frac{\partial \bar{\Theta}}{\partial x_j} \\ &\quad - C_4 \frac{k^2}{\varepsilon^2} \left(\frac{\bar{u}_i \bar{u}_k}{\varepsilon} \frac{\partial \bar{U}_j}{\partial x_k} - \frac{\bar{u}_j \bar{u}_k}{\varepsilon} \frac{\partial \bar{U}_i}{\partial x_k} \right) \frac{\partial \bar{\Theta}}{\partial x_j} \end{aligned} \quad (59)$$

Note that the above description is in the original model form [44]. In order to apply to the model evaluation in rotating channel flow, the velocity gradient, $\partial \bar{U}_i / \partial x_j$, must be replaced with $S_{ij} + W_{ij}$ in the model. However, the evaluation is conducted using the original form, because we should know the adverse effect of using it.

Evaluation results are shown in Figures 8.30 and 8.31. In Case 1, it can be seen that the NLSA model accurately reproduces the streamwise heat flux at various rotation numbers, because the NLSA model was improved in order to properly predict the streamwise turbulent heat flux [43]. As for an evaluation for a prediction of

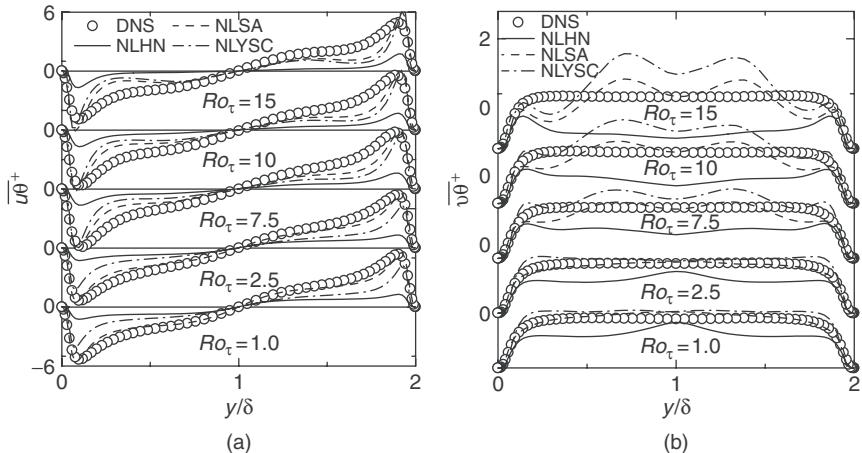


Figure 8.31. Evaluations of turbulent heat fluxes (Case 2); (a) streamwise ($\bar{u}\theta$), (b) wall-normal ($\bar{v}\theta$).

wall-normal heat flux, the NLHN model slightly underpredicts in case of the lowest rotation number, but the NLHN model gives good agreement with DNS data with increase in rotation number. In Case 2, the NLSA and NLYSC give adequate predictions of both turbulent heat fluxes in case of the lowest rotation number. Even though the NLYSC does not employ the rotational form, it is noteworthy that the NLYSC model nearly gives equivalent results in comparison with the NLSA model. However, it is obvious that all models cannot predict DNS results with increase in rotation number. In particular, the wall-normal turbulent heat flux is not reproduced by evaluated models in case of the highest rotation number. From this evaluation of Case 2, it may be concluded that the mean temperature cannot be adequately predicted using evaluated models. Thus, the model should be improved to accurately predict heated channel flows with arbitrary rotating axes and various rotation numbers.

8.8.2 Proposal of nonlinear eddy diffusivity model for wall-bounded turbulent heat transfer

The formulation of nonlinear eddy diffusivity for a heat model (NLEDHM) is derived below [39]. The transport equation for turbulent heat flux is written:

$$\frac{D\bar{u}_j\theta}{Dt} = D_{j\theta} + T_{j\theta} + P_{j\theta} + \Phi_{j\theta} - \varepsilon_{j\theta} + C_{j\theta} \quad (60)$$

where we introduce the nondimensional heat-flux parameter a_j^* [45]:

$$a_j^* = \frac{\bar{u}_j\theta}{\sqrt{k}\sqrt{k_\theta}} \quad (61)$$

where $k_\theta = \overline{\theta^2}/2$.

The transport equation for a_j^* is derived to neglect the diffusive effects as follows:

$$\begin{aligned} \frac{D a_j^*}{Dt} &= \frac{D}{Dt} \left(\frac{\bar{u}_j \theta}{\sqrt{k} \sqrt{k_\theta}} \right) = \frac{1}{\sqrt{k} \sqrt{k_\theta}} \frac{D \bar{u}_j \theta}{Dt} - \frac{1}{2} \frac{\bar{u}_j \theta}{\sqrt{k} \sqrt{k_\theta}} \left(\frac{1}{k} \frac{Dk}{Dt} + \frac{1}{k_\theta} \frac{Dk_\theta}{Dt} \right) \\ &= \frac{1}{\sqrt{k} \sqrt{k_\theta}} (P_{j\theta} + \Phi_{j\theta} - \varepsilon_{j\theta} + C_{j\theta}) \\ &\quad - \frac{1}{2} \frac{\bar{u}_j \theta}{\sqrt{k} \sqrt{k_\theta}} \left[\frac{\varepsilon}{k} \left(\frac{P_k}{\varepsilon} - 1 \right) + \frac{\varepsilon_\theta}{k_\theta} \left(\frac{P_{k_\theta}}{\varepsilon_\theta} - 1 \right) \right] \end{aligned} \quad (62)$$

where the transport equations for k and k_θ [see equations (9) and (11)] are employed.

The most general linear expression is adopted for the modeled pressure-temperature gradient correlation term, $\Phi_{j\theta}$, and dissipation term, $\varepsilon_{j\theta}$:

$$\Phi_{j\theta} - \varepsilon_{j\theta} = -C_1 \frac{\bar{u}_j \theta}{\tau_u} + C_2 \bar{u}_k \theta \frac{\partial \bar{U}_j}{\partial x_k} + C_3 \bar{u}_k \theta \frac{\partial \bar{U}_k}{\partial x_j} \quad (63)$$

where $\tau_u = k/\varepsilon$.

The production and Coriolis terms in equation (60) are given by:

$$P_{j\theta} = -\bar{u}_j \bar{u}_k \frac{\partial \bar{\Theta}}{\partial x_k} - \bar{u}_k \theta \frac{\partial \bar{U}_j}{\partial x_k} \quad (64)$$

$$C_{j\theta} = -2\varepsilon_{jm} \Omega_m \bar{u}_k \theta \quad (65)$$

Therefore, the first term on the right-hand side of equation (62) is expanded as follows:

$$\begin{aligned} \frac{1}{\sqrt{k} \sqrt{k_\theta}} (P_{j\theta} + \Phi_{j\theta} - \varepsilon_{j\theta} + C_{j\theta}) &= \frac{1}{\sqrt{k} \sqrt{k_\theta}} \frac{C_1}{\tau_u} \left[-\bar{u}_j \theta - \frac{1}{C_1} \tau_u \bar{u}_j \bar{u}_k \frac{\partial \bar{\Theta}}{\partial x_k} \right. \\ &\quad \left. - \frac{1 - C_2}{C_1} \tau_u \bar{u}_k \theta \frac{\partial \bar{U}_j}{\partial x_k} + \frac{C_3}{C_1} \tau_u \bar{u}_k \theta \frac{\partial \bar{U}_k}{\partial x_j} - \frac{2}{C_1} \tau_u \varepsilon_{jm} \Omega_m \bar{u}_k \theta \right] \end{aligned} \quad (66)$$

where the model constants in equation (66) are replaced as $C_{T1} = 1/C_1$, $C_{T2} = (1 - C_2)/C_1$, and $C_{T3} = C_3/C_1$. Thus:

$$\begin{aligned} \frac{1}{\sqrt{k} \sqrt{k_\theta}} (P_{j\theta} + \Phi_{j\theta} - \varepsilon_{j\theta} + C_{j\theta}) &= \frac{1}{\sqrt{k} \sqrt{k_\theta}} \frac{1}{C_{T1} \tau_u} \left[-\bar{u}_j \theta - C_{T1} \tau_u \bar{u}_j \bar{u}_k \frac{\partial \bar{\Theta}}{\partial x_k} \right. \\ &\quad \left. - (C_{T2} - C_{T3}) \tau_u \bar{u}_k \theta S_{jk} - (C_{T2} + C_{T3}) \tau_u \bar{u}_k \theta \left(W_{jk} + \frac{2C_{T1}}{C_{T2} + C_{T3}} \varepsilon_{jm} \Omega_m \right) \right] \end{aligned} \quad (67)$$

Consequently, substituting equation (67) for equation (62), the following equation is obtained:

$$\begin{aligned} \frac{D\bar{a}_j^*}{Dt} = & \frac{1}{\sqrt{k}\sqrt{k_\theta}} \frac{1}{C_{\theta 1}\tau_u} \left[-\bar{u}_j\bar{\theta} - C_{\theta 1}\tau_u\bar{u}_j\bar{u}_k \frac{\partial\bar{\Theta}}{\partial x_k} - C_{\theta 2}\tau_u\bar{u}_k\bar{\theta}S_{jk} \right. \\ & \left. - C_{\theta 3}\tau_u\bar{u}_k\bar{\theta}(W_{jk} + C_{\theta 4}\epsilon_{jm k}\Omega_m) \right] \\ & - \frac{1}{2} \frac{\bar{u}_j\bar{\theta}}{\sqrt{k}\sqrt{k_\theta}} \left[\frac{\varepsilon}{k} \left(\frac{P_k}{\varepsilon} - 1 \right) + \frac{\varepsilon_\theta}{k_\theta} \left(\frac{P_{k_\theta}}{\varepsilon_\theta} - 1 \right) \right] \end{aligned} \quad (68)$$

where $C_{\theta 1} = C_{T1}$, $C_{\theta 2} = C_{T2} - C_{T3}$, $C_{\theta 3} = C_{T2} + C_{T3}$, and $C_{\theta 4} = 2C_{T1}/(C_{T2} + C_{T3})$.

In the local equilibrium state, the following relation holds:

$$\frac{D\bar{a}_j^*}{Dt} = 0 \quad (69)$$

With the above condition of equation (69), equation (68) becomes as follows:

$$\begin{aligned} & \frac{\bar{u}_j\bar{\theta}}{\sqrt{k}\sqrt{k_\theta}} \frac{1}{\tau_u} \left[1 + \frac{C_{\theta 1}}{2} \left(\frac{P_k}{\varepsilon} - 1 \right) + \frac{C_{\theta 1}}{2R} \left(\frac{P_{k_\theta}}{\varepsilon_\theta} - 1 \right) \right] \\ & = -(\delta_{jk} + 3b_{jk}) \frac{2}{3} \frac{C_{\theta 1}k}{\sqrt{k}\sqrt{k_\theta}} \frac{\partial\bar{\Theta}}{\partial x_k} - (C_{\theta 2}S_{jk} + C_{\theta 3}W'_{jk}) \frac{\bar{u}_k\bar{\theta}}{\sqrt{k}\sqrt{k_\theta}} \end{aligned} \quad (70)$$

where $W'_{jk} = W_{jk} + C_{\theta 4}\epsilon_{jm k}\Omega_m$, and the following characteristic time-scale in the left-hand side of equation (70) is replaced with the mixed time-scale τ_m for the near-wall treatment:

$$\frac{\tau_u}{1 + \frac{C_{\theta 1}}{2} \left(\frac{P_k}{\varepsilon} - 1 \right) + \frac{C_{\theta 1}}{2R} \left(\frac{P_{k_\theta}}{\varepsilon_\theta} - 1 \right)} \rightarrow \tau_m \quad (71)$$

Finally, if we introduce the nondimensional given by the following equations:

$$\left. \begin{aligned} b_{jk}^* &= 3b_{jk}, \\ \Theta_k^* &= \frac{2}{3} \frac{C_{\theta 1}k\tau_m}{\sqrt{k}\sqrt{k_\theta}} \frac{\partial\bar{\Theta}}{\partial x_k}, \\ S_{jk}^* &= C_{\theta 2}\tau_m S_{jk}, \\ W_{jk}^* &= C_{\theta 3}\tau_m W_{jk}. \end{aligned} \right\} \quad (72)$$

Thus, equation (70) becomes:

$$a_j^* = -(\delta_{jk} + b_{jk}^*)\Theta_k^* - (S_{jk}^* + W_{jk}^*)a_k^* \quad (73)$$

Thus, we obtain the following explicit expression for the nondimensional turbulent heat flux [45]:

$$a_j^* = \frac{1}{1 + \frac{1}{2} (W^{*2} - S^{*2})} [-\delta_{jk} - b_{jk}^* + (S_{jk}^* + W_{jk}^*) + (S_{j\ell}^* + W_{j\ell}^*) b_{\ell k}^*] \Theta_k^* \quad (74)$$

where $W^{*2} = W_{ij}^* W_{ij}^*$ and $S^{*2} = S_{ij}^* S_{ij}^*$. The derived equation (74) can be rewritten in the conventional form as follows:

$$\begin{aligned} \overline{u_j \theta} &= - \frac{C_{\theta 1} \tau_m}{f_{RT}} \overline{u_j u_k} \frac{\partial \overline{\Theta}}{\partial x_k} \\ &+ \frac{C_{\theta 1} \tau_m^2}{f_{RT}} \overline{u_k u_\ell} \left(C_{\theta 2} S_{j\ell} \frac{\partial \overline{\Theta}}{\partial x_k} + C_{\theta 3} W_{j\ell} \frac{\partial \overline{\Theta}}{\partial x_k} + 2C_{\theta 1} \epsilon_{jm\ell} \Omega_m \frac{\partial \overline{\Theta}}{\partial x_k} \right) \end{aligned} \quad (75)$$

where $C_{\theta 4} = 2C_{\theta 1}/C_{\theta 3}$ and $f_{RT} = 1 + \frac{1}{2} \tau_m^2 (C_{\theta 3}^2 W^2 - C_{\theta 2}^2 S^2)$, and the first term of right-hand side can be rewritten using the following anisotropic eddy diffusivity tensor for heat:

$$\alpha_{jk}^t = \frac{C_{\theta 1}}{f_{RT}} \overline{u_j u_k} \tau_m \quad (76)$$

Thus, the final form for NLEDHM given in equation (56) is obtained as follows:

$$\overline{u_j \theta} = -\alpha_{jk}^t \frac{\partial \overline{\Theta}}{\partial x_k} + \frac{C_{\theta 1}}{f_{RT}} \tau_m^2 \overline{u_k u_\ell} (C_{\theta 2} S_{j\ell} + C_{\theta 3} W_{j\ell} + 2C_{\theta 1} \epsilon_{jm\ell} \Omega_m) \frac{\partial \overline{\Theta}}{\partial x_k} \quad (77)$$

Based on the foregoing, the NLEDHM should be improved to accurately predict heated channel flows with arbitrary rotating axes and various rotation numbers. Although the NLHN properly predicts the rotating heated channel flow with arbitrary rotating axes, the streamwise turbulent heat flux is slightly underpredicted. Therefore, in order to construct a turbulence model having higher-prediction precision, we reconstruct the NLEDHM based on the NLHN model [38].

As for a prediction of streamwise turbulent heat flux, Hishida et al. [46] proposed the relation $-\overline{u \theta} = C_h (-\overline{uv}/k) v \overline{\theta}$ from an experimental result, and Hattori et al. [24] proposed the modified temperature-pressure gradient correlation and dissipation terms for a buoyancy-affected thermal field, in which the prediction precision of NLEDHM is improved. In this study, we introduce the dimensionless Reynolds stress tensor $A_{ij} (= \overline{u_i u_j}/k)$ in an anisotropic eddy diffusivity tensor for heat as indicated in equation (60), which is adopted in the NLSA model [43] to adequately predict the streamwise turbulent heat flux, for the NLHN model indicated in equation [56]. Also, introducing the dimensionless Reynolds stress tensor

into the NLEDHM, the NLEDHM successfully predicts turbulent buoyant flows as mentioned in Hattori et al. [24].

$$\alpha_{jk}^t = \left(\frac{C_{\theta 1}^*}{f_{RT}} \right) \overline{u_i u_j} A_{ik} \tau_m \quad (78)$$

$$f_{RT} = 1 + \frac{1}{2} \tau_m^2 [C_{\theta 2}^2 (W^2 - S^2) + (C_{\theta 3}^2 - C_{\theta 2}^2) W^2] \quad (79)$$

where $C_{\theta 1}^*$, $C_{\theta 2}$, and $C_{\theta 3}$ are model constants and τ_m is the characteristic time-scale.

In introducing the dimensionless Reynolds stress tensor A_{ij} , $\alpha_{jk}^t (\partial \bar{\Theta} / \partial x_k)$ of the streamwise turbulent heat flux, $\overline{u \theta}$ ($j = 1$), is given as follows:

$$\alpha_{xy}^t = \left(\frac{C_{\theta 1}^*}{f_{RT}} \right) \left[\frac{\overline{uv}}{k} (\overline{u^2} + \overline{v^2}) \right] \tau_m \frac{\partial \bar{\Theta}}{\partial y} \quad (80)$$

where, regarding the derivation of equation (80), a fully developed channel flow is assumed.

The original model of equation (57) makes for the streamwise turbulent heat flux $\alpha_{xy}^t = (C_{\theta 1}^* / f_{RT}) \overline{v^2} \tau_m$. Comparing both equations, the effect of dimensionless Reynolds stress tensor for the prediction of the streamwise turbulent heat flux is obviously obtained, i.e., the streamwise velocity intensity, $\overline{u^2}$, which has a strong correlation with $\overline{u \theta}$ [47], evidently improves the prediction.

8.9 Model Performances

8.9.1 Prediction of rotating channel flow using NLEDMM

The evaluations for the newly improved NLEDM (or NLEDMM) are shown in Figures 8.32–8.42. In order to calculate the present model, the numerical technique used is a finite-volume method [48].

First, cases of wall-normal rotating channel flows are shown in Figures [8.32–8.42]. In this case, the streamwise mean velocity decreases with the increasing rotation number due to the exchange momentum between the streamwise and spanwise velocity as indicated in Figures 8.32a and c. Thus, the spanwise velocity increases at the lower rotation numbers, but the spanwise velocity reaches maximum value at a critical rotation number. Then, the spanwise velocity decreases because the streamwise velocity becomes almost zero at higher rotation numbers. In comparison with DNS results, the present NLEDMM gives better predictions than the predictions of the NLCLS model. In particular, Reynolds shear stresses, \overline{vw} , are predicted very well as shown in Figure 8.32d. Figure 8.33 shows relation between streamwise and spanwise velocities in this case, in which the laminar solutions are included for comparison. Obviously, the flow tends to be laminarized with the increasing rotation number, and the present model can predict properly this tendency. Also, anisotropy in turbulent intensities is reproduced by the present cubic

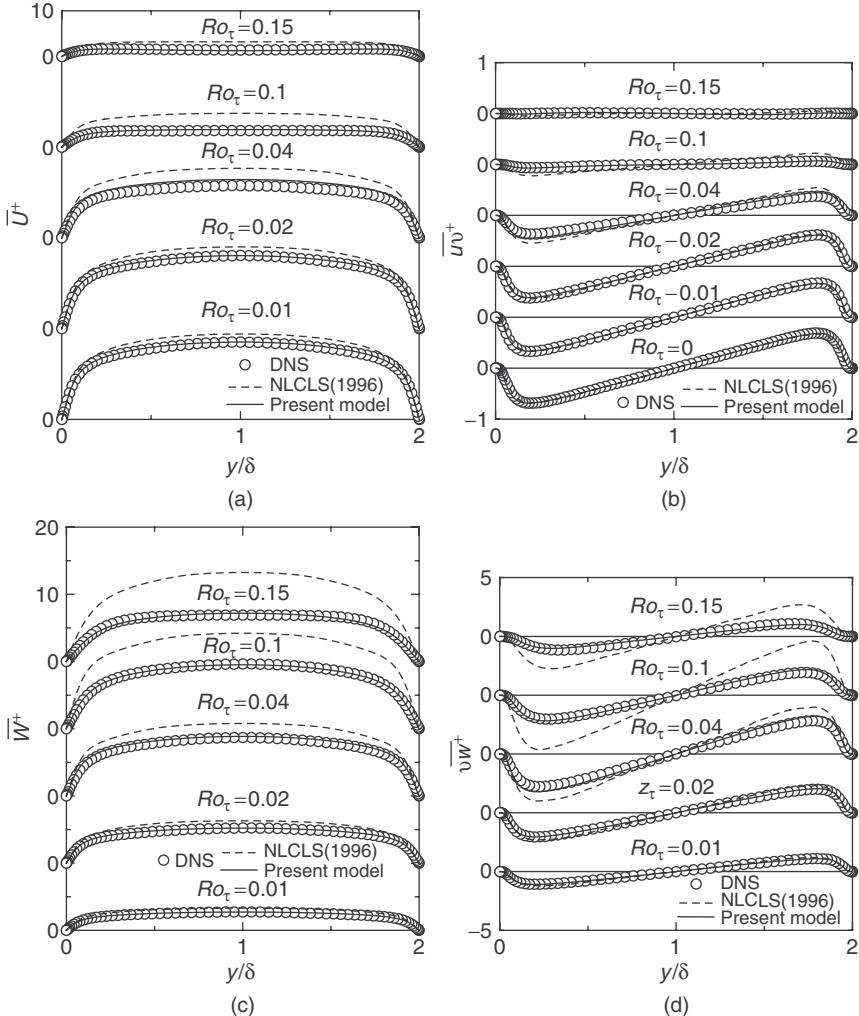


Figure 8.32. Distributions of predicted wall-normal rotating flows (Case 1):
 (a) streamwise mean velocities, (b) Reynolds shear stresses, \bar{uv} ,
 (c) spanwise mean velocities, and (d) Reynolds shear stresses, \bar{vw} .

NLEDMM as shown in Figure 8.34a. As mentioned above, the NLEDMM can predict a turbulent velocity field similar in many respects to the NLHN model as shown in Figure 8.28, but the wall-limiting behavior is satisfied exactly by the improved model, as indicated in Figure 8.34b. Therefore, it is obvious that the present cubic NLEDMM gives higher performance for the predictions of rotating channel flows while maintaining the performance of the basis model, e.g., NLHN model.

Next, the predictions of fully developed streamwise rotating channel flows calculated by the present model are shown in Figure 8.35. The results with the

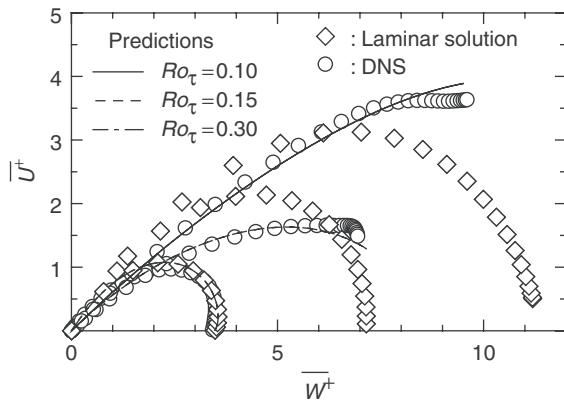
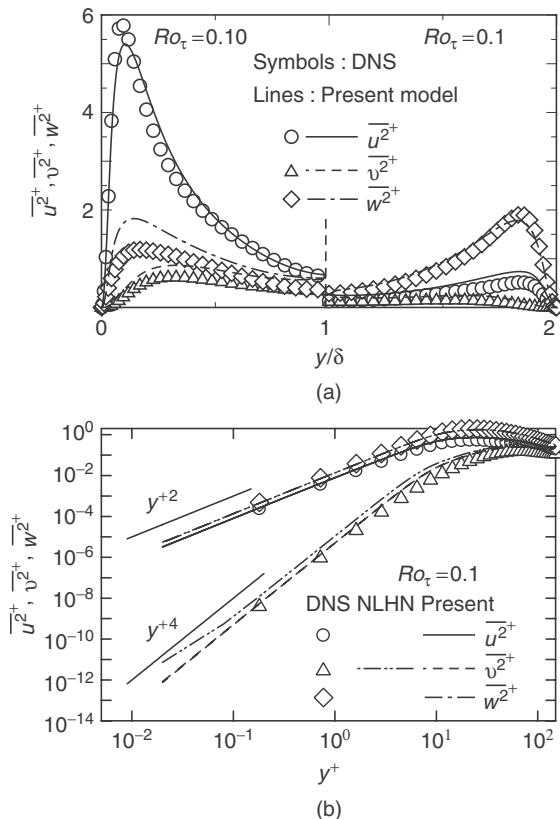
Figure 8.33. Relation of between \bar{U} and \bar{W} in wall-normal rotating channel flow.

Figure 8.34. Distributions of predicted wall-normal rotating flows (Case 1): (a) turbulent intensities and (b) wall-limiting behavior of Reynolds stress components.

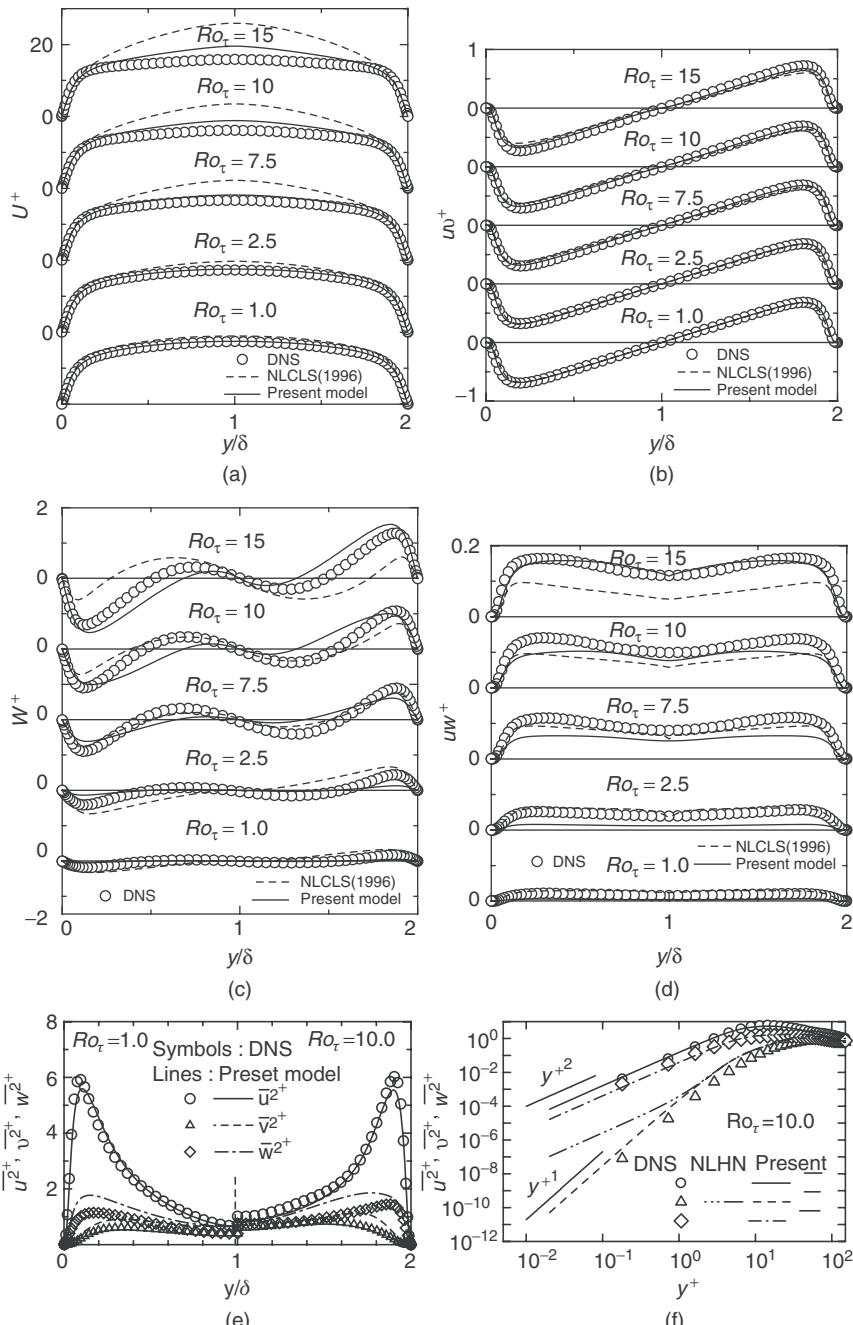


Figure 8.35. Distributions of predicted streamwise rotating flows (Case 2): (a) streamwise mean velocities, (b) Reynolds shear stresses, $\bar{u}\bar{w}$, (c) spanwise mean velocities, (d) Reynolds shear stresses, $\bar{v}\bar{w}$, (e) turbulent intensities and (f) wall-limiting behavior of Reynolds stress components.

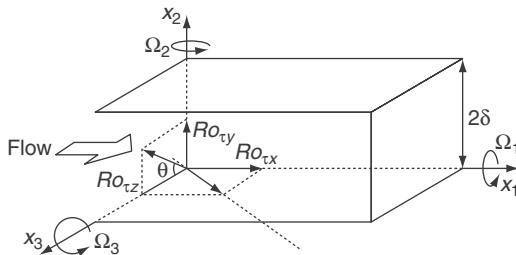


Figure 8.36. Coordinate system and arbitrary rotating axis in rotating channel flow.

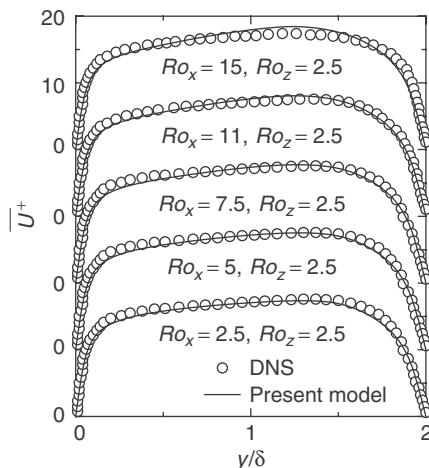


Figure 8.37. Distributions of mean velocities in an arbitrary axis rotating channel flow: Case STSP: streamwise mean velocities.

NLCLS model [40] are also included in the figure for comparison. It can be seen in Figures 8.35a–d that the present NLEDMM gives accurate predictions of both mean velocities and Reynolds shear stresses in all cases, and the test case of the highest rotational number is adequately reproduced by the NLEDMM. Also, the Reynolds normal stresses are indicated in Figures 8.35e and f. Now, the present NLEDMM adequately reproduces redistribution of Reynolds normal stresses, and can predict wall-limiting behaviour exactly. These are because the modified characteristic time-scale τ_{Rw} given as equation (46) is introduced in the proposed NLEDMM.

To confirm the performance of the NLEDMM model, cases of rotating channel flow with *combined rotational axes* are calculated as shown in Figure 8.36, in which DNS databases of these cases are provided by Wu and Kasagi [17]. The rotation numbers of Case STSP are given as $Ro_{tx} = 2.5 \sim 15$ and $Ro_{tz} = 2.5$, those of Case WNSP1 as $Ro_{ty} = 0.04$ and $Ro_{tz} = 2.5 \sim 11$ and those of Case WNSP2 as $Ro_{ty} = 0.01 \sim 0.04$ and $Ro_{tz} = 2.5$, respectively. The predicted mean velocities are

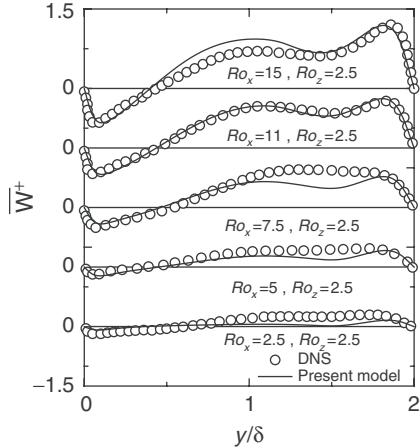


Figure 8.38. Distributions of mean velocities in an arbitrary axis rotating channel flow: Case STSP: spanwise mean velocities.

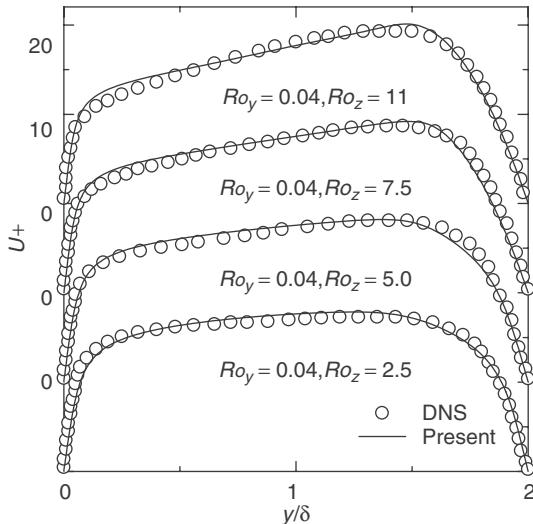


Figure 8.39. Distributions of mean velocities in an arbitrary axis rotating channel flow: Case WNSP1: streamwise mean velocities.

shown in Figures 8.37–8.42 with DNS data [17]. It can be seen that the proposed NLEDMM gives accurate predictions of all cases, since the proper modeling for rotating flows with arbitrary rotating axes is introduced in the present two-equation model with the cubic NLEDMM.

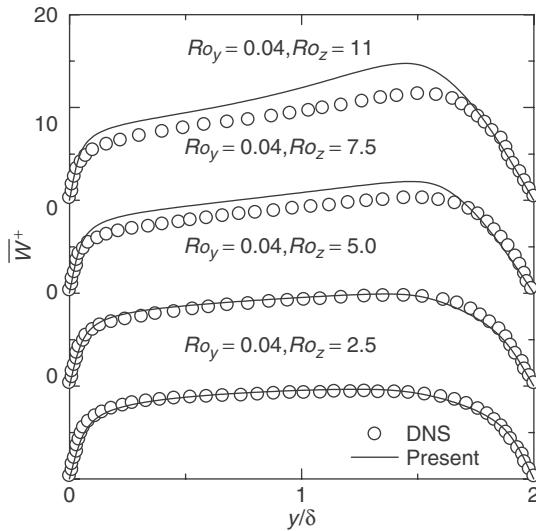


Figure 8.40. Distributions of mean velocities in an arbitrary axis rotating channel flow: Case WNSP1: spanwise mean velocities.

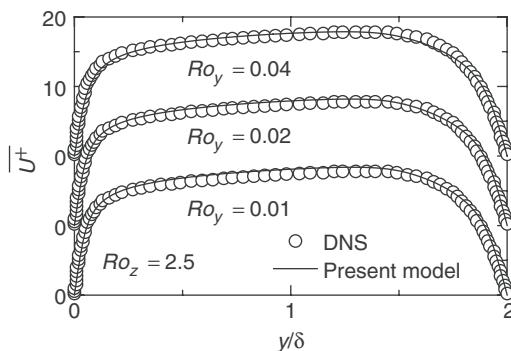


Figure 8.41. Distributions of mean velocities in an arbitrary axis rotating channel flow: Case WNSP2: streamwise mean velocities.

8.9.2 Prediction of rotating channel-flow heat transfer using NLEDHM

Calculations identical with DNS conditions indicated in Table 8.5 using turbulence models are carried out in comparison with DNS results. Again, Figures 8.32 and 8.35 indicate predicted streamwise and spanwise mean velocities in both cases, and it is obvious that mean velocities in both cases are adequately predicted by the NLEDMM. Since a thermal field is passive scalar, turbulent quantities of velocity field should be accurately predicted by the turbulence model for the velocity field. Therefore, the proposed NLEDHM can closely incorporate the NLEDMM.

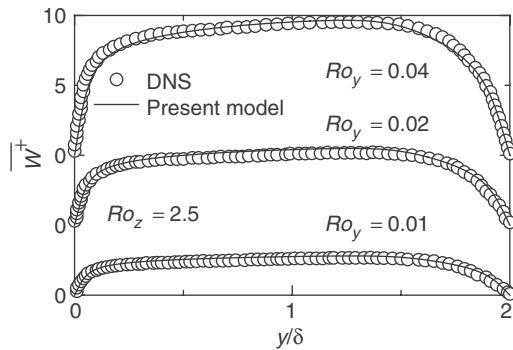


Figure 8.42. Distributions of mean velocities in an arbitrary axis rotating channel flow: Case WNSP2: spanwise mean velocities.

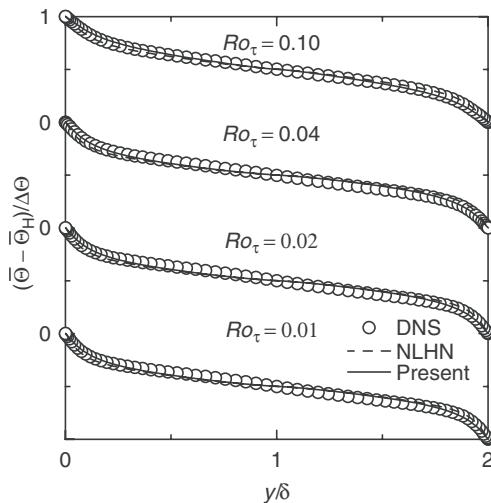


Figure 8.43. Predicted mean temperatures (Case 1).

Note that the following results are given from a calculation of full transport equations given in equations (4)–(12) with equation (44) and the present NLEDHM or the NLHN model.

First, the distributions of mean temperature, which are estimated by the present NLEDHM are shown in Figure 8.43. The NLHN model (16) is also included in the figures for comparison. Accurate prediction of the mean temperature of Case 1 can be done with both models. On the other hand, the proposed NLEDHM is apparently improved to predict a streamwise turbulent heat flux as indicated in Figure 8.44a. Although the streamwise turbulent heat flux does not affect the distribution of mean temperature directly, the turbulent heat transfer model should

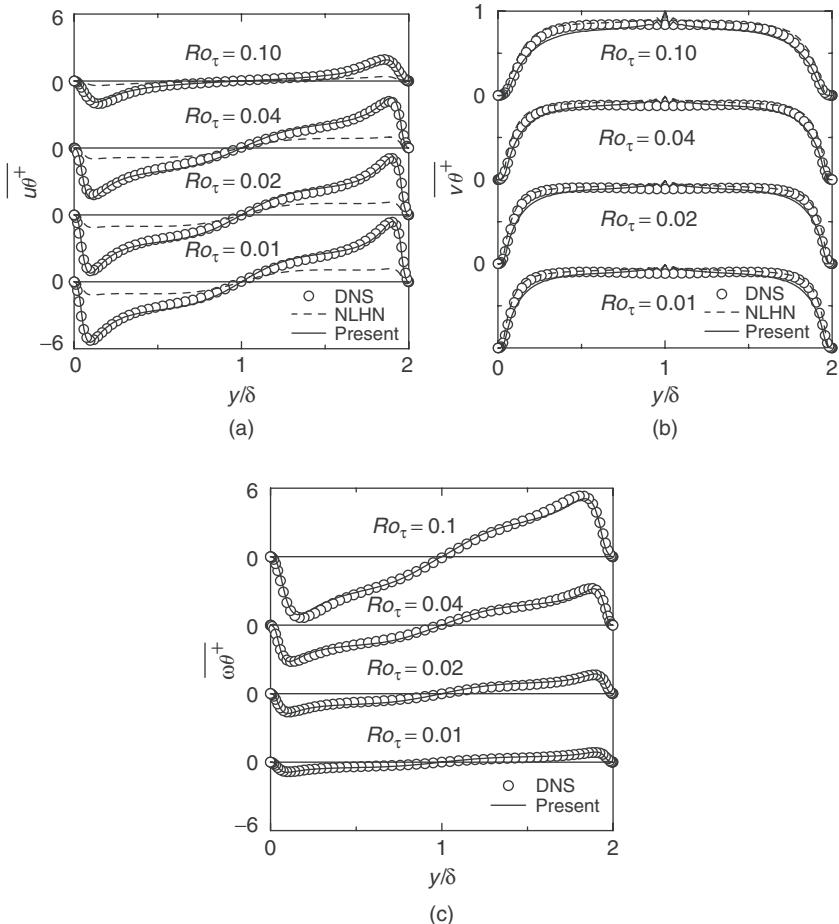


Figure 8.44. Predicted turbulent heat fluxes (Case 1); (a) streamwise ($\overline{u\theta}$), (b) wall-normal ($\overline{v\theta}$), (c) spanwise ($\overline{w\theta}$).

predict the streamwise turbulent heat flux to apply some turbulent heat transfer problem such as buoyant flow. As for the predicted wall-normal heat flux as shown in Figure 8.44b, a discontinuous line can be faintly seen in the channel center region. This is caused by the predicted wall-normal velocity intensity. Since the wall-normal velocity intensity is slightly overpredicted in this case by the NLEDMM, the wall-normal turbulent heat flux is remarkably affected by the prediction. The predicted spanwise turbulent heat flux is shown in Figure 8.44c. The present predictions are in good agreement with DNS. Figure 8.45 shows the wall-limiting behavior of both streamwise and wall-normal turbulent heat fluxes. Since the heat is carried from the wall in the case of the heated wall condition, the wall-limiting behavior should be reproduced to exactly estimate heat transfer rate. It can be seen that the present

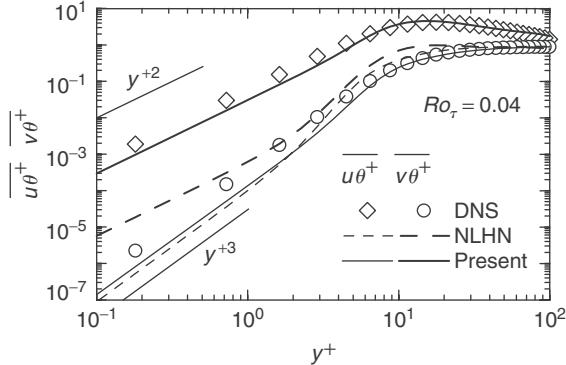


Figure 8.45. Wall-limiting behavior of turbulent heat fluxes (Case 1).

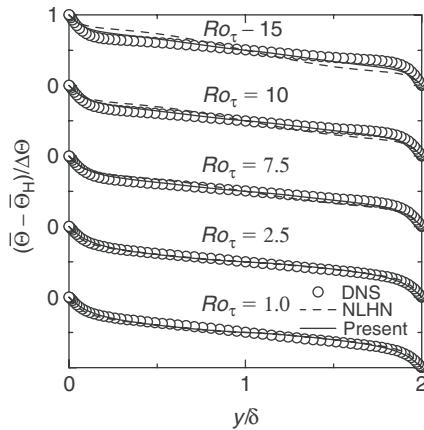


Figure 8.46. Predictions of mean temperature (Case 2).

NLEDHM appropriately predicts both turbulent heat fluxes. Note that the present NLEDHM exactly satisfies the difference between turbulent heat fluxes, i.e., the streamwise turbulent heat flux is proportional to y^2 , and the wall-normal turbulent heat flux is proportional to y^3 .

In Case 2, although the mean temperature indicated in Figure 8.46 is slightly underpredicted by the NLHN model in the highest rotation number, the proposed model can satisfactorily reproduce it. Also, in this case, it can be seen that the proposed NLEDHM adequately predicts both the streamwise and the wall-normal turbulent heat fluxes as shown in Figure 8.47. On the other hand, it is difficult for the present model to predict the spanwise turbulent heat flux in Case 2. The spanwise

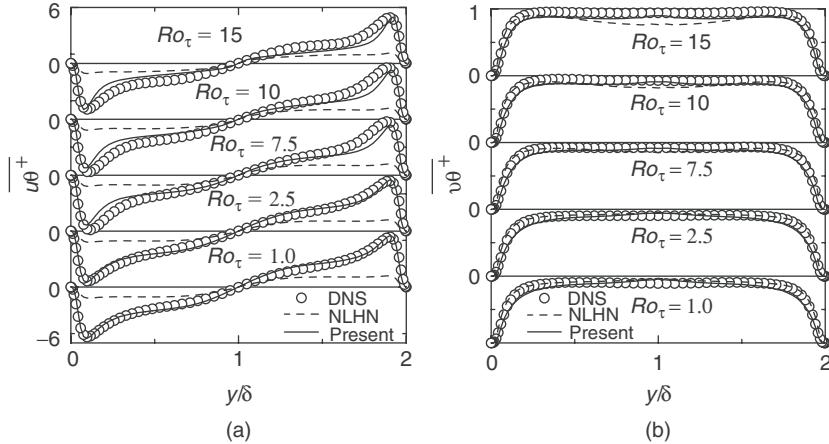


Figure 8.47. Predicted turbulent heat flux (Case 2); (a) streamwise ($\bar{u}\theta$), (b) wall-normal ($\bar{v}\theta$).

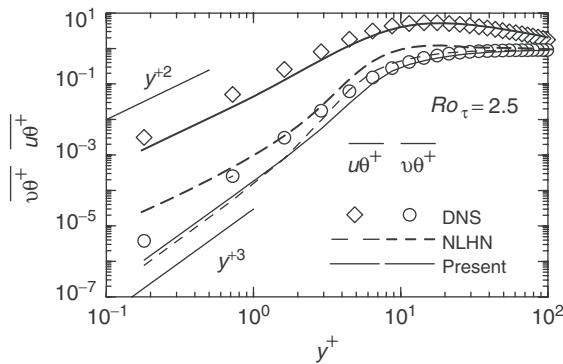


Figure 8.48. Wall-limiting behavior of turbulent heat fluxes (Case 2).

turbulent heat flux, $\bar{w}\theta$, is expressed by the present model in Case 2 as follows:

$$\begin{aligned} \bar{w}\theta &= - \left(\frac{C_{\theta 1}^*}{f_{RT}} \right) \left(\frac{\bar{u}w}{k} + \frac{\bar{v}w}{k} \right) \tau_{mo} \frac{\partial \bar{\Theta}}{\partial y} \\ &\quad + \frac{C_{\theta 1}}{f_{RT}} \tau_{mo}^2 \bar{v}^2 \left[(C_{\theta 2} + C_{\theta 3}) \frac{\partial \bar{W}}{\partial y} + 2C_{\theta 1}\Omega_1 \right] \frac{\partial \bar{\Theta}}{\partial y} \end{aligned} \quad (81)$$

The wall-limiting behaviors of turbulent heat fluxes are shown in Figure 8.48, in which the wall-limiting behaviors are properly satisfied by the present NLEDHM.

On the basis of these evidences, it would not be an overstatement to say that the proposed model can be applied to predict the rotating heated channel flows with arbitrary rotating axes.

8.10 Concluding Remarks

Based on our current works, the recent trends in DNS and turbulence models for velocity and thermal fields have been overviewed in this chapter. Without dispute, the DNS gives us a possibility to obtain highly accurate information on various turbulence quantities. Within the foreseeable future, DNS researchers will actively tackle more complex problems, e.g., flows at high Reynolds or Prandtl numbers, buoyancy-driven complex flows, and atmospheric boundary layer flows. Of course, it is true that the power of the present supercomputer system does not offer realistic solutions to the above problems. However, improvement of computer systems, in particular, the further development of massive parallel-computing systems will conquer most of the serious difficulties which confront us now. In fact, there are emerging industries which aim for petaflops (PFLOPS) computing. New researches concerning exascale supercomputing design are also in progress.

The latest turbulence models for velocity and thermal fields deal with the individual elemental processes of turbulence with the aid of detailed information given by DNS. As a result, such models can mimic the DNS with high accuracy. Also, many attempts to construct a universal turbulence model are being made. If a number of DNS databases on complex turbulent flows are constructed in the future, we will be able to approach the universal turbulence model.

Acknowledgment

This study was partially supported by a Grant-in-Aid for Scientific Research (S), 17106003, from the Japan Society for the Promotion of Science (JSPS).

References

- [1] Robinson, S. K. Coherent motions in the turbulent boundary layer, *Annu. Rev. Fluid Mech.*, 23, pp. 601–639, 1991.
- [2] Kasagi, N. Direct numerical simulation data bases: an effective tool in fundamental studies of turbulent heat transfer, *Computers and Computing in Heat Transfer Science and Engineering*, W. Nakayama and K.T. Yang, eds., CRC Press, Florida, pp. 97–117, 1993.
- [3] Mansour, N. N., Kim, J., and Moin, P. Reynolds-stress and dissipation-rate budgets in a turbulent channel flow, *J. Fluid Mech.*, 194, pp. 15–44, 1988.
- [4] Nagano, Y., and Shimada, M. Critical assessment and reconstruction of dissipation-rate equations using direct simulations, *The Recent Developments in Turbulence Research*, Z. S. Zhang and Y. Miyake, eds., International Academic Publishers, Beijing, pp. 189–217, 1994.

- [5] Rogers, M. M., Mansour, N. N., and Reynolds, W. C. An algebraic model for the turbulent flux of a passive scalar, *J. Fluid Mech.*, 203, pp. 77–101, 1989.
- [6] Kim, J., Moin, P., and Moser, R. Turbulence statistics in fully developed channel flow at low Reynolds number, *J. Fluid Mech.*, 177, pp. 133–166, 1987.
- [7] Moser, R. D., Kim, J., and Mansour, N. N. Direct numerical simulation of turbulent channel flow up to $Re_\tau = 590$, *Phys. Fluids*, 11, pp. 943–945, 1999.
- [8] Spalart, P. R. Direct simulation of a turbulent boundary layer up to $R_\theta = 1410$, *J. Fluid Mech.*, 187, pp. 61–98, 1988.
- [9] Spalart P. R., and Watmuff, J. H. Experimental and numerical study of a turbulent boundary layer with pressure gradients, *J. Fluid Mech.*, 249, pp. 337–371, 1993.
- [10] Le, J.-H., and Sung, H. J. Effects of an adverse pressure gradient on a turbulent boundary layer, *Int. J. Heat Fluid Flow*, 29, pp. 568–578, 2008.
- [11] Le, H., Moin, P., and Kim, J. Direct numerical simulation of turbulent flow over a backward-facing step, *J. Fluid Mech.*, 330, pp. 349–374, 1997.
- [12] Choi, H., Moin, P., and Kim, J. Direct numerical simulation of turbulent flow over ripples, *J. Fluid Mech.*, 255, pp. 503–539, 1993.
- [13] Nagano, Y., Hattori, H., and Houra, T. DNS of velocity and thermal fields in turbulent channel flow with transverse-rib roughness, *Int. J. Heat Fluid Flow*, 25, pp. 393–403, 2004.
- [14] Kristoffersen, R., and Andersson, H. I. Direct simulations of low-reynolds-number turbulent flow in a rotating channel, *J. Fluid Mech.*, 256, pp. 163–195, 1993.
- [15] Oberlack, M., Cabot, W., and Rogers, M. M. Turbulent channel flow with streamwise rotation: lie group analysis, DNS and modeling, *Proc. 1st Int. Symp. on Turbulence and Shear Flow Phenomena*, pp. 85–90, 1999.
- [16] Nagano, Y., and Hattori, H. DNS and modelling of spanwise rotating channel flow with heat transfer, *J. Turbulence*, 4-010, pp. 1–15, 2003.
- [17] Wu, H., and Kasagi, N. Effects of arbitrary directional system rotation on turbulent channel flow, *Phys. Fluids*, 16, pp. 979–990, 2004.
- [18] Hattori, H., Ohiwa, N., and Nagano, Y. Nonlinear eddy diffusivity model for wall-bounded flow with arbitrary rotating axes, *Int. J. Heat Fluid Flow*, 27, pp. 838–851, 2006.
- [19] Huser, A., and Biringen, S. Direct numerical simulation of turbulent flow in a square duct, *J. Fluid Mech.*, 257, pp. 65–95, 1993.
- [20] Sumitani, Y., and Kasagi, N. Direct numerical simulation of turbulent transport with uniform wall injection and suction, *AIAA J.*, 33, pp. 1220–1228, 1995.
- [21] Iida, O., and Kasagi, N. Direct numerical simulation of unstably stratified turbulent channel flow, *Trans. ASME, J. Heat Transfer*, 119, pp. 53–61, 1997.
- [22] Hattori, H., and Nagano, Y. Direct numerical simulation of turbulent heat transfer in plane impinging jet, *Int. J. Heat Fluid Flow*, 25, pp. 749–758, 2004.
- [23] Hattori, H., Houra, T., and Nagano, Y. Direct numerical simulation of stable and unstable turbulent thermal boundary layers, *J. Heat Fluid Flow*, 28, pp. 1262–1271, 2007.
- [24] Hattori, H., Morita, A., and Nagano, Y. Nonlinear eddy diffusivity models reflecting buoyancy effect for wall shear flows and heat transfer, *Int. J. Heat Fluid Flow*, 27, pp. 671–683, 2006.
- [25] Speziale, C. G., Sarkar, S., and Gatski, T. B. Modelling the pressure-strain correlation of turbulence: an invariant dynamics system approach, *J. Fluid Mech.*, 227, pp. 245–272, 1997.

- [26] Kajishima, T. Finite-difference method for convective terms using non-uniform grid, *Trans. JSME (B)*, 65, pp. 1607–1612, 1999.
- [27] Perry, A. E., Schofield, W. H., and Joubert, P. N. Rough wall turbulent boundary layers, *J. Fluid Mech.*, 37, pp. 383–413, 1969.
- [28] Townsend, A. A. *The Structure of Turbulent Shear Flow*, 2nd Edition, Cambridge University Press, Cambridge, pp. 139–143, 1976.
- [29] Hanjalić, K., and Launder, B. E. Fully developed asymmetric flow in a plane channel, *J. Fluid Mech.*, 51, pp. 301–335, 1972.
- [30] Kasagi, N., and Nishimura, M. Direct Numerical simulation of combined forced and natural turbulent convection in a vertical plane channel, *Int. J. Heat Fluid Flow*, 18, pp. 88–99, 1997.
- [31] Kays, W. M., and Crawford, M. E. *Convective Heat and Mass Transf.*, 3rd Edition, McGraw-Hill, New York, pp. 323–325, 1993.
- [32] Dean, R. B. Reynolds number dependence of skin friction and other bulk flow variables in two-dimensional rectangular duct flow, *Trans. ASME, J. Fluid Engng.*, 100, pp. 215–223, 1978.
- [33] Ashrafiān, A., and Andersson, H. I. DNS of turbulent flow in a rod-roughened channel, *Proc. 3rd Int. Symp. on Turbulence and Shear Flow Phenomena*, 1, pp. 117–122, 2003.
- [34] Leonardi, S., Orlandi, P., Smalley, R. J., Djenidi, L., and Antonia, R. A. Direct numerical simulations of turbulent channel flow with transverse square bars on one wall, *J. Fluid Mech.*, 491, pp. 229–238, 2003.
- [35] Vogel, J. C., and Eaton, J. K. Combined heat transfer and fluid dynamic measurements downstream of a backward-facing step, *Trans. ASME, J. Heat Transfer*, 107, pp. 922–929, 1985.
- [36] Kasagi, N., Sumitani, Y., Suzuki, Y., and Iida, O. Kinematics of the quasi-coherent vortical structure in near-wall turbulence, *Int. J. Heat Fluid Flow*, 16, pp. 2–10, 1995.
- [37] Iida, O., Iwatsuki, M., and Nagano, Y. Vortical turbulence structure and transport mechanism in a homogeneous shear flow, *Phys. Fluids*, 12, pp. 2895–2905, 2000.
- [38] Hattori, H., Ohiwa, N., Kozuka, M., and Nagano, Y. Improvement of nonlinear eddy diffusivity model for rotational turbulent heat transfer at various rotating axes, *Fluid Dynam. Res.*, 41, 2009.
- [39] Craft, T. J., Launder, B. E., and Suga, K. Development and application of a cubic eddy-viscosity model of turbulence, *Int. J. Heat Fluid Flow*, 17, pp. 108–115, 1996.
- [40] Nagano, Y., and Hattori, H. An improved turbulence model for rotating shear flows, *J. Turbulence*, 3, pp. 1–13, 2002.
- [41] Gatski, T. B., and Speziale, C. G. On explicit algebraic stress models for complex turbulent flows, *J. Fluid Mech.*, 254, pp. 59–78, 1993.
- [42] Pope, S. B. A more general effective-viscosity hypothesis, *J. Fluid Mech.*, 72, pp. 331–340, 1975.
- [43] Suga, K., and Abe, K. Nonlinear eddy viscosity modelling for turbulence and heat transfer near wall and shear-free boundaries, *Int. J. Heat Fluid Flow*, 21, pp. 37–48, 2000.
- [44] Younis, B. A., Speziale, C. G., and Clark, T. T. A Rational Model for the Turbulent Scalar Fluxes, *Proc. R. Soc. Lond. S. A.*, 461, pp. 575–594, 2005.
- [45] Abe, K., Kondoh, T., and Nagano, Y. A two-equation heat-transfer model reflecting second-moment closures for wall and free turbulent flows, *Int. J. Heat Fluid Flow*, 17, pp. 228–237, 1996.

- [46] Hishida, M., Nagano, Y., and Tagawa, M. Transport processes of heat and momentum in the wall region of turbulent pipe flow, *Proc. 8th Int. Heat Transfer Conf.*, 3, pp. 925–930, 1986.
- [47] Kim, J., and Moin, P. Transport of passive scalars in a turbulent channel flow, *Turbulent Shear Flows 6*. J. C. André, J. Cousteix, F. Durst, B. E. Launder, F. W. Schmidt, and J. H. Whitelaw, eds. Springer, Berlin, pp. 85–96, 1989.
- [48] Hattori, H., and Nagano, Y. Calculation of turbulent flows with pressure gradients using a k- ε model, *Trans. JSME*, 59-560B, pp. 1043–11048, 1995.

9 Analytical wall-functions of turbulence for complex surface flow phenomena

K. Suga
Osaka Prefecture University, Sakai, Japan

Abstract

The recently emerged analytical wall-function (AWF) methods for surface boundary conditions of turbulent flows are summarised. Since the AWFs integrate transport equations of momentum and scalars over the control volumes adjacent to surfaces with some simplifications, it is easy to include complex surface-flow physics into the wall-function formula. Hence, the AWF schemes are successful to treat turbulent flow and scalar transport near solid, smooth, rough and permeable walls. High Prandtl or Schmidt number scalar transport near walls or free surfaces are also well handled by the AWFs. This chapter, thus, summarises their rationale, fundamental equations and procedures with illustrative application results.

Keywords: Analytical wall-function, High Prandtl number, High Schmidt number, Permeable wall, Rough wall, Turbulent flows

9.1 Introduction

Wall functions have been used from the early stage of turbulence CFD. Since numerical integration of turbulence equations down to a solid wall requires dense grid nodes to resolve the steep variations of the turbulence quantities, it was virtually impossible for the people in 1970s to perform turbulence CFD of engineering flows even by the most powerful computer in those days. Moreover, flow physics very near the wall (inside sublayer), which is required for constructing a more advanced turbulence model, was not well known.

However, according to the development of computers, many studies on low Reynolds number (LRN) turbulence models have been made and such LRN models have replaced wall functions in most numerical studies (Launder [1]). The emergence of direct numerical simulations (Kim, Moin and Moser [2]) enhanced this tendency, and so many LRN turbulence models have been proposed since then.

Despite the fact that many LRN turbulence models perform satisfactorily, industrial engineers still routinely make use of classical wall-function approaches for representing near-wall turbulence and heat transfer (e.g., Ahmed and Demoulin [3]). One reason for this is that, even with advances in computing power, their near-wall resolution requirements make LRN models prohibitively expensive in complex three-dimensional industrial heat and fluid flows.

This is easily understandable when one considers flows over rough and/or porous surfaces, which are common in industrial applications. Because one cannot hope to resolve the details of tiny elements composing the rough and/or porous walls, the wall-function approach may be the only practical strategy for such industrial applications. Another difficulty of performing LRN models is treating scalar fields of high Prandtl or Schmidt numbers. In such flows, the sublayer thickness of the scalar is much thinner than that of the viscous sublayer of the flow. This requires much finer grid nodes for the scalar fields than those for flow fields.

The most part of the ‘standard’ log-law wall-function (LWF) strategies was proposed in the 1970s with the assumption of semi-logarithmic variations of the near-wall velocity and temperature (e.g. Launder and Spalding [4]). After the establishment of the LWF, Chieng and Launder [5] improved the wall function by allowing for a linear variation of both the shear stress and the turbulent kinetic energy across the near-wall cell. Amano [6] also attempted to improve the wall functions. However, all the above attempts were still based on the log-law and it is well known that such a condition does not apply in flows with strong pressure gradients and separation (e.g. Launder [1, 7]).

Thus, several attempts for developing new wall-function approaches were made. Recently, Craft et al. [8] proposed an alternative wall-function strategy. While still semi-empirical in nature, their model makes assumptions at a deeper, more general level than the log-law based schemes. This approach is called the analytical wall-function (AWF) and integrates simplified mean flow and energy equations analytically over the control volumes adjacent to the wall, assuming a near-wall variation of the turbulent viscosity. The resulting analytical expressions then produce the value of the wall shear stress and other quantities which are required over the nearwall cell.¹

Following this strategy, the present author and his colleagues discussed and proposed extensions of the AWF approach that allow for the effects of fine-grain surface roughness, porous walls and a wide range of Prandtl and Schmidt numbers [10–13]. This chapter summarises the outlines and the application results² of those extensions of the AWF method for complex surface turbulence.

¹ In the context of large eddy simulations, Cabot and Moin [9] discussed a wall model whose basic idea was similar to that of the AWF, though it used an implicit formula of the wall shear stress.

² All the results presented have been computed by in-house finite volume codes which employ the SIMPLE pressure-correction algorithm [14] with Rhie–Chow interpolation [15] and the third-order MUSCL-type scheme (e.g. [16]) for convection terms.

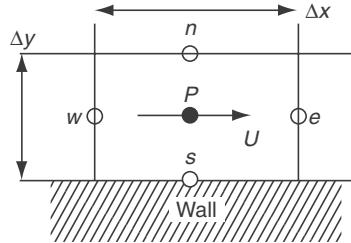


Figure 9.1. Near-wall grid arrangement.

9.2 Numerical Implementation of Wall Functions

In this section, we recall the main features of the implementation of the wall functions into a numerical code.

A simplified transport equation for ϕ near walls:

$$\frac{\partial}{\partial x}(\rho U \phi) = \frac{\partial}{\partial y} \left(\Gamma \frac{\partial \phi}{\partial y} \right) + S_\phi \quad (1)$$

can be integrated using the finite volume method over the cells illustrated in Figure 9.1 giving

$$\int_s^n \int_w^e \frac{\partial}{\partial x}(\rho U \phi) dx dy = \int_s^n \int_w^e \frac{\partial}{\partial y} \left(\Gamma \frac{\partial \phi}{\partial y} \right) dx dy + \int_s^n \int_w^e S_\phi dx dy \quad (2)$$

$$[(\rho U \phi)_e - (\rho U \phi)_w] \Delta y = \left[\left(\Gamma \frac{d\phi}{dy} \right)_n - \left(\Gamma \frac{d\phi}{dy} \right)_s \right] \Delta x + \overline{S_\phi} \Delta x \Delta y \quad (3)$$

where $\overline{S_\phi}$ is the averaged source term over the wall-adjacent cell P . Note that x is the wall-parallel coordinate while y is the wall normal coordinate. (Although two-dimensional forms are written here, extending them to three dimensions is straightforward.)

When the wall-parallel component of the momentum equation is considered ($\phi = U$) the term $(\Gamma d\phi/dy)_s$ in equation (3) corresponds to the wall shear stress τ_w , while in the energy equation it corresponds to the wall heat flux q_w . Instead of calculating these from the standard discretization, they are obtained from the algebraic wall-function expressions.

In the case of the transport equation for the turbulence energy k in incompressible flows, the averaged source term over the wall adjacent cell is written as:

$$\overline{S_k} = \overline{\rho P_k - \rho \varepsilon} = \rho(\overline{P_k} - \bar{\varepsilon}) \quad (4)$$

The terms $\overline{P_k}$ and $\bar{\varepsilon}$ thus also need to be provided by the wall function. Note that the wall function also can provide k and ε at P as the boundary conditions for their transport equations.

9.3 Standard Log-Law Wall-Function (LWF)

Before introducing the AWF, the standard wall function approach using the log-law is surveyed briefly.

The first requirement, when using the standard wall function, is that the node point P is sufficiently remote from the wall for y^+ to be much greater than that of the viscous sublayer. The value, at least, greater than 30 ($y^+ \geq 30$) is normally applied. Then, the fluxes of momentum and thermal fields to the wall are supposed to obey the following logarithmic profiles:

$$U^+ = \frac{1}{\kappa} \ln y^+ + B \quad (5)$$

$$\Theta^+ = \frac{1}{\kappa_t} \ln y^+ + C \quad (6)$$

where the constants $\kappa = 0.38$ to 0.42 and $\kappa_t = 0.47$ to 0.48 . The constant $B = 5.0$ to 5.5 while

$$C = (3.85 \text{Pr}^{1/3} - 1.3)^2 + 1/\kappa_t \ln \text{Pr}$$

for smooth wall cases. The wall shear stress τ_w and heat flux q_w at the $(N + 1)$ step of the iterative computation are then obtained as:

$$\sqrt{\tau_w / \rho}^{(N+1)} = \left(\frac{U_P}{\frac{1}{\kappa} \ln y_P^+ + B} \right)^{(N)} \quad (7)$$

$$(q_w)^{N+1} = \left(\frac{\rho c_p U_\tau \Theta_P}{\frac{1}{\kappa_t} \ln y_P^+ + C} \right)^{(N)} \quad (8)$$

In the local equilibrium boundary layers, the eddy viscosity formula gives $-\bar{uv} = \nu_t \partial U / \partial y$, and thus the production of the turbulence kinetic energy k can be written as:

$$P_k = -\bar{uv} \frac{\partial U}{\partial y} = \nu_t \frac{\partial U}{\partial y} \left(\frac{\partial U}{\partial y} \right) \quad (9)$$

Then, since $\nu_t = c_\mu k^2 / \varepsilon$ and $P_k = \varepsilon$ in local equilibrium,³

$$\left(\frac{-\bar{uv}}{k} \right)^2 = \frac{(\nu_t \partial U / \partial y)^2}{k^2} = \frac{\nu_t}{k^2} P_k = \frac{c_\mu}{\varepsilon} P_k = c_\mu \quad (10)$$

Thus, using the following relation:

$$U_\tau = \sqrt{\tau_w / \rho} \simeq \sqrt{-\bar{uv}} \simeq \sqrt{c_\mu^{1/2} k} \simeq c_\mu^{1/4} k_P^{1/2} \quad (11)$$

³The coefficient c_μ can be obtained from the experimental results of local equilibrium boundary layers: $-\bar{uv}/k \simeq 0.3$, as $c_\mu = (-\bar{uv})^2/k^2 = 0.3^2 = 0.09$.

one can rewrite equation (7) as:

$$\tau_w = \frac{\rho c_\mu^{1/4} k_P^{1/2} U_P}{\frac{1}{\kappa} \ln(c_\mu^{1/4} y_P^*) + B} \quad (12)$$

with the relation of $y^+ = c_\mu^{1/4} y^*$ in local equilibrium [4]. Note that the superscripts meaning steps are omitted for simplification hereafter. This form avoids a shortcoming of using y^+ which becomes zero where the wall shear stress τ_w vanishes.

For turbulence energy k , one can obtain its value at P from equation (11) as:

$$k_P = \frac{U_\tau^2}{\sqrt{c_\mu}} \quad (13)$$

for the boundary condition. The dissipation rate of k is also given as:

$$\varepsilon_P = \frac{U_\tau^3}{\kappa y} \quad (14)$$

for the boundary condition at P by manipulating equations (10) and (11) with the relation $v_t/v = \kappa y^+$. Note that the friction velocity $U_\tau (= \sqrt{\tau_w/\rho})$ should be obtained from equation (7) or (12).

9.4 Analytical Wall-Function (AWF)

9.4.1 Basic strategy of the AWF

In the AWF, the wall shear stress and heat flux are obtained through the analytical solution of simplified near-wall versions of the transport equations for the wall-parallel momentum and temperature [8]. In case of the forced convection regime, the main assumption required for the analytical integration of the transport equations is a prescribed variation of the turbulent viscosity μ_t . The distribution of μ_t over the wall-adjacent cell P is modelled as in a one-equation turbulence model:

$$\mu_t = \rho c_\mu k^{1/2} \ell = \rho c_\mu k^{1/2} c_\ell y \simeq \alpha \mu y^* \quad (15)$$

where ℓ is the turbulent length scale, $\alpha = c_\ell c_\mu$ and $y^* \equiv y k_P^{1/2} / v$. In order to consider viscous sub-layer effects, instead of introducing a damping function, the profile of μ_t is modelled as:

$$\mu_t = \max\{0, \alpha \mu(y^* - y_v^*)\} \quad (16)$$

in which μ_t is still linear in y^* and grows from the edge of the viscous sub-layer: $y_v^* (\equiv y_v k_P^{1/2} / v)$.

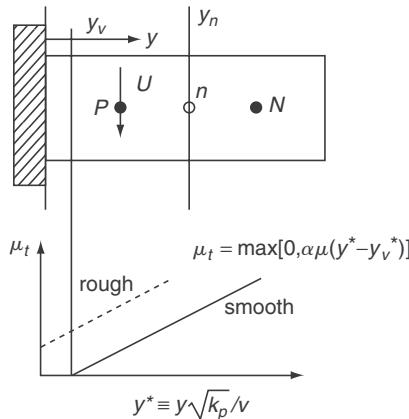


Figure 9.2. Near-wall cells.

In the context of Figure 9.2, the near-wall simplified forms of the momentum and energy equations become

$$\frac{\partial}{\partial y^*} \left[(\mu + \mu_t) \frac{\partial U}{\partial y^*} \right] = \underbrace{\frac{\nu^2}{k_p} \left[\frac{\partial}{\partial x} (\rho U U) + \frac{\partial P}{\partial x} \right]}_{C_U} \quad (17)$$

$$\frac{\partial}{\partial y^*} \left[\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t} \right) \frac{\partial \Theta}{\partial y^*} \right] = \underbrace{\frac{\nu^2}{k_p} \left[\frac{\partial}{\partial x} (\rho U \Theta) + S_\theta \right]}_{C_T} \quad (18)$$

where Pr_t is a prescribed turbulent Prandtl number, taken as 0.9. The further assumption made is that convective transport and the wall-parallel pressure gradient $\partial P / \partial x$ do not change across the wall-adjacent cell which is a standard treatment in the finite volume method. Thus, the right-hand side (rhs) terms C_U and C_T of equations (17) and (18) can be treated as constant. Then, the equations can be integrated analytically over the wall-adjacent cell giving:

if $y^* < y_v^*$

$$\frac{dU}{dy^*} = (C_U y^* + A_U) / \mu \quad (19)$$

$$\frac{d\Theta}{dy^*} = Pr(C_T y^* + A_T) / \mu \quad (20)$$

$$U = \frac{C_U}{2\mu} y^{*2} + \frac{A_U}{\mu} y^* + B_U \quad (21)$$

$$\Theta = \frac{Pr C_T}{2\mu} y^{*2} + \frac{Pr A_T}{\mu} y^* + B_T \quad (22)$$

if $y^* \geq y_v^*$

$$\frac{dU}{dy^*} = \frac{C_U y^* + A'_U}{\mu \{1 + \alpha(y^* - y_v^*)\}} \quad (23)$$

$$\frac{d\Theta}{dy^*} = \frac{\text{Pr}(C_T y^* + A'_T)}{\mu \{1 + \alpha_\theta(y^* - y_v^*)\}} \quad (24)$$

$$U = \frac{C_U}{\alpha\mu} y^* + \left\{ \frac{A'_U}{\alpha\mu} - \frac{C_U}{\alpha^2\mu} (1 - \alpha y_v^*) \right\} \ln [1 + \alpha(y^* - y_v^*)] + B'_U \quad (25)$$

$$\begin{aligned} \Theta = & \frac{\text{Pr}C_T}{\alpha_\theta\mu} y^* \\ & + \left\{ \frac{\text{Pr}A'_T}{\alpha_\theta\mu} - \frac{\text{Pr}C_T}{\alpha_\theta^2\mu} (1 - \alpha_\theta y_v^*) \right\} \ln [1 + \alpha_\theta(y^* - y_v^*)] + B'_T \end{aligned} \quad (26)$$

where $\alpha_\theta = \alpha \text{Pr}/\text{Pr}_t$. The integration constants A_U, B_U, A_T, B_T etc. are determined by applying boundary conditions at the wall, y_v and the cell face point n . The values at n are determined by interpolation between the calculated node values at P and N , whilst at y_v a monotonic distribution condition is imposed by ensuring that U, Θ and their gradients should be continuous at $y = y_v$. Notice that to determine the integration constants the empirical log-law is not referred to at all and the obtained logarithmic velocity equation (25) includes C_U . The latter implies that the velocity profile has sensitivity to the pressure gradient since C_U includes $\partial P/\partial x$.

The result is that the wall shear stress and wall heat flux can be expressed as:

$$\tau_w = \mu \left. \frac{dU}{dy} \right|_w = \mu \frac{k_p^{1/2}}{\nu} \left. \frac{dU}{dy^*} \right|_w = \frac{k_p^{1/2} A_U}{\nu} \quad (27)$$

$$q_w = - \frac{\rho c_p \nu}{\text{Pr}} \left. \frac{d\Theta}{dy} \right|_w = - \frac{\rho c_p \nu}{\text{Pr}} \frac{k_p^{1/2}}{\nu} \left. \frac{d\Theta}{dy^*} \right|_w = - \frac{\rho c_p k_p^{1/2} A_T}{\mu} \quad (28)$$

The local generation rate of k , $P_k (= \nu_t(dU/dy)^2)$, is written as:

$$P_k = \begin{cases} 0, & \text{if } y^* < y_v^* \\ \frac{\alpha k_p}{\nu} (y^* - y_v^*) \left(\frac{C_U y^* + A'_U}{\mu \{1 + \alpha(y^* - y_v^*)\}} \right)^2, & \text{if } y^* \geq y_v^* \end{cases} \quad (29)$$

which can then be integrated over the wall-adjacent cell to produce an average value \bar{P}_k for use in solving the k equation of the cell P .

For the dissipation rate ε , the following model is employed:

$$\varepsilon = \begin{cases} 2\nu k_p / y_\varepsilon^2, & \text{if } y < y_\varepsilon \\ k_p^{1.5} / (c_\ell y), & \text{if } y \geq y_\varepsilon \end{cases} \quad (30)$$

Table 9.1. Model coefficients

α	c_ℓ	c_μ	y_{vs}^*	y_ε^*	Pr_t^∞
$c_\ell c_\mu$	2.55	0.09	10.7	5.1	0.9

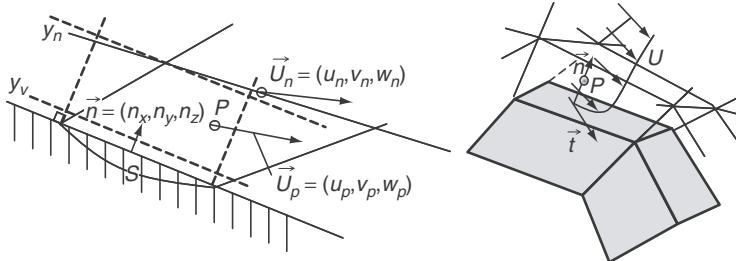


Figure 9.3. Skewed near-wall cells.

The characteristic dissipation scale y_ε can be defined as $y_\varepsilon^* = 2c_\ell$ to ensure a continuous variation of ε at y_ε . Thus, the cell averaged value is obtained as:

$$\bar{\varepsilon} = \begin{cases} 2k_P^2/(vy_\varepsilon^{*2}), & \text{if } y_\varepsilon^* > y_n^* \\ \frac{1}{y_n} \left(y_\varepsilon \frac{2vk_P}{y_\varepsilon^2} + \int_{y_\varepsilon}^{y_n} \frac{k_P^{1.5}}{c_\ell y} dy \right) = \frac{k_P^2}{vy_n^*} \left[\frac{2}{y_\varepsilon^*} + \frac{1}{c_\ell} \ln \left(\frac{y_n^*}{y_\varepsilon^*} \right) \right], & \text{if } y_\varepsilon^* \leq y_n^* \end{cases} \quad (31)$$

Through numerical experiments, the value of the constant y_v^* was optimised to be 10.7 which corresponds to approximately half the thickness of the conventionally defined viscous sub-layer of $y^+ = 11$. The other model coefficients are listed in Table 9.1.

9.4.2 AWF in non-orthogonal grid systems

In application codes which are written in either structured or unstructured grid method, the near-wall cells are not always orthogonal to the walls as shown in Figure 9.3. Thus, the way of obtaining the cell face values, U_n and y_n , for the AWF approach is as follows.

When the velocity vector \vec{U}_P at the node P is decomposed to the normal and tangential directions of the wall, they can be written as:

$$\vec{U}_P^n = (\vec{U}_P \cdot \vec{n}) \vec{n} \quad (32)$$

$$\vec{U}_P^t = \vec{U}_P - \vec{U}_P^n \quad (33)$$

Since the tangential unit vector is obtained as $\vec{t} = \vec{U}_P^t / |\vec{U}_P^t|$, the velocity at the face n can be obtained as:

$$U_n = \vec{t} \cdot \vec{U}_n \quad (34)$$

Although the distance y_n of a tetrahedral or pyramidal (triangle in 2D) cell is determined as a wall-normal distance from its apex, that of a hexahedral or prismatic (quadrilateral in 2D) cell is obtained by

$$y_n = \text{Vol}/S \quad (35)$$

where Vol and S are the volume and the wall area of the cell, respectively.

When one calculates the velocity components, the wall shear stress $\tau_w \vec{t}$ should be decomposed to each direction.

Examples of applications

Although the AWF has shown its superb performance in many flow fields [8, 17–19], the results of a relatively complicated case are shown here.

Figure 9.4a shows the geometry and the mesh used for an IC Engine port-valve-cylinder flow [18, 19]. The total cell number of 300,000 is applied for a half model. The flow comes through the intake port and the valve section into the cylinder at $\text{Re} \simeq 10^5$. The turbulence model used is the standard $k-\varepsilon$ model.

As shown in Figure 9.4b, it is clear that the standard LWF produces too large k distribution in the region between the valve and the valve-seat circled by broken lines. The AWF, however, reduces such production due to its inclusion of the pressure gradient dependency. This leads to the improved distribution of the gas discharge coefficient C_d by the AWF as shown in Figure 9.5 though it is not very significant. The tumble ratio, which is the momentum ratio of the in-cylinder longitudinal vortex and the inlet flow, at the point of 6.8 mm of the valve lift is also better predicted by the AWF.

9.4.3 AWF for rough wall turbulent flow and heat transfer

In a rough wall turbulent boundary layer, as shown in Figure 9.6, the logarithmic velocity profile shifts downward depending on the equivalent sand grain roughness height h and written as an empirical formula [20]:

$$U^+ = \frac{1}{\kappa} \ln \frac{y^+}{h^+} + 8.5 \quad (36)$$

for completely rough flows. This implies that the flow becomes more turbulent due to the roughness. It is thus considered that the viscous sub-layer is destroyed in the ‘fully rough’ regime while it partly exists in the ‘transitional roughness’ regime.

In common with conventional wall functions (e.g. Cebeci and Bradshaw [21]), this extension of the AWF strategy [10] to flows over rough surfaces involves the use of the dimensionless roughness height. In this case, however, instead of h^+ , h^* is used to modify the near-wall variation of the turbulent viscosity. More specifically, in a rough wall turbulence, y_v^* is no longer fixed at 10.7 and is modelled to become smaller. This provides that the modelled distribution of μ_t shown in Figure 9.2 shifts towards the wall depending on h^* . At a certain value of the dimensionless

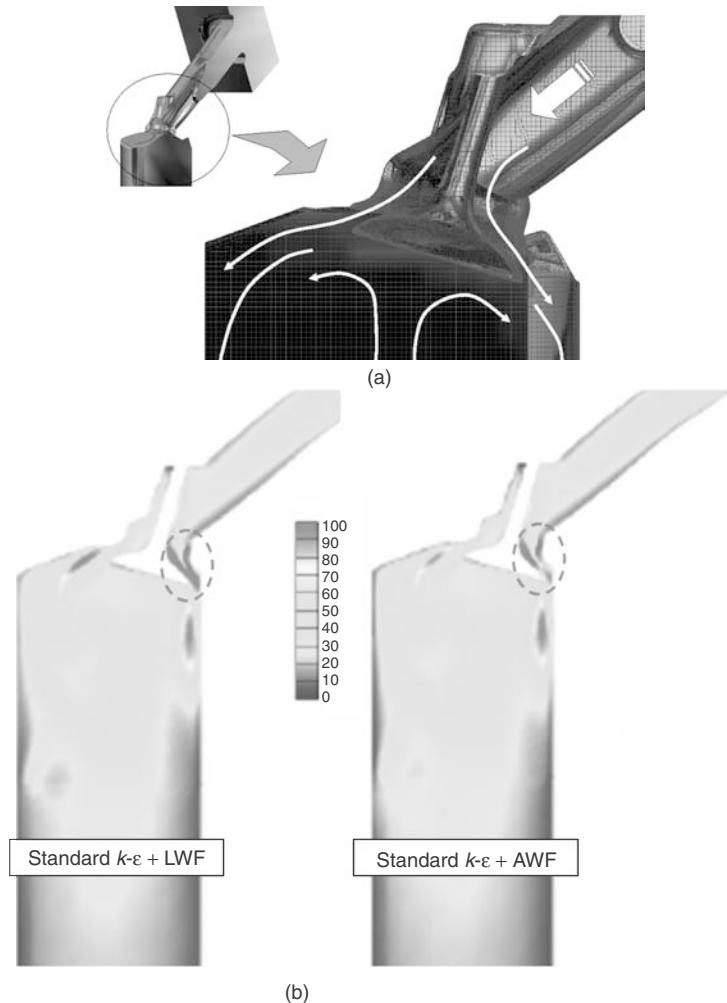


Figure 9.4. Port-valve-cylinder flow: (a) mesh, (b) k distribution.

roughness height $h^* = A$, $y_v^* = 0$ is assumed and at $h^* > A$ it is expediently allowed to have a negative value of y_v^* to give a positive value of μ_t at the wall as:

$$y_v^* = y_{vs}^* \{1 - (h^*/A)^m\} \quad (37)$$

where y_{vs}^* is the viscous sub-layer thickness in the smooth wall case. The optimised values for A and m have been determined through a series of numerical experiments and comparisons with available flow data. The resultant form is:

$$y_v^* = y_{vs}^* \{1 - (h^*/70)^m\} \quad (38)$$

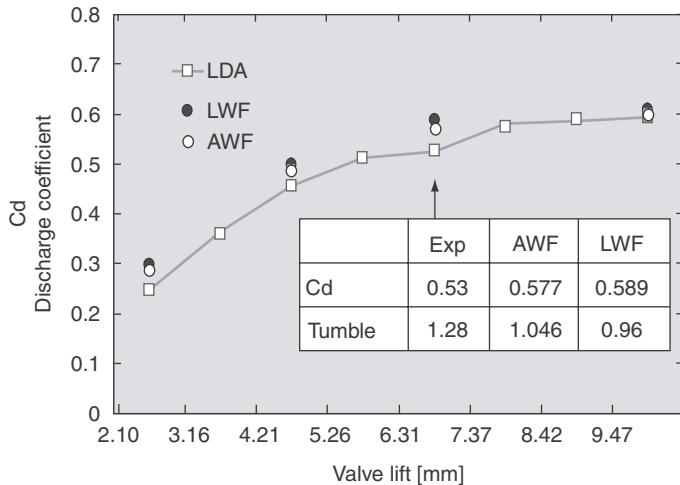


Figure 9.5. Distribution of gas discharge coefficient.

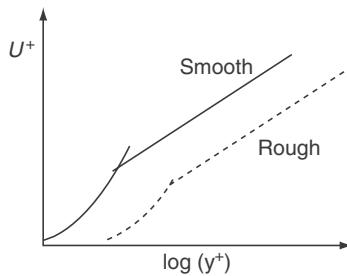


Figure 9.6. Surface roughness effects on velocity profile.

with

$$m = \max \left\{ \left(0.5 - 0.4 \left(\frac{h^*}{70} \right)^{0.7} \right), \left(1 - 0.79 \left(\frac{h^*}{70} \right)^{-0.28} \right) \right\} \quad (39)$$

For $h^* < 70$, the viscosity-dominated sub-layer exists, but with the above modified value for y_v^* . When $h^* > 70$, corresponding to $y_v^* < 0$, it is totally destroyed. Note that although the experimentally known condition for the full rough regime is $h^+ \geq 70$, the condition when the viscous sub-layer vanishes from equation (38) is $h^* = 70$ that corresponds to $h^+ \simeq 40$ in a typical channel flow case. This is rather consistent with the relation of the viscous sub-layer thicknesses in the smooth wall case. (In the smooth wall case, the viscous sub-layer thickness of $y_v^* = 10.7$ corresponds to approximately half the thickness of the conventionally defined viscous sub-layer of $y^+ = 11$.) It is thus considered that a part of the transitional roughness regime is somehow effectively modelled into the region without the viscous sub-layer by the present strategy due to simply assuming the near-wall turbulence behaviour.

Unlike in a sub-layer over a smooth wall, the total shear stress now includes the drag force from the roughness elements in the inner layer which is proportional to the local velocity squared and becomes dominant away from the wall, compared to the viscous force. In fact, the data reported by Krogstad et al. [22] and Tachie et al. [23] showed that the turbulent shear stress away from the wall sometimes became larger than the wall shear stress. This implies that the convective and pressure gradient contributions should be represented somewhat differently across the inner layer, below the roughness element height. Hence, the practice of simply evaluating the rhs of equation (17) in terms of nodal values needs modifying. In the present strategy a simple approach has been taken by assuming that the total shear stress remains constant across the roughness element height. Consequently, one is led to

$$C_U = \begin{cases} 0, & \text{if } y^* \leq h^* \\ \frac{v^2}{k_p} \left[\frac{\partial}{\partial x} (\rho UU) + \frac{\partial P}{\partial x} \right], & \text{if } y^* > h^* \end{cases} \quad (40)$$

In the energy equation, \Pr_t is also no longer constant over the wall-adjacent cell. The reason for this is that since the fluid trapped around the roots of the roughness elements forms a thermal barrier, the turbulent transport of the thermal energy is effectively reduced relative to the momentum transport. (The results of the rib-roughened channel flow direct numerical simulation (DNS) by Nagano et al. [24] support this consideration since their obtained turbulent Prandtl number increases significantly towards the wall in the region between the ripples.) Thus, as illustrated in Figure 9.7, although it might be better to model the distribution of \Pr_t with a non-linear function, a simple linear profile is assumed in the roughness region of $y \leq h$ as:

$$\Pr_t = \Pr_t^\infty + \Delta\Pr_t \quad (41)$$

$$\Delta\Pr_t = C_h \max(0, 1 - y^*/h^*) \quad (42)$$

After a series of numerical experiments referring to the experimental correlation, the following form for C_h has been adopted within the roughness elements ($y \leq h$):

$$C_h = \frac{5.5}{1 + (h^*/70)^{6.5}} + 0.6 \quad (43)$$

Over the rest of the field ($y > h$), $\Pr_t = \Pr_t^\infty$ is applied. (See Table 9.1 for the model coefficients.) Note that since the turbulent viscosity is defined as zero in the region $y < y_v$, the precise profile adopted for the turbulent Prandtl number in the viscous sub-layer ($y < y_v$) does not affect the computation.

At a high \Pr flow ($\Pr \gg 1$), the sub-layer, across which turbulent transport of thermal energy is negligible, becomes thinner than the viscous sub-layer. Thus, the assumption that the turbulent heat flux becomes negligible when $y < y_v$ no longer applies. (See the high \Pr version of the AWF for such flows presented in section 9.4.5.)

The analytical solutions of both mean flow and energy equations then can be obtained in the four different cases illustrated in Figure 9.8 assuming that the wall-adjacent cell height is always greater than the roughness height. Although one can

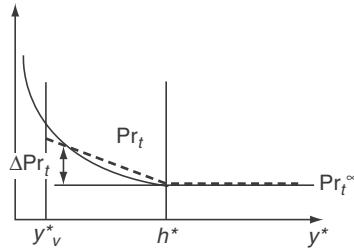
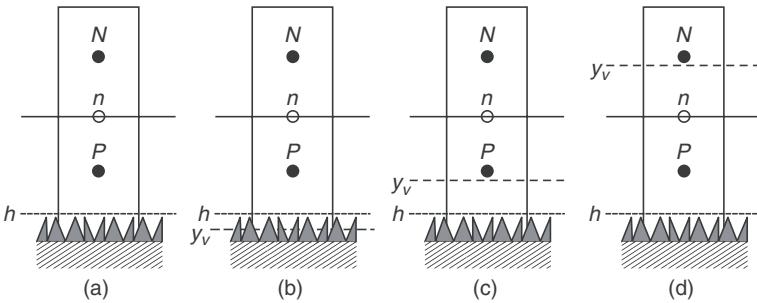


Figure 9.7. Modelled turbulent Prandtl number distribution.

Figure 9.8. Near-wall cells over a rough wall: (a) $y_v \leq 0$, (b) $0 < y_v \leq h$, (c) $h < y_v \leq y_n$, (d) $y_n < y_v$.

apply the models of equations (40) and (41) at any node point, limiting them to the wall-adjacent cells is preferable from the numerical view point since it only requires a list of the cells facing to walls for wall boundary conditions. (Despite that, in simple flow cases, the AWF is still applicable to wall-adjacent cells whose height is lower than the roughness height [10].) Even in the case without any viscous sub-layer, case(a) of Figure 9.8a, the resultant expressions for τ_w and q_w are of the same form as those of equations (27) and (28). However, different values of A_U and A_T , which are functions of the roughness height, are obtained, corresponding to the four different cases. (See Appendix A for the detailed derivation process.)

In Table 9.A1, the cell averaged generation term \bar{P}_k and A_U are listed, introducing $Y^* \equiv 1 + \alpha(y^* - y_v^*)$. Note that equations (30) and (31) are still used for the dissipation, and the integration for \bar{P}_k in Table 9.A1 can be performed as follows:

$$\begin{aligned} \int_a^b (y - y_v) \left(\frac{Cy + A}{1 + \alpha(y - y_v)} \right)^2 dy \\ = \left[\frac{C^2}{2\alpha^2} y^2 + \frac{C(2A + Cy_v - 2C/\alpha)}{\alpha^2} y + \frac{(A + Cy_v - C/\alpha)^2}{\alpha^2 [1 + \alpha(y - y_v)]} \right]_a^b \\ + \frac{(A + Cy_v - C/\alpha)(A + Cy_v - 3C/\alpha)}{\alpha^2} \ln [1 + \alpha(y - y_v)] \Big|_a^b \end{aligned} \quad (44)$$

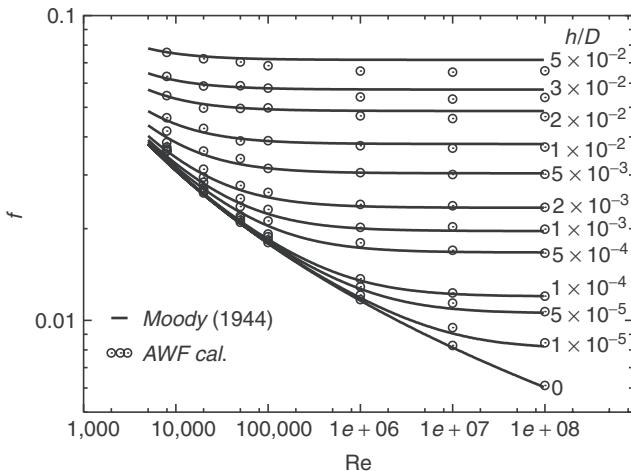


Figure 9.9. Friction factors in pipe flows (Moody chart).

For heat transfer, the resultant form of the integration constant A_T can be written as:

$$A_T = \{\mu(\Theta_n - \Theta_w)/\text{Pr} + C_T E_T\}/D_T \quad (45)$$

where the coefficients D_T and E_T are listed in Table 9.A2, defining $\alpha_T \equiv \alpha\text{Pr}/(\text{Pr}_t^\infty)$, $\beta_T \equiv C_0/(h^*\text{Pr}_t^\infty)$, $Y^{\alpha T} \equiv 1 + \alpha_T(y^* - y_v^*)$, $Y^{\beta T} \equiv 1 + \beta_T(y^* - y_v^*)$, and $\lambda_b \equiv Y_0^{\alpha T} + \beta_T h^*$.

In the case of a constant wall heat flux condition, the wall temperature is obtained by rewriting equations (28) and (45) as:

$$\Theta_w = \Theta_n + \frac{\text{Pr} q_w}{\rho c_p k_p^{1/2}} D_T + \frac{\text{Pr} C_T E_T}{\mu} \quad (46)$$

Examples of applications

The AWF has been implemented with the ‘standard’ linear $k-\varepsilon$ (Launder and Spalding [4], Launder and Sharma: LS [25]) and also with the cubic non-linear $k-\varepsilon$ model of Craft, Launder and Suga: CLS [26]. (Although the LS and the CLS models are LRN models, with the wall-function grids, the LRN parts of the model terms do not contribute to the computation results.) For the turbulent heat flux in the core region, the usual eddy diffusivity model with a prescribed turbulent Prandtl number $\text{Pr}_t = 0.9$ is used.

Pipe flows. Figure 9.9 compares the predicted friction coefficient and the experimental correlation for turbulent pipe flows, known as the Moody chart [27]. The turbulence is modelled by the linear $k-\varepsilon$ model with the AWF. In the range of $h/D = 0$ to 0.05 (D : pipe diameter) and $Re = 8000$ to 10^8 , it is confirmed that the AWF performs reasonably well over a wide range of Reynolds numbers and roughness heights.

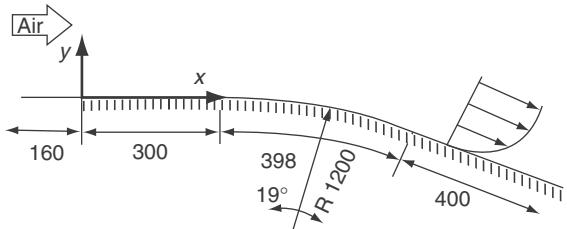


Figure 9.10. Flow geometry of the convex rough wall boundary layers of Turner et al. [28].

Curved wall boundary layer flows. Heat transfer along curved surfaces is very common and important in engineering applications such as in heat sinks and around turbine blades. Thus, although the AWF itself does not explicitly include sensitivity to streamline curvature, it is useful to confirm its performance when applied in combination with a turbulence model which does capture streamline curvature effects. Hence, the turbulence model used here is the cubic non-linear k - ε model (CLS).

For comparison, the rough wall heat transfer experiments over a convex surface by Turner et al. [28] are chosen. The flow geometry is shown in Figure 9.10. The working fluid was air at room temperature and the wall was isothermally heated. The comparison is made in the cases of trapezoidal-shaped roughness elements. According to Turner et al., the equivalent sand grain roughness height h is 1.1 times the element height. The computational mesh used is 90×20 whose wall adjacent cell height is 5 mm, which is greater than the equivalent sand grain roughness heights. (A fine mesh of 90×50 whose wall adjacent cell height is 1 mm is also used to confirm the sensitivity to the wall-adjacent cell heights. The corresponding discussion is addressed in the following paragraph.)

Figure 9.11 compares the heat transfer coefficient α distribution under zero-pressure gradient conditions. Figure 9.11a shows the case of $h = 0.55$ mm. The inlet velocities of $U_0 = 40, 22$ m/s, respectively, correspond to $h^+ \approx 90, 50$ and thus they are in the full and the transitional roughness regimes. In the case of $h = 0.825$ mm, shown in Figure 9.11b, $U_0 = 40, 22$ m/s correspond to $h^+ \approx 135, 80$ which are both in the fully rough regime. From the comparisons, although there can be seen some discrepancy, it is recognised that the agreement between the experiments and the predictions is acceptable in both the full and transitional roughness regimes. In Figure 9.11b, the result by the fine mesh of 90×50 whose wall adjacent cell height is 1 mm is also plotted for the case of $U_0 = 22$ m/s. It is readily seen that the AWF is rather insensitive to the near-wall mesh resolution in such a flow case since two profiles from very different resolutions are nearly the same.

Figure 9.11 also shows the effects of the wall curvature on the heat transfer coefficients in the case of $U_0 = 40$ m/s. (The curved section is in the region of $300 \text{ mm} \leq x \leq 698 \text{ mm}$.) Although the curvature effects observed are not large, since the curvature is not very strong, they are certainly predicted by the computations with the AWF and cubic non-linear k - ε model. Turner et al. reported that the curvature appeared to cause an increase of 2 to 3% in the heat transfer coefficient α .

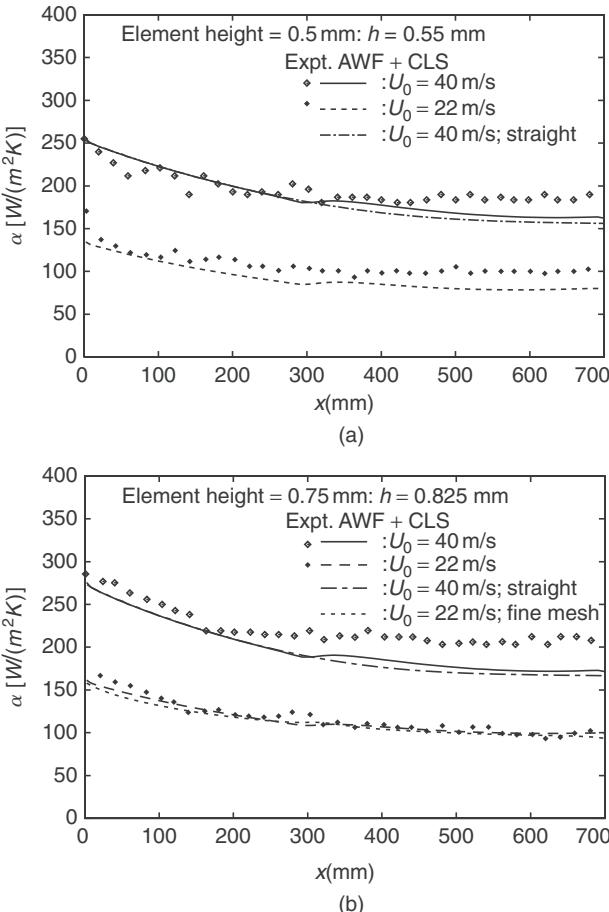


Figure 9.11. Heat transfer coefficient distribution in convex rough wall boundary layers at $\text{Pr} = 0.71$.

Computations are consistent with this experimental observation. Note that the present curvature effects are of the entrance region of the convex wall. Due to the sudden change of the curvature rate, the pressure gradient becomes locally stronger near the starting point of the convex wall resulting in flow acceleration and thus heat transfer enhancement.

Sand dune flow. Figure 9.12 shows the geometry and a computational mesh used for a water flow over a sand dune of van Mierlo and de Ruiter [29]. Their experimental rig consisted of a row of 33 identical 2D sand dunes covered with sand paper, whose averaged sand grain height was 2.5 mm. Since they measured the flow field around one of the central dunes of the row, streamwise periodic boundary conditions are imposed in the computation. In the experiments, the free surface

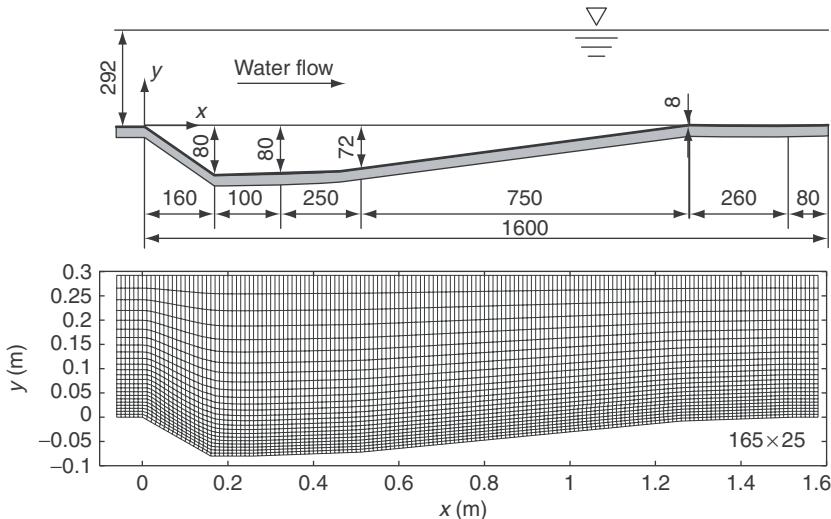


Figure 9.12. Dune profile and computational grid.

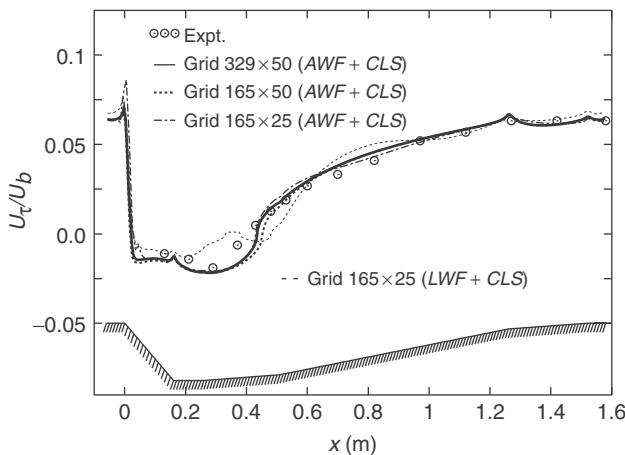


Figure 9.13. Friction velocity over the sand dune.

was located at $y = 292$ mm and the bulk mean velocity was $U_b = 0.633$ m/s. Thus, the Reynolds number based on U_b and the surface height was 175,000. In this flow, a large recirculation zone appears behind the dune. Due to the flow geometry, the separation point is fixed at $x = 0$ m as in a back-step flow.

Figure 9.13 compares the predicted friction velocity distribution on meshes of 330×50 , 165×50 and 165×25 . In the finest and coarsest meshes, the heights of the first cell facing the wall are, respectively, 4 and 8 mm, and larger than the grain height of 2.5 mm which corresponds to $h^+ \simeq 75$ at the inlet. The turbulence model

applied is the cubic CLS $k-\varepsilon$ model with the differential length-scale correction term for the dissipation rate equation [30]. The lines of the predicted profiles are almost identical to one another, proving that the AWF is rather insensitive to the computational mesh. (In the following discussions on flow field quantities, results by the finest mesh of 330×50 are used.)

Figure 9.13 also shows the result by the LWF. Although the LWF, equation (36) should perform reasonably in flat plate boundary layer type of flows, it is obvious that the LWF produces an unstable wiggled profile in the recirculating region of $0 \text{ m} < x < 0.6 \text{ m}$. The computational costs by the AWF and the LWF are almost the same as each other, with the more complex algebraic expressions of the AWF requiring slightly more processing time.

Figure 9.14 compares flow field quantities predicted by the CLS and the LS models with the AWF. In the distribution of the mean velocity and the Reynolds shear stress, both models agree reasonably well with the experiments as shown in Figure 9.14a and b while the CLS model predicts the streamwise normal stress better than the LS model (Figure 9.14c). These predictive trends of the models are consistent with those in separating flows by the original LRN versions and thus it is confirmed that coupling with the AWF preserves the original capabilities of the LRN models.

Ramp flow. Figure 9.15 illustrates the flow geometry and a typical computational mesh (220×40) used for the computations of the channel flows with a ramp on the bottom wall by Song and Eaton [31]. A 2D wind tunnel whose height was 152 mm with a ramp (height $H = 21 \text{ mm}$, length $L = 70 \text{ mm}$, radius $R = 127 \text{ mm}$) was used in their experiments. Air flowed from the left at a free stream velocity $U_e = 20 \text{ m/s}$ with developed turbulence ($\text{Re}_\theta = 3,400$ at $x = 0 \text{ mm}$ for the smooth wall case; $\text{Re}_\theta = 3,900$ for the rough wall case). Since for the rough wall case sand paper with an averaged grain height of 1.2 mm, which corresponds to $h^+ \simeq 100$ at $x = 0 \text{ mm}$, covered the ramp part, the height of the first computational cell from the wall is set as 1.5 mm. (Note that since the location of the separation point is not fixed in this flow case, the grid sensitivity test, which is not shown here, suggests that unlike in the other flow cases large wall adjacent cell heights affect prediction of the recirculation zone.) This flow field includes an adverse pressure gradient along the wall and a recirculating flow whose separation point is not fixed, unlike in the sand dune flow. The measured velocity fields implied that the recirculation region extended between $0.74 \leq x'(:x/L) \leq 1.36$ and $0.74 < x' \leq 1.76$ in the smooth and rough wall cases, respectively. Thus, relatively finer streamwise resolution is applied to the computational mesh around the ramp part.

Figure 9.16 compares the predicted pressure coefficients of the smooth wall case by the AWF and the standard LWF with the results of the LRN computation. The turbulence model used is the CLS model. The first grid nodes from the wall (y_1) are located at $y_1^+ = 15 \sim 150$ for the wall-function models while those for the LRN are located at $y_1^+ \leq 0.2$. (The mesh used for the LRN case has 100 cells in the y direction.) Obviously, the AWF produces profiles that are closer to those of the experiment than the profiles of the LWF model. The AWF reasonably captures the

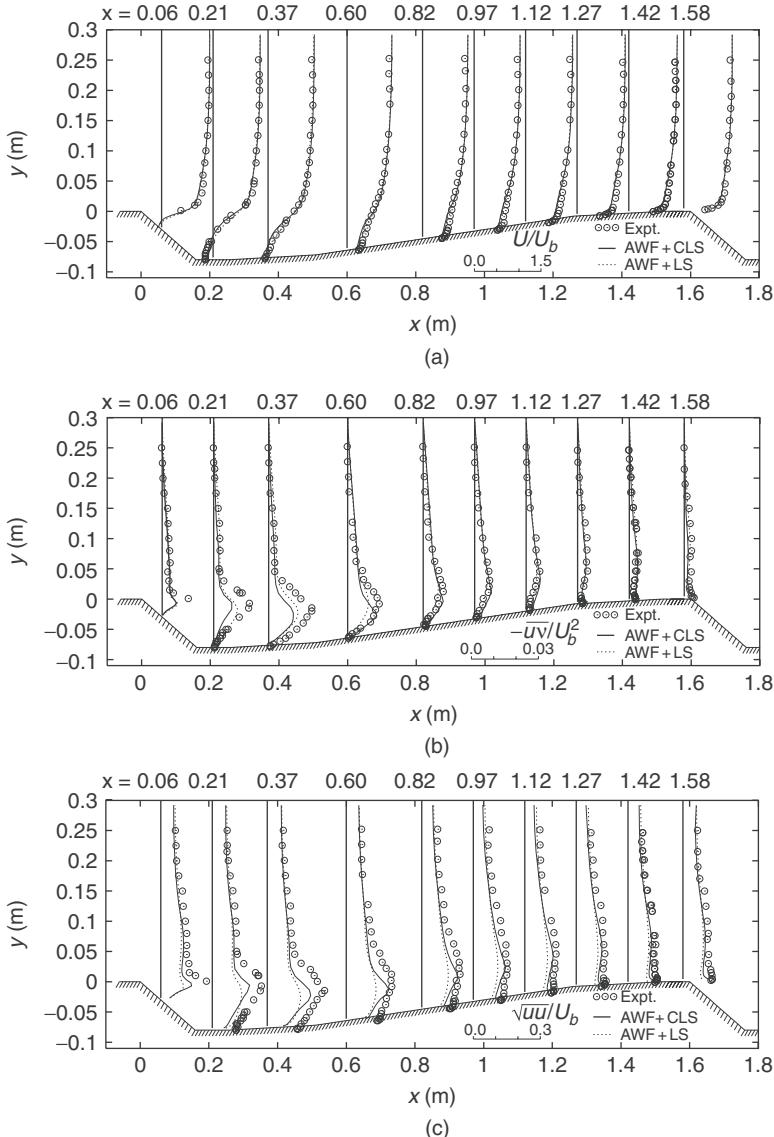


Figure 9.14. Mean velocity and Reynolds stress distribution of the sand dune flow.

effects of adverse pressure gradients due to the inclusion of the sensitivity to pressure gradients in its form. Hence, its predictive tendency is similar to that of the LRN model.

Figure 9.17a compares the mean velocity profiles for both the smooth and rough wall cases. For both the cases, the agreement between the experiments and the predictions by the AWF coupled to the CLS model is reasonably good.

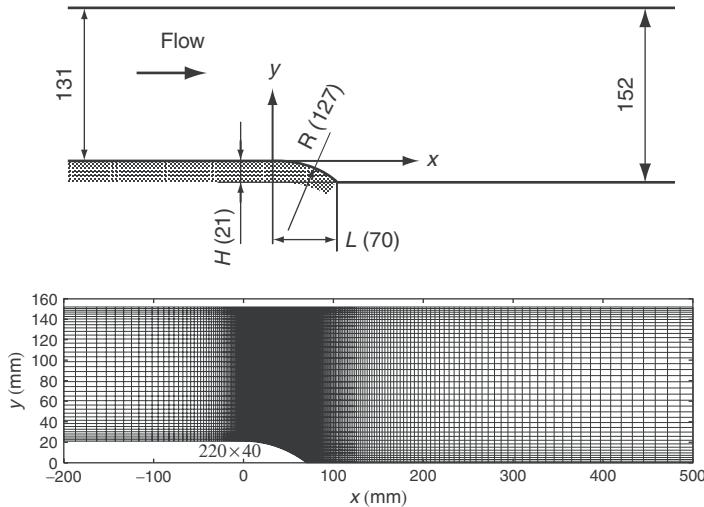


Figure 9.15. Ramp geometry and computational grid.

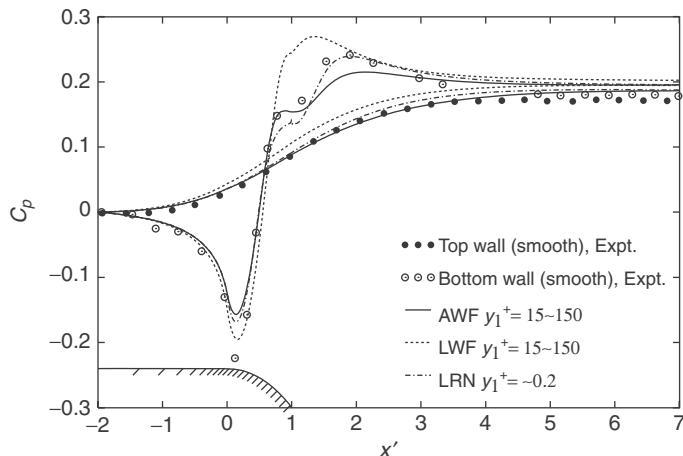


Figure 9.16. Pressure coefficient distribution of the smooth ramp flow; CLS is used in all the computations.

The predicted recirculation zones are $0.69 < x' < 1.40$ and $0.57 < x' < 1.60$ in the smooth and rough wall cases, respectively. They, however, do not correspond well with the experimentally estimated ones. Figure 9.17b-d compare the distribution of the Reynolds stresses at the corresponding locations to those in Figure 9.17a. (The values are normalised by the reference friction velocity $U_{\tau,\text{ref}}$ at $x' = -2.0$.) The agreement between the prediction and the experiments is again reasonably good for each quantity.

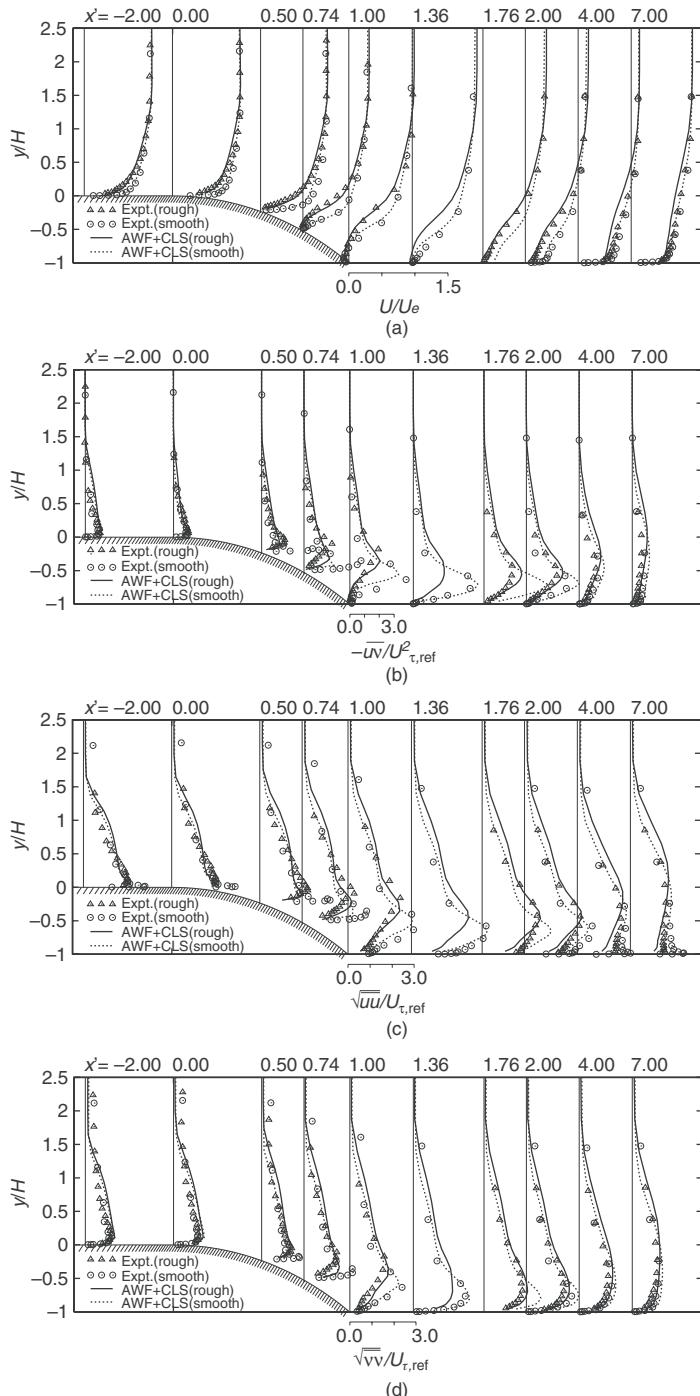


Figure 9.17. Mean velocity and Reynolds stress distribution of the ramp flow.

9.4.4 AWF for permeable walls

Flows over permeable walls can be seen in many industrially important devices such as catalytic converters, metal foam heat exchangers and separators of fuel cells. Some geophysical fluid flows, i.e. water flows over river beds etc., are often concerned as flows over permeable surfaces as well.

The effects of a porous medium on the flow in the interface region between the porous wall and clear fluid (named ‘interface region’ hereafter) have been focused on by many researchers. Due to the severer complexity of the flow phenomena, however, experimental turbulent flow studies in the interface regions are rather limited. Zippe and Graf [32] and others experimentally found that the friction factors of turbulent flows over permeable beds became higher than those over impermeable walls with the same surface roughness. The recent DNS study of the interface region by Breugem et al. [33] also indicated that the friction velocity became higher associated with the increase of the permeability. All these results imply that the wall permeability effects on turbulence should be solely taken into account for flow simulations.

This part, thus, introduces an attempt [11] to construct an AWF which includes the effects of the wall permeability as well as the wall roughness for computing the interface regions over porous media.

Modelling for porous/fluid interfacial turbulence

It is natural to consider a slip tangential velocity U_w at the interface between a clear fluid flow and a porous surface when one introduces a volume averaging concept. Before discussing the slip velocity, it is essential to define a nominal location of the interface. In some porous media, their outermost perimeter may be reasonable as the interface [34]. This means when the pores are filled with solid material, a smooth impermeable boundary is recovered. However, in case that a porous medium is composed of beads, the recovered impermeable surface has some roughness by defining the pores as the spaces surrounded by the beads. This definition is consistent with those in the experiments by Zippe and Graf [32] and

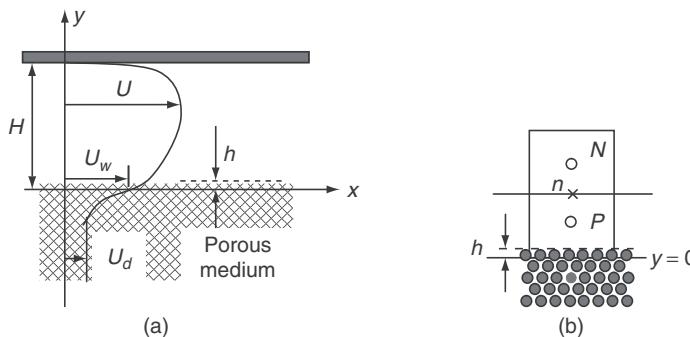


Figure 9.18. (a) Velocity profile of a channel flow over a porous medium, (b) near-wall cell.

others. Therefore, the surface roughness h of porous/fluid interface is considered as in Figure 9.18a and that the interface is located at the bottom of the roughness elements. In order to obtain the tangential velocity U_w at the interface ($y=0$), the velocity distribution in a porous medium is estimated as follows.

Following Whitaker [35], the volume averaged Navier-Stokes equations for incompressible flows in isotropic porous media using the decomposition:

$$u_i = \langle u_i \rangle^f + \tilde{u}_i \quad (47)$$

are:

$$\frac{\partial \langle u_i \rangle^f}{\partial t} + \langle u_j \rangle^f \frac{\partial \langle u_i \rangle^f}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \langle p \rangle^f}{\partial x_i} + \nu \frac{\partial^2 \langle u_i \rangle^f}{\partial x_j^2} - \frac{\partial \tau_{ij}}{\partial x_j} + f_i \quad (48)$$

$$\frac{\partial \langle u_i \rangle}{\partial x_i} = 0 \quad (49)$$

where p , ρ , ν , and φ are, respectively, the pressure, the density, the kinematic viscosity of the fluid and the porosity of the porous medium. The volume averaged values $\langle \phi \rangle$ and $\langle \phi \rangle^f$ are *superficial* and *intrinsic* averaged values of a variable ϕ , respectively. (The superficial averaging is defined by taking average of a variable over a volume element of the medium consisting of both solid and fluid materials, while the intrinsic averaging is defined over a volume consisting of fluid only.) Between them the following relation exists:

$$\langle \phi \rangle = \varphi \langle \phi \rangle^f \quad (50)$$

Although the sub-filter-scale dispersion $\partial \tau_{ij} / \partial x_j$ needs a closure model as in large eddy simulations, it could be neglected in porous media since it is small enough compared with the other terms. As in Whitaker [35], the drag force f_i in isotropic porous media can be parametrised as:

$$f_i \simeq -\nu \frac{\langle u_j \rangle}{K_{ij}} - \nu \frac{F_{ij} \langle u_k \rangle}{K_{jk}} \quad (51)$$

where K_{ij} and F_{ij} are, respectively, the permeability tensor and the Forchheimer correction tensor. These tensors are empirically given by Macdonald et al. [37] being based on the modified Ergun equation [36] as:

$$K_{ij} = \underbrace{\frac{d_p^2 \varphi^3}{180(1-\varphi)^2}}_K \delta_{ij} \quad (52)$$

$$F_{ij} = \underbrace{\frac{\varphi}{100(1-\varphi)} \frac{d_p}{\nu}}_F \sqrt{(\langle u_k \rangle^f)^2} \delta_{ij} \quad (53)$$

where d_p is the mean particle diameter.

Thus, by the Reynolds averaging of the volume averaged momentum equations one can obtain the following equations for a developed turbulent flow in a homogeneous porous medium:

$$0 = -\frac{1}{\rho} \frac{\partial P}{\partial x} - \frac{\partial \bar{u}\bar{v}}{\partial y} + \frac{\nu}{\varphi} \frac{\partial^2 U}{\partial y^2} - \frac{\nu}{K} U - \frac{\nu\varphi F}{K} \overline{q(u)^f} \quad (54)$$

$$0 = -\frac{1}{\rho} \frac{\partial P}{\partial y} - \frac{\partial \bar{v}^2}{\partial y} - \frac{\nu\varphi F}{K} \overline{q(v)^f} \quad (55)$$

where $q = \sqrt{\langle u_k \rangle^f \langle u_k \rangle^f}$, $\bar{u}_i \bar{u}_j = \overline{(\langle u_i \rangle^f - \langle u_i \rangle^f)(\langle u_j \rangle^f - \langle u_j \rangle^f)}$, $U = \overline{\langle u \rangle}$, and $P = \overline{\langle p \rangle^f}$.

In laminar flow cases where the permeability Reynolds number $\text{Re}_K \equiv \sqrt{K}U_\tau/\nu$, which is based on the permeability K and the friction velocity U_τ , is sufficiently small, the Reynolds stress and the Forchheimer drag terms can be neglected:

$$0 = -\frac{1}{\rho} \frac{\partial P}{\partial x} + \frac{\nu}{\varphi} \frac{\partial^2 U}{\partial y^2} - \frac{\nu}{K} U \quad (56)$$

$$0 = -\frac{1}{\rho} \frac{\partial P}{\partial y} \quad (57)$$

The solution of the above equations (56) and (57), namely, the Brinkman equations [38] is a decaying exponential function which meets the interfacial (slip) velocity U_w at $y = 0$ and the Darcy velocity U_d deep inside the porous medium.

$$U = U_d + (U_w - U_d) \exp \left(y \sqrt{\frac{\varphi}{K}} \right) \quad (58)$$

The Darcy velocity is:

$$U_d = -\frac{K}{\mu} \frac{\partial P}{\partial x} \quad (59)$$

In the case of high Re_K , turbulent diffusion and Forchheimer drag terms in the momentum equations should be considered and thus their general analytical solution cannot be easily obtained. Breugem et al. [33] found that a good approximation of the turbulent flow inside the porous media can be obtained by the following exponential formula:

$$U \simeq U_d + (U_w - U_d) \exp \left(\alpha_p y \sqrt{\frac{\varphi}{K}} \right) \quad (60)$$

where α_p is an empirical coefficient. If Re_K is small enough, $\alpha_p = 1$. In very large Re_K cases, where the turbulent diffusion and the Forchheimer drag terms should

balance each other out:

$$-\frac{\partial \bar{u}\bar{v}}{\partial y} \simeq \frac{\nu\varphi F}{K} q\langle u \rangle^f \quad (61)$$

The Reynolds stress estimated with the eddy viscosity ($\nu_t \simeq u_t y$; u_t is an appropriate velocity scale) and the velocity gradient from equation (60) leads to

$$-\bar{u}\bar{v} = \nu_t \frac{1}{\varphi} \frac{\partial U}{\partial y} \simeq u_t y \frac{U}{\varphi} \alpha_p \sqrt{\frac{\varphi}{K}} \simeq \alpha_p u_t^2 y \sqrt{\frac{\varphi}{K}} \quad (62)$$

then the LHS of equation (61) may be rewritten as

$$-\frac{\partial \bar{u}\bar{v}}{\partial y} \simeq \alpha_p \sqrt{\frac{\varphi}{K}} u_t^2 \quad (63)$$

Thus, α_p can be estimated as

$$\alpha_p \sqrt{\frac{\varphi}{K}} u_t^2 \simeq \frac{\nu\varphi F}{K} u_t^2 \Rightarrow \alpha_p \simeq \nu \sqrt{\frac{\varphi}{K}} F \quad (64)$$

Therefore, the following form can be a model for α_p :

$$\alpha_p = f_{\alpha_p} + (1 - f_{\alpha_p}) \nu \sqrt{\frac{\varphi}{K}} F \quad (65)$$

$$f_{\alpha_p} = \exp\left(-\frac{\text{Re}_K}{\beta_p}\right) \quad (66)$$

bridging between 1 and $\nu \sqrt{\varphi/K} F$ depending on the permeability. By referring to the distribution of the coefficient α_p obtained from the DNS [33], the model coefficient β_p can be functionalised as

$$\beta_p = \frac{90}{\text{Re}_K^{0.68}} \quad (67)$$

Hence

$$f_{\alpha_p} = \exp\left(-\frac{\text{Re}_K^{1.68}}{90}\right) \quad (68)$$

(Note that the above form is rewritten in a different way for the AWF as in equation (73).)

Consequently, the slip velocity U_w can be expressed as

$$U_w = U_d + \frac{\partial U / \partial y|_{y=0}}{\alpha_p \sqrt{\varphi/K}} \quad (69)$$

with the velocity gradient $\partial U / \partial y|_{y=0}$ given by the AWF.

In the AWF, the wall shear stress is obtained by the analytical solution of a simplified near-wall version of the transport equation for the wall-parallel momentum. The resultant shear stress form here is the same as that for the rough wall turbulence but with the slip velocity U_w in the integration constant A_U .

Since the wall permeability also makes the flow more turbulent, y_v^* should have its dependency. Equation (38) is thus further modified as

$$y_v^* = y_{vs}^* \left\{ 1 - (h^*/70)^m - f_K \right\} \quad (70)$$

where f_K is a function of the permeability. By referring to the experiments and the DNS data [32, 33], the range of y_v^* which depends on the wall permeability is estimated. Considering the roughness effects in y_v^* , the optimised function in equation (70) for the permeability effects is

$$f_K = 3.7 \left[1 - \exp \left\{ - \left(\frac{K^*}{6} \right)^2 \right\} \right] \quad (71)$$

where $K^* = \sqrt{Kk_p}/\nu$. In this form, K^* is used rather than Re_K since the AWF uses $\sqrt{k_p}$ for the velocity scale. In order to keep consistency with such normalisation, Equation (68) is rewritten as

$$f_{\alpha_p} \simeq \exp \left(- \frac{K^{*1.68}}{250} \right) \quad (72)$$

using the relation of $\text{Re}_K \simeq c_\mu^{1/4} K^*$. The following modified form, however, is employed after further tuning of the model function

$$f_{\alpha_p} = \exp \left(- \frac{K^{*1.76}}{180} \right) \quad (73)$$

Examples of applications

Permeable-wall channel flows. In order to confirm the performance of the method, computations of fully developed turbulent flows in a plane channel with a permeable bottom wall at $y/H = 0$ and a solid top wall at $y/H = 1$ (see Figure 9.18a) are shown here. The considered flow conditions are the same as those in the DNS [33]. The mean diameter of the particle composing the permeable beds is $d_p/H = 0.01$ and their porosity are $\varphi = 0.95, 0.80, 0.60, 0.00$. The permeability and the Forchheimer term are then obtained by equations (52) and (53). The bulk flow Reynolds number is $\text{Re}_b = 5,500$ in all the cases. The roughness height is considered to be $h = d_p$ in this practice. The grid cell number used is only 10 in the wall normal direction due to the use of the wall-function approach for such relatively low Reynolds number flows. The turbulence models used are the standard $k-\varepsilon$ model [4] and the two-component limit (TCL) second moment closure of Craft and Launder [39].

Figure 9.19a compares the mean velocity distribution inside the porous walls. The obtained velocity profiles are from equation (60). Whilst there can be seen

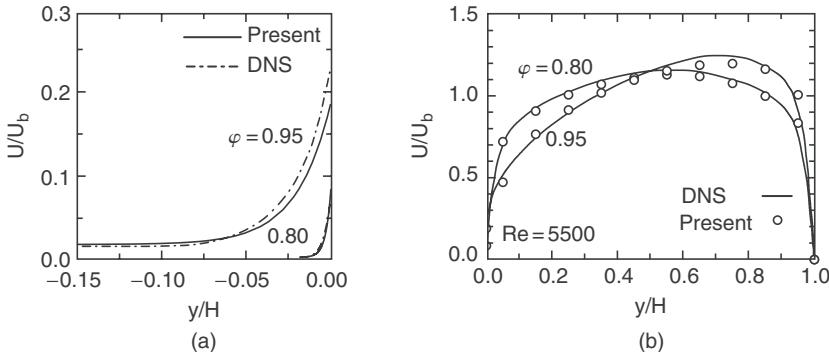


Figure 9.19. Mean velocity distribution: (a) inside porous walls, (b) channel region.

a margin to be improved in the comparison between the present and the DNS results in the case of $\varphi = 0.95$, the agreement is virtually perfect in the case of $\varphi = 0.80$.

The mean velocity distribution in the channel region is compared in Figure 9.19b. It is obvious that the computation well reproduces the characteristic velocity profiles near the permeable walls.

The AWF is also applicable to the second moment closures. Figure 9.20a–c compares the obtained Reynolds stress distribution by the second moment closure [39]. Due to the use of the AWF, it seems hard to capture the near wall peaks of the Reynolds normal stresses as in Figure 9.20a and b. However, general tendencies of the distribution profiles are satisfactorily reproduced by the present scheme.

Turbulent permeable boundary layer flows. The other test cases are turbulent boundary layer flows over permeable walls. The same flow conditions of the experiments of Zippe and Graf [32] are imposed in the computations. The free stream turbulence is $\sqrt{\bar{u}^2}/U_0 = 0.35 \times 10^{-2}$. The Reynolds number based on the momentum thickness ranges $Re_\theta = 1.3 - 2.1 \times 10^4$. Although it was not reported in Ref. [32], the permeability can be estimated by equation (52) as $K = 2.54 \times 10^{-9} \text{ m}^2$ since the experimentally used mean diameter of the ellipsoidal beads composing the permeable bed was $d_p = 2.883 \text{ mm}$. The presently applied porosity is $\varphi \simeq 0.30$ considering the effects of non-spherical shapes of beads used in the experiments. (The porosity of a fully packed body-centred cubic structure by spherical beads is $\varphi = 0.26$. Zippe and Graf did not report about the porosity as well.) The grid cell number used is 120×150 for the developing boundary layer computations. The turbulence model used is the standard k - ε model of Launder and Spalding [4]. Since Zippe and Graf [32] also did not report the roughness height of their test cases, it is presently estimated as $h^+ \simeq 116$ comparing the experimental velocity profile with the formula of Nikuradse [20]: Equation (36).

Figure 9.21 compares the mean velocity profiles where their boundary layer thicknesses correspond to each other. The cases are R (non-permeable rough

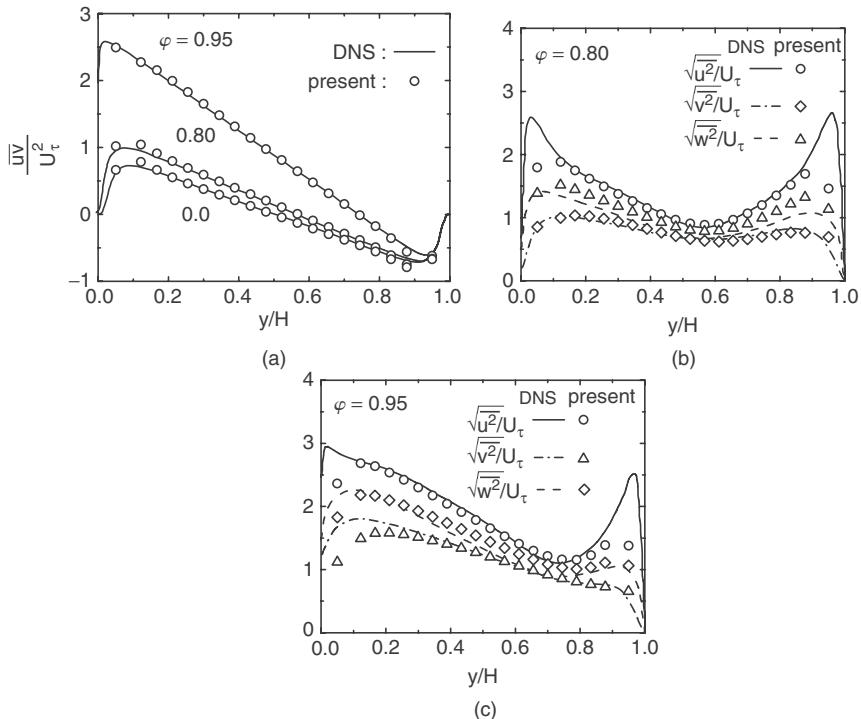


Figure 9.20. Reynolds stress distribution in permeable channel flows.

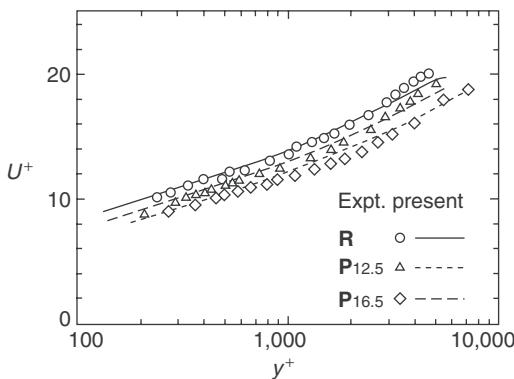


Figure 9.21. Mean velocity distribution in permeable boundary layers.

wall boundary layer), $P_{12.5}$ (permeable wall boundary layer with $U_0 = 12.5$ m/s) and $P_{16.5}$ (permeable wall boundary layer with $U_0 = 16.5$ m/s). Obviously, the present scheme reasonably reproduces the experimentally obtained velocity profiles with/without the permeability.

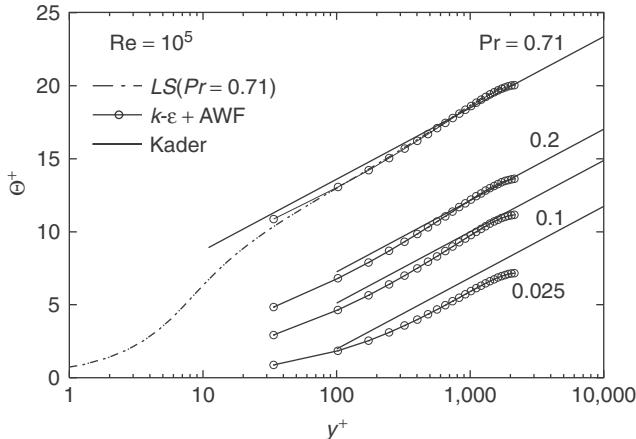


Figure 9.22. Mean temperature profiles in turbulent smooth channel flows at $\text{Pr} < 1$.

9.4.5 AWF for high Prandtl number flows

If one considers to predict turbulent wall heat transfer of high Prandtl number fluid flows such as cooling oil and IC engine water-jacket flows, it is essential to analyse the thermal boundary layer which is much thinner than that of the flow boundary layer. Thus, near-wall modelling which resolves the viscous sub-layer has been thought to be essential for high Pr thermal fields. For example, at the development of a novel turbulent heat flux model applicable to general Pr cases, Rogers et al. [40] supposed correct near-wall stress distribution and Suga and Abe [41] employed an LRN nonlinear $k-\varepsilon$ model. So and Sommer [42] also applied an LRN $k-\varepsilon$ as well as a near-wall stress transport model for flows at $\text{Pr} = 1000$.

However, even with the recent development of LRN heat transfer models, wall-function approaches still attract industrial engineers. Its reasons are a high computational cost of the LRN computation and difficulty of generating quality near-wall grids for complex 3D flow fields such as an IC engine water jacket.

Although the AWF performs reasonably well at $\text{Pr} < 1$ as shown in Figure 9.22,⁴ its applicability to higher Pr cases has not been discussed well so far. Therefore, this part focuses on the improvement of the thermal AWF for high Pr turbulent flows with and without wall roughness [12].

High Pr AWF for smooth wall heat transfer

In the AWF for smooth wall heat transfer, μ_t variation is assumed that μ_t is zero for $y^* \leq y_v^* = 10.7$ and then increases linearly as of equation (16). Since the theoretical

⁴ In the cases shown in Figure 9.22, the constant $\text{Pr}_t = 0.9$ is used for convenience. However, for lower Pr cases, it is well known that higher values of Pr_t should be used [43]. There is thus a tendency for the AWF to underpredict the mean temperature profile, particularly, at $\text{Pr} = 0.025$.

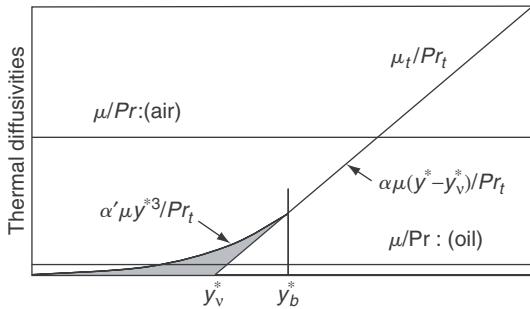


Figure 9.23. Near-wall thermal diffusivity distribution.

wall-limiting variation of μ_t is proportional to y^3 , this assumption does not count a certain amount of turbulent viscosity in the viscous sub-layer. Despite that, its effect is not serious for flow field prediction since the contribution from the molecular viscosity is more significant in the sub-layer. This is also true for the thermal field prediction of fluids whose Pr is less than 1.0. However, in high Pr fluid flows such as oil flows, since the effect of the molecular thermal diffusivity (μ/Pr) becomes very small as illustrated in Figure 9.23, it is then necessary to consider the contribution from the turbulent thermal diffusivity inside the sub-layer. (Note that a prescribed constant turbulent Prandtl number Pr_t is assumed in Figure 9.23.)

In order to compensate the thermal diffusivity inside the sub-layer, Gerasimov [17] introduced an *ad hoc* effective molecular Prandtl number as:

$$\text{Pr}_{\text{eff}} = \frac{\text{Pr}}{1 + 0.017\text{Pr}(1 + 2.9|F_{\varepsilon} - 1|)^{1.5}} \quad (74)$$

where F_{ε} is a model function. This effective Pr approach was tailored for water flows. Thus, its performance in oil flows whose Pr is over 100 is not guaranteed.

In order to improve the μ_t profile inside the sub-layer, it is assumed that the profile of Equation (16) is connected to a function $\alpha'y^*^3$ at the point y_b^* , as illustrated in Figure 9.23.

$$\mu_t/\mu = \begin{cases} \alpha'y^*^3, & \text{for } 0 \leq y^* \leq y_b^* \\ \alpha(y^* - y_v^*), & \text{for } y_b^* \leq y^* \end{cases} \quad (75)$$

Thus,

$$\Gamma_{\theta} = \begin{cases} 1 + \frac{\alpha'\text{Pr}y^*^3}{\text{Pr}_t} = \Gamma_{\theta a}, & \text{for } 0 \leq y^* \leq y_b^* \\ 1 + \frac{\alpha\text{Pr}(y^* - y_v^*)}{\text{Pr}_t} = \Gamma_{\theta b}, & \text{for } y_b^* \leq y^* \end{cases} \quad (76)$$

where $\mu\Gamma_{\theta}/\text{Pr}$ is the total thermal diffusivity. By referring to the near-wall profile of μ_t in a DNS dataset, the value of y_b^* is optimised as $y_b^* = 11.7$ and thus α' is

obtainable as:

$$\alpha' = \frac{\alpha(y_b^* - y_v^*)}{y_b^{*3}} = \frac{\alpha}{y_b^{*3}} \quad (77)$$

Using equation (76), integration in equation (18) can be made. Although the modification of the model is very simple, it makes the analytical integration a little cumbersome as below.

When $y_b \leq y_n$, with $P_2 = 1/\Gamma_{\theta a}$, $P'_2 = 1/\Gamma_{\theta b}$, $y_0 = 0$, $y_1 = y_b$ and $y_2 = y_n$, the coefficients D_T and E_T of equation (45) and (46) are:

$$D_T = S_2(y_1) - S_2(y_0) + \{S'_2(y_2) - S'_2(y_1)\} \frac{P_2(y_1)}{P'_2(y_1)} \quad (78)$$

$$E_T = S_1(y_0) - S_1(y_1) + S'_1(y_1) - S'_1(y_2) \\ + \{S'_2(y_1) - S'_2(y_2)\} \frac{P_1(y_1) - P'_1(y_1)}{P'_2(y_1)} \quad (79)$$

where $P_1 = y^*P_2$, $P'_1 = y^*P'_2$, $S_i = \int P_i dy^*$.

In the case of $y_n < y_b$, with $y_0 = 0$, $y_1 = y_n$, they are:

$$D_T = S_2(y_1) - S_2(y_0) \quad (80)$$

$$E_T = S_1(y_0) - S_1(y_1) \quad (81)$$

(See Appendix B for the results of the integration of $1/\Gamma_{\theta a}$ etc.)

High Pr AWF for rough wall heat transfer

Since the AWF heat transfer model presented so far was only validated in air flows, the coefficient C_h needs recalibration in high Pr flows and the obtained polynomial form is:

$$C_h = \max(0, C_3 \text{Pr}^3 + C_2 \text{Pr}^2 + C_1 \text{Pr} + C_0) \quad (82)$$

$$C_3 = -0.48/h^* + 0.0013, \quad C_2 = 9.90/h^* - 0.0291$$

$$C_1 = -72.35/h^* + 0.3067, \quad C_0 = 98.98/h^* + 0.2103$$

Since the rough wall AWF modifies y_v^* of Equation (38) as:

$$y_v^* = y_{vs}^* \left\{ 1 - \left(\frac{h^*}{70} \right)^m \right\} = y_{vs}^* - \delta_v \quad (83)$$

the turbulent viscosity form of equation (75) changes to:

$$\mu_t/\mu = \begin{cases} \alpha'(y^* + \delta_v)^3, & \text{for } y^* \leq y_b^* \\ \alpha(y^* - y_v^*), & \text{for } y_b^* < y^* \end{cases} \quad (84)$$

Then, the thermal diffusivity has the following forms:

$$\Gamma_\theta = \begin{cases} 1 + \frac{\alpha' \text{Pr}(y^* + \delta_v)^3}{\text{Pr}_t^\infty + C_h \max(0, 1 - y^*/h^*)} = \Gamma_{\theta c}, & \text{for } y^* < y_b^* \\ 1 + \frac{\alpha \text{Pr}(y^* - y_v^*)}{\text{Pr}_t^\infty + C_h \max(0, 1 - y^*/h^*)} = \Gamma_{\theta d}, & \text{for } y_b^* \leq y^* \end{cases} \quad (85)$$

The analytical solutions of energy equations then can be obtained in the four different cases illustrated in Figure 9.8. The resultant expressions for q_w and A_T are of the same form as those presented so far. For cases (a) and (d) of Figure 9.8, D_T and E_T have the forms of Equations (78) and (79) with some changes. For case (a), they are $P_2 = P'_2 = 1/\Gamma_{\theta d}$, $y_0 = 0$, $y_1 = h$ and $y_2 = y_n$. For case (d), they are $P_2 = P'_2 = 1/\Gamma_{\theta c}$, $y_0 = 0$, $y_1 = h$ and $y_2 = y_n$.

In cases (b) and (c), D_T and E_T have the following forms:

$$D_T = S_2(y_1) - S_2(y_0) + \{S'_2(y_2) - S'_2(y_1)\} \frac{P_2(y_1)}{P'_2(y_1)} + \{S''_2(y_3) \\ - S'_2(y_2)\} \frac{P_2(y_1)P'_2(y_2)}{P'_2(y_1)P''_2(y_2)} \quad (86)$$

$$E_T = S_1(y_0) - S_1(y_1) + S'_1(y_1) - S'_1(y_2) + \{S'_2(y_1) - S'_2(y_2)\} \\ \times \frac{P_1(y_1) - P'_1(y_1)}{P'_1(y_1)} + S''_1(y_2) - S''_1(y_3) + \{S''_2(y_2) - S''_2(y_3)\} \\ \times \left(\frac{P_1(y_1) - P'_1(y_1)}{P'_1(y_1)} \cdot \frac{P'_2(y_2)}{P''_2(y_2)} + \frac{P'_1(y_2) - P''_1(y_2)}{P''_2(y_2)} \right) \quad (87)$$

For case (b), $P_2 = 1/\Gamma_{\theta c}$, $P'_2 = P''_2 = 1/\Gamma_{\theta d}$, $y_0 = 0$, $y_1 = y_b$, $y_2 = h$, and $y_3 = y_n$. For case (c), $P_2 = P'_2 = 1/\Gamma_{\theta c}$, $P''_2 = 1/\Gamma_{\theta d}$, $y_0 = 0$, $y_1 = h$, $y_2 = y_b$, and $y_3 = y_n$. (See Appendix B for the results of the integration of $1/\Gamma_{\theta c}$ etc.)

Examples of applications

Smooth wall heat transfer. In order to confirm the effects of the corrected turbulent viscosity on the flow fields, Figure 9.24 compares the mean velocity profiles in turbulent channel flows at the bulk Reynolds number, $\text{Re} = 10^5$. (The standard high Reynolds number $k-\varepsilon$ model [4] and the eddy diffusivity model with $\text{Pr}_t = 0.9$ are used for the computation of the core fields of the present computations.) Although the result by the μ_t correction almost perfectly lies on the log-law line and there can be seen a slight discrepancy between the results with and without the correction, both the results well accord with the LRN LS model and the log-law profiles. (The meshes used for the AWF and the LRN computations have, respectively, 50 and 100 node points in the wall normal direction. Their first cell heights are $y^+ \simeq 30$ and $y^+ \simeq 0.2$, respectively.) This confirms that the correction in the momentum equation may not be totally necessary for engineering flow field computations, and thus the present scheme does not employ the correction for the flow field AWF. This

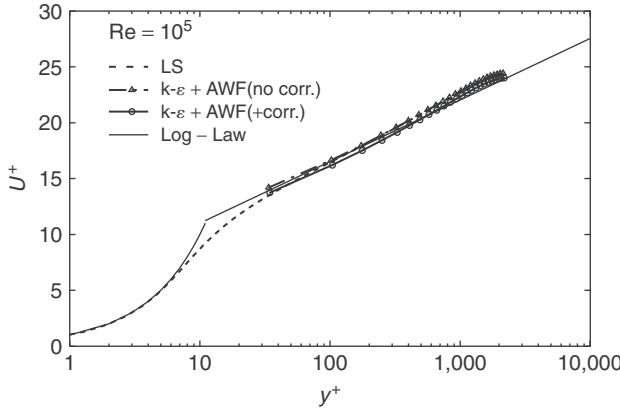


Figure 9.24. Mean velocity profiles in turbulent smooth channel flows.

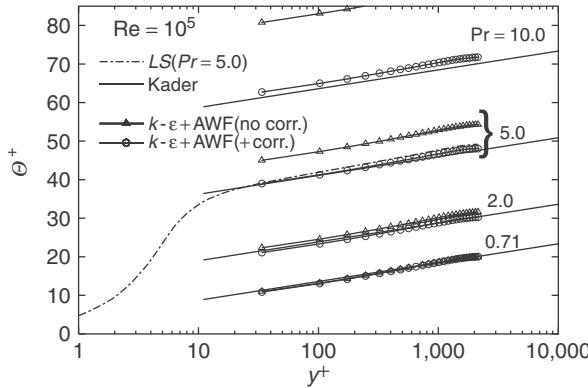


Figure 9.25. Mean temperature profiles in turbulent smooth channel flows.

means that the correction of μ_t is made only in the energy equation in the present strategy.

Figure 9.25 clearly indicates that without the correction, the AWF does not properly reproduce the logarithmic temperature profiles in high Pr flows of $\text{Pr} \geq 5.0$. Note that the experimentally suggested logarithmic distribution by Kader [44] for a wide range of Pr is:

$$\Theta^+ = 2.12 \ln(y^+ \text{Pr}) + (3.85 \text{Pr}^{1/3} - 1.3)^2 \quad (88)$$

In the case of $\text{Pr} = 0.71$, the profiles of the AWF with and without the correction are virtually identical and confirm that the near-wall correction of μ_t is effective for flows at $\text{Pr} > 1.0$.

As shown in Figure 9.26a, the corrected AWF proves its good performance in the range of $50 \leq \text{Pr} \leq 10^3$. However, both the LRN LS model and the AWF with

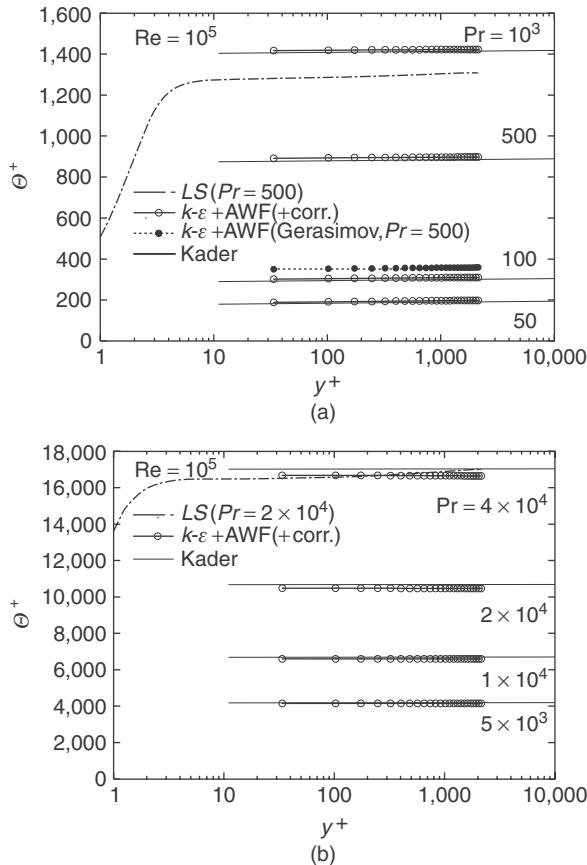


Figure 9.26. Mean temperature profiles in turbulent smooth channel flows at higher Pr .

Gerasimov's [17] effective molecular Prandtl number scheme fail to predict the thermal field at $\text{Pr} = 500$. The former predicts the temperature too high and the latter does too low. Figure 9.26b also confirms that the corrected AWF performs well up to $\text{Pr} = 4 \times 10^4$ though the LRN LS model predicts the thermal field too high. Note that the same grid resolution as that for $\text{Pr} = 5.0$ is used in the LRN computations. This reasonably implies that the grid resolution used is too coarse and a much finer grid is needed for such a high Pr computations by the LRN models. Obviously, it highlights the merit of using the AWF which does not require a finer grid resolution for a higher Pr flow.

Rough wall heat transfer. Figure 9.27 compares the predicted temperature fields of turbulent rough channel flows of $h/D = 0.01$ and 0.03 , (D is channel height). In the cases of $h/D = 0.01$, the corresponding roughness Reynolds number is $h^+ \simeq 60$ which is in the transitional roughness regime, while $h/D = 0.03$ corresponds to $h^+ \simeq 220$ which is well in the fully rough regime. For each roughness case,

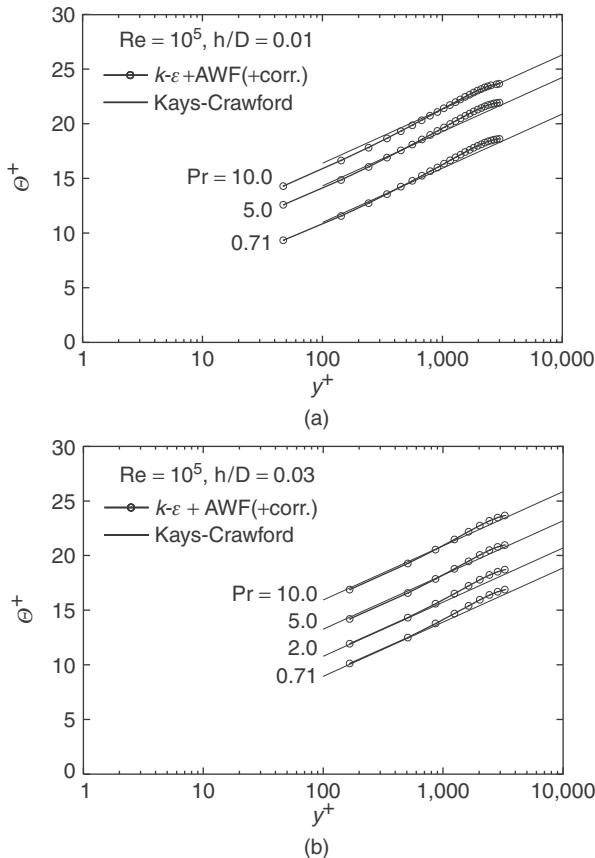


Figure 9.27. Mean temperature profiles in turbulent rough channel flows.

it is obvious that the corrected AWF reasonably well reproduces the temperature distribution for rough walls [45]:

$$\Theta^+ = \frac{1}{0.8h^{+-0.2}\Pr_t^{-0.44}} + \frac{\Pr_t}{\kappa} \ln \frac{32.6y^+}{h^+} \quad (89)$$

where $\Pr_t = 0.9$ and $\kappa = 0.418$. This correlation is based on the experiments of $\Pr = 1.20$ to 5.94 at the order of Re is 10^4 to 10^5 .

9.4.6 AWF for high Schmidt number flows

Turbulent mass transfer across liquid interfaces is sometimes very important in engineering and environmental issues. For example, in order to estimate the amount of the absorption of atmospheric CO_2 (carbon dioxide) at the sea surface, one should consider predicting turbulent mass transfer across air–water interfaces. The difficulty of analysing such a phenomenon comes from that the concentration (scalar)

boundary layer is very much thinner than that of the flow boundary layer. In fact, the Schmidt number Sc is the order of $O(10^3)$. Since space resolved accurate experiments of such free-surface mass transfer are rather difficult compared with those of solid walls, numerical approaches are promising (Calmet and Magnaudet [46]; Hasegawa and Kasagi [47]) to investigate the detailed mass transfer mechanisms. Although those approaches are based on LES or DNS which require high grid resolutions for capturing fine-scale flow and scalar transfer, they are not practical to be used in the estimation of the CO_2 absorption at a large water surface even with the recent computer environment. In order to provide a more practical strategy, the AWF scheme, which requires only rather coarse wall-function grids instead of fine grids resolving the thin scalar boundary layers, can be applied [13].

AWF modelling for high Schmidt number flows

The theoretical limiting variations of the velocity components and their fluctuations near an air–liquid interface are:

$$\begin{aligned} u &= a_u + b_{uy} + c_{uy}y^2 + \dots, v = a_v + b_{vy} + c_{vy}y^2 + \dots \\ u' &= a'_u + b'_{uy} + c'_{uy}y^2 + \dots, v' = a'_v + b'_{vy} + c'_{vy}y^2 + \dots \end{aligned} \quad (90)$$

Since the eddy viscosity relates the Reynolds stress with the mean velocity gradient as $-\rho\bar{u}\bar{v} = \mu_t \partial U / \partial y$, the above relations lead to:

$$-\rho\{\bar{a}'_u\bar{a}'_v + (\bar{a}'_u\bar{b}'_v + \bar{a}'_v\bar{b}'_u)y + \dots\} = \mu_t(\bar{b}_u + 2\bar{c}_u y + \dots) \quad (91)$$

and it is obvious that $\mu_t \propto O(y^0)$. Thus, the near-interface variation of μ_t is modelled as:

$$\mu_t = \alpha\beta\mu \max\{0, (y^* - y_v^*)\} \quad (92)$$

where β is a factor for adjusting the length scale distribution for the near-free-surface turbulence. In the case of non-surface disturbance where $a'_v = 0$, equation (92)

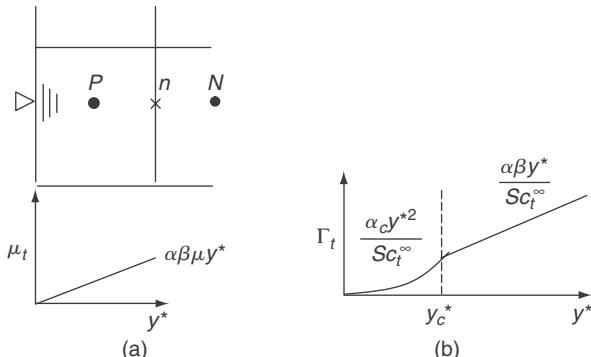


Figure 9.28. (a) Near surface cell arrangement and the eddy viscosity distribution, (b) turbulent diffusivity distribution.

reduces to $\mu_t = \alpha\beta\mu y^*$ as in Figure 9.28a since equation (91) leads to $\mu_t \propto O(y)$. On constant surface concentration conditions, the surface limiting behaviour of concentration and its fluctuation are:

$$c = a_c + b_c y + c_c y^2 + \dots, c' = a'_c + b'_c y + c'_c y^2 + \dots \quad (93)$$

When the turbulent concentration flux $-\rho\bar{v}\bar{c}$ is modelled as $-\rho\bar{v}\bar{c} = \mu\Gamma_t \partial\bar{c}/\partial y$,

$$-\rho\{\bar{a}'_c a'_v + (\bar{b}'_c a'_v + \bar{a}'_c b'_v)y + (\bar{a}'_c c'_v + \bar{c}'_c a'_v)y^2 + \dots\} = \mu\Gamma_t(\bar{b}_c + 2\bar{c}_c y + \dots) \quad (94)$$

Thus, $\Gamma_t \propto O(y^0)$. (In the case of non-surface-disturbance, $\Gamma_t \propto O(y^2)$ due to $a'_v = 0$ and $a'_c = 0$.) In the context of the eddy viscosity models, the turbulent scalar diffusivity Γ_t is modelled using a turbulent Schmidt number as $\Gamma_t = \mu_t/(\mu Sc_t)$; thus, with equation (92) one can rewrite this as:

$$\Gamma_t = \alpha\beta \max\{0, (y^* - y_v^*)\}/Sc_t \quad (95)$$

In order to satisfy the limiting behaviour of Γ_t near the surface, the limiting behaviour of Sc_t is required as $O(y^{-1})$. Thus, Sc_t is:

$$Sc_t = \begin{cases} Sc_t^\infty / (y^*/y_c^*), & 0 \leq y^* \leq y_c^* \\ Sc_t^\infty, & y_c^* < y^* \end{cases} \quad (96)$$

where Sc_t^∞ is a prescribed constant. This simple two-segment variation profile of Sc_t leads to the turbulent diffusivity distribution as in Figure 9.28b:

$$\Gamma_t = \begin{cases} \alpha_c y^{*2} / Sc_t^\infty, & 0 \leq y^* \leq y_c^* \\ \alpha\beta y^* / Sc_t^\infty, & y_c^* < y^* \end{cases} \quad (97)$$

for the case of non-surface disturbance where $\Gamma_t \propto O(y^2)$. The coefficient α_c and y_c^* have the relation $\alpha_c = \alpha\beta/y_c^*$ for connecting the two segments. Then, with the assumption that the rhs terms can be constant over the cell, the simplified concentration equation in the surface adjacent cell:

$$\frac{\partial}{\partial y^*} \left[\left(\frac{\mu}{Sc} + \mu\Gamma_t \right) \frac{\partial\bar{c}}{\partial y^*} \right] = \underbrace{\frac{\nu^2}{k_p} \left[\frac{\partial}{\partial x} (\rho U \bar{c}) \right]}_{C_C} \quad (98)$$

can be easily integrated analytically to form the boundary conditions of the concentration at the surface, namely the surface concentration flux:

$$q_s = -\frac{\mu}{Sc} \frac{d\bar{c}}{dy} \Big|_s = -\frac{k_p^{1/2}}{\nu} A_C \quad (99)$$

or the surface concentration:

$$\bar{c}_s = \bar{c}_n + \frac{q_s Sc D_C}{\rho k_p^{1/2}} + \frac{Sc C_C E_C}{\mu} \quad (100)$$

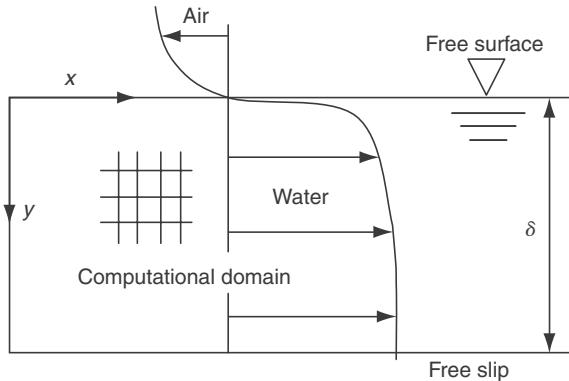


Figure 9.29. Computational domain of free-surface flows.

where the integration constants A_C , D_C and E_C are

$$A_C = \{\mu(\bar{c}_n - \bar{c}_s)/Sc + C_C E_C\} / D_C \quad (101)$$

$$D_C = \frac{1}{\alpha_{cd}^{1/2}} \tan^{-1} (\alpha_{cd}^{1/2} y_c^*) - \frac{1}{\alpha_{ct}} \ln \left(\frac{1 + \alpha_{ct} y_c^*}{1 + \alpha_{ct} y_n^*} \right) \quad (102)$$

$$E_C = \frac{1}{\alpha_{ct}} (y_c^* - y_n^*) - \frac{1}{\alpha_{ct}^2} \ln \left(\frac{1 + \alpha_{ct} y_c^*}{1 + \alpha_{ct} y_n^*} \right) - \frac{1}{2\alpha_{cd}} \ln (1 + \alpha_{ct} y_c^*) \quad (103)$$

with $\alpha_{ct} = \alpha\beta Sc_t / Sc$ and $\alpha_{cd} = \alpha_{ct}/y_c^*$. The model coefficients are $Sc_t^\infty = 0.9$, $\beta = 0.55$ and $y_c^* = 11.7$.

Examples of applications

The example case is a fully developed counter-current air–water flow driven by a constant pressure gradient. Figure 9.29 illustrates the coordinate system and the computational domain applied in the present computations. Only the water phase is solved with the non-disturbance and the constant concentration conditions at the free surface. At the bottom of the domain, a free slip and a constant concentration-flux condition are imposed. The coupled turbulence model for core flows is the standard k - ε model. Figure 9.30a–c compare the present application results with the DNS results of the free-surface turbulent flow with surface-shear whose Reynolds number is $Re_\tau = 150$ by Hasegawa and Kasagi [47]. Figure 9.30a shows that the agreement between the present and the DNS results of the mean velocity distribution is satisfactory and the profiles are significantly lower than that of the log-law line for wall turbulence. Figure 9.30b and c compare the mean concentration fields of $1 \leq Sc \leq 10^3$. The concentration profiles reasonably agree with the DNS results in the various Schmidt numbers. This means that the AWF model can reproduce essential near-surface behaviour of high Schmidt number turbulent flows.

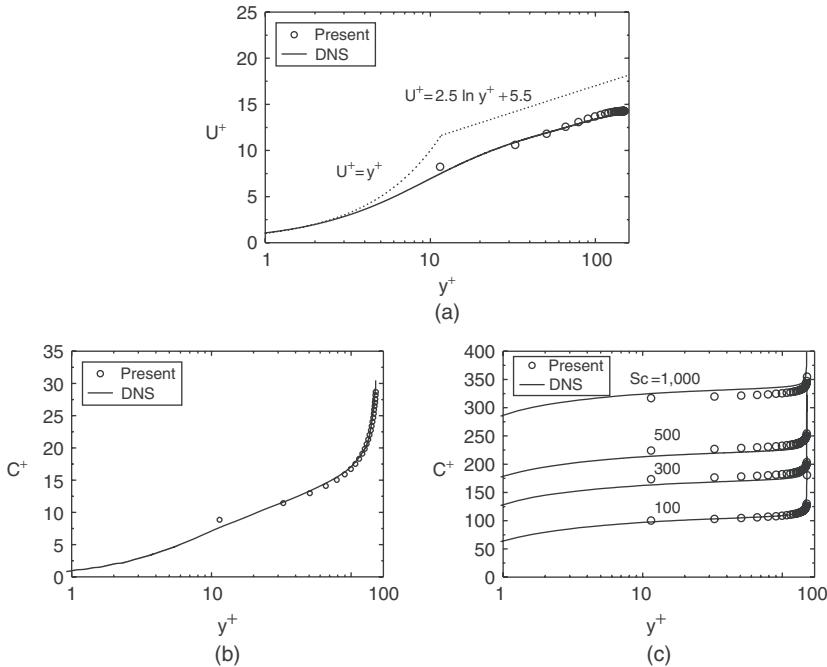


Figure 9.30. (a) Mean velocity distribution of a turbulent free-shear boundary layer, (b) mean concentration distribution near a free surface at $\text{Sc} = 1$, (c) mean concentration distribution near a free surface at $\text{Sc} = 100 - 1,000$.

9.5 Conclusions

The AWFs, which are for the replacement of the LWF and the treatment of complex surface flow phenomena, have been summarised. Their overall strategies for derivation and implementation have been presented with typical application results. From the results, it has been shown that the AWFs are valid in many important flows over a wide range of Reynolds, Prandtl and Schmidt numbers. The effects of wall roughness and/or wall permeability can be also included in the AWF as well. Although they do not include the conventional log-laws, the AWFs accurately reproduce the logarithmic profiles of mean velocities and temperatures without grid dependency. The AWFs perform well with second moment closures, linear and nonlinear $k - \varepsilon$ models while the predictive accuracy depends on the turbulence model employed in the main flow field. The computational cost required for the AWFs is comparable to that of the standard LWF.

In the rough-wall version of the AWF, the effects of wall roughness on turbulence and heat transfer are considered by employing functional forms of equivalent sand grain roughness for the non-dimensional thickness of the modelled viscous sub-layer and the turbulent Prandtl number immediately adjacent to the wall.

Although the permeable-wall version of the AWF does not need to resolve the region inside a permeable wall, by non-linearly linking to the solution of the modified Brinkman equations for the flows inside porous media, it can bridge the flows inside and outside of the porous media successfully. Since the computed turbulent flow fields over permeable porous surfaces are reasonably in accord with the corresponding data, the AWF for permeable wall turbulence models such complicated wall-flow interactions.

By linking to the correct near-wall variation of turbulent viscosity $\mu_t \propto y^3$, the AWF can be successfully applicable to flows over a wide range of Prandtl numbers up to $\text{Pr} = 4 \times 10^4$, for smooth wall cases. (For flow fields and thermal fields at $\text{Pr} \leq 1$, it is not totally necessary to employ the correct near-wall variation of turbulent viscosity.) For rough wall cases, it is confirmed that the model function of Pr_t inside roughness elements enables the AWF to perform well in high Pr flows at least at $\text{Pr} \leq 10$.

With a slight modification, the AWF can reproduce the characteristics of the turbulence near free surfaces. The predictive performance of such a version of the AWF is reasonable in turbulent concentration fields in the range of $\text{Sc} = 1$ to 1,000.

Acknowledgements

The author thanks Professors B. E. Launder, H. Iacovides and Dr. T. J. Craft of the University of Manchester for their fruitful discussions and comments on the AWFs.

Appendix A

The analytical integration of the momentum and energy equations (17) and (18) is performed in the four different cases illustrated in figure 9.8. The process for the case (a) $y_v < 0$ is as follows:

The integration of equation (17) with equation (40) is:

If $y^* \leq h^*$,

$$\frac{dU}{dy^*} = \frac{A_U}{\mu\{1 + \alpha(y^* - y_v^*)\}} = \frac{A_U}{\mu Y^*} \quad (\text{A.1})$$

$$U = \frac{A_U}{\alpha\mu} \ln Y^* + B_U \quad (\text{A.2})$$

At a wall: $y^* = 0$, the condition $U = 0$ leads to:

$$B_U = -\frac{A_U}{\alpha\mu} \ln(1 - \alpha y_v^*) = -\frac{A_U}{\alpha\mu} \ln Y_w^* \quad (\text{A.3})$$

When $h^* < y^*$, the integration leads to:

$$\frac{dU}{dy^*} = \frac{C_U y^* + A'_U}{\mu Y^*} \quad (\text{A.4})$$

Table 9.A1. Cell averaged generation \bar{P}_k , and integration constant: A_U ; (a) $y_v < 0$, (b) $0 \leq y_v \leq h$, (c) $h \leq y_v \leq y_n$, (d) $y_n \leq y_v$; $U_w = 0$ for non-permeable walls

Case	\bar{P}_k
(a)	$\frac{\alpha}{\mu^2 y_n^*} \frac{k_p}{v} \left\{ \int_0^{h^*} (y^* - y_v^*) \left(\frac{A_U}{Y^*} \right)^2 dy^* + \int_{h^*}^{y_n^*} (y^* - y_v^*) \left(\frac{C_U(y^* - h^*) + A_U}{Y^*} \right)^2 dy^* \right\}$
(b)	$\frac{\alpha}{\mu^2 y_n^*} \frac{k_p}{v} \left\{ \int_{y_v^*}^{h^*} (y^* - y_v^*) \left(\frac{A_U}{Y^*} \right)^2 dy^* + \int_{h^*}^{y_n^*} (y^* - y_v^*) \left(\frac{C_U(y^* - h^*) + A_U}{Y^*} \right)^2 dy^* \right\}$
(c)	$\frac{\alpha}{\mu^2 y_n^*} \frac{k_p}{v} \int_{y_v^*}^{y_n^*} (y^* - y_v^*) \left(\frac{C_U(y^* - h^*) + A_U}{Y^*} \right)^2 dy^*$
(d)	0
Case	A_U
(a)	$\left\{ \alpha \mu (U_n - U_w) - C_U (y_n^* - h^*) + C_U \left(\frac{Y_w^*}{\alpha} + h^* \right) \ln [Y_n^* / Y_h^*] \right\} / \ln [Y_n^* / Y_w^*]$
(b)	$\left\{ \alpha \mu (U_n - U_w) - C_U (y_n^* - h^*) + C_U \left(\frac{Y_w^*}{\alpha} + h^* \right) \ln [Y_n^* / Y_h^*] \right\} / (\alpha y_v^* + \ln Y_n^*)$
(c)	$\left\{ \alpha \mu (U_n - U_w) - C_U (y_n^* - y_v^*) + C_U \left(\frac{Y_w^*}{\alpha} + h^* \right) \ln Y_n^* - \frac{\alpha}{2} C_U (y_v^{*2} - 2h^* y_v^* + h^{*2}) \right\} / (\alpha y_v^* + \ln Y_n^*)$
(d)	$\left\{ \mu (U_n - U_w) - \frac{1}{2} C_U (y_n^{*2} - 2h^* y_n^* + h^{*2}) \right\} / y_n^*$

$$U = \frac{C_U}{\alpha \mu} y^* + \left(\frac{A'_U}{\alpha \mu} - \frac{Y_w^* C_U}{\alpha^2 \mu} \right) \ln Y^* + B'_U \quad (A.5)$$

Then, a monotonic distribution condition of dU/dy^* and U at $y^* = h^*$ gives:

$$\frac{A_U}{\mu Y_h^*} = \frac{C_U h^* + A'_U}{\mu Y_h^*} \quad (A.6)$$

$$\frac{A_U}{\alpha \mu} \ln Y_h^* + B_U = \frac{C_U}{\alpha \mu} h^* + \left(\frac{A'_U}{\alpha \mu} - \frac{Y_w^* C_U}{\alpha^2 \mu} \right) \ln Y_h^* + B'_U \quad (A.7)$$

Since at $y^* = y_n^*$ the velocity is $U = U_n$ (U_n is obtainable by interpolating the calculated node values at P, N),

$$U_n = \frac{C_U}{\alpha \mu} y_n^* + \left(\frac{A'_U}{\alpha \mu} - \frac{Y_w^* C_U}{\alpha^2 \mu} \right) \ln Y_n^* + B'_U \quad (A.8)$$

The integration constants A_U, B_U, A'_U and B'_U are thus easily obtainable by solving equations (A.3, A.6–A.8).

Table 9.A2. Coefficients in equation (45)

Case	D_T
(a)	$\frac{-\beta_T h^*}{\alpha_T - \beta_T} + \frac{\alpha_T Y_h^{\beta T}}{(\alpha_T - \beta_T)^2} \ln(Y_h^{\alpha T} / \lambda_b) + \frac{1}{\alpha_T} \ln(Y_n^{\alpha T} / Y_h^{\alpha T})$
(b)	$\frac{\alpha_T y_v^* - \beta_T h^*}{\alpha_T - \beta_T} + \frac{\alpha_T Y_h^{\beta T}}{(\alpha_T - \beta_T)^2} \ln(Y_h^{\alpha T} / Y_h^{\beta T}) + \frac{1}{\alpha_T} \ln(Y_n^{\alpha T} / Y_h^{\alpha T})$
(c)	$\frac{1}{\alpha_T} \ln Y_n^{\alpha T} + y_v^*$
(d)	y_n^*
Case	E_T
(a)	$\frac{h^* - y_n^*}{\alpha_T} + \frac{1}{\alpha_T} \left\{ \frac{Y_w^{\alpha T}}{\alpha_T} + h^* - \frac{Y_h^{\alpha T}}{\alpha_T - \beta_T} \left(\frac{\lambda_b \beta_T}{\alpha_T - \beta_T} + 1 \right) + \frac{\lambda_b \alpha_T Y_h^{\beta T}}{(\alpha_T - \beta_T)^2} \right\} \ln(Y_n^{\alpha T} / Y_h^{\alpha T})$ $+ \frac{\alpha_T \lambda_b Y_h^{\beta T}}{(\alpha_T - \beta_T)^3} \ln(Y_h^{\alpha T} / \lambda_b) - \frac{h^*}{\alpha_T - \beta_T} \left(1 + \frac{\beta_T h^*}{2} + \frac{\beta_T \lambda_b}{\alpha_T - \beta_T} \right)$
(b)	$\frac{h^* - y_n^*}{\alpha_T} + \frac{1}{\alpha_T} \left\{ \frac{Y_w^{\alpha T}}{\alpha_T} + h^* - \frac{Y_h^{\alpha T}}{\alpha_T - \beta_T} \left(\frac{\lambda_b \beta_T}{\alpha_T - \beta_T} + 1 \right) + \frac{\lambda_b \alpha_T Y_h^{\beta T}}{(\alpha_T - \beta_T)^2} \right\} \ln(Y_n^{\alpha T} / Y_h^{\alpha T})$ $+ \frac{\alpha_T \lambda_b Y_h^{\beta T}}{(\alpha_T - \beta_T)^3} \ln(Y_h^{\alpha T} / Y_h^{\beta T}) - \frac{h^*}{\alpha_T - \beta_T} \left(1 + \frac{\beta_T h^*}{2} + \frac{\beta_T \lambda_b}{\alpha_T - \beta_T} \right) + \frac{y_v^*}{\alpha_T - \beta_T} \left(\frac{\alpha_T Y_h^{\beta T}}{\alpha_T - \beta_T} - \frac{\alpha_T y_v^*}{2} \right)$
(c)	$\frac{y_v^* - y_n^*}{\alpha_T} - y_v^{*2}/2 + \frac{Y_w^{\alpha T}}{\alpha_T^2} \ln Y_n^{\alpha T}$
(d)	$-y_n^{*2}/2$

In this case, the wall shear stress τ_w is written as:

$$\tau_w = (\mu + \mu_t) \frac{dU}{dy} \Big|_{y=0} = \mu Y_w^* \frac{k_p^{1/2}}{\nu} \frac{dU}{dy^*} \Big|_{y^*=0} = \frac{k_p^{1/2} A_U}{\nu} \quad (\text{A.9})$$

whose resultant form is the same as equation (27) and can be calculated with A_U .

Since the cell averaged production term $\overline{P_k}$ of the turbulence energy k is written as:

$$\begin{aligned} \overline{P_k} &= \frac{1}{y_n} \left\{ \int_0^h v_t \left(\frac{dU}{dy} \right)^2 dy + \int_h^{y_n} v_t \left(\frac{dU}{dy} \right)^2 dy \right\} \\ &= \frac{\alpha}{y_n^*} \frac{k_p}{\nu} \left\{ \int_0^{h^*} (y^* - y_v^*) \left(\frac{dU}{dy^*} \right)^2 dy^* + \int_{h^*}^{y_n^*} (y^* - y_v^*) \left(\frac{dU}{dy^*} \right)^2 dy^* \right\} \end{aligned} \quad (\text{A.10})$$

it can be calculated by substituting dU/dy^* with equations (A.1) and (A.4).

As for the thermal field, when $y^* \leq h^*$, applying equations (41) and (42) to the energy equation (18), one can lead to:

$$\frac{\partial}{\partial y^*} \left[\left\{ \frac{\mu}{\text{Pr}} + \frac{\alpha \mu (y^* - y_v^*)}{\text{Pr}_t^\infty + C_0(1 - y^*/h^*)} \right\} \frac{\partial \Theta}{\partial y^*} \right] = C_T \quad (\text{A.11})$$

Its integration follows:

$$\left\{ \frac{\mu}{\text{Pr}} + \frac{\alpha\mu(y^* - y_v^*)}{\text{Pr}_t^\infty + C_0(1 - y^*/h^*)} \right\} \frac{d\Theta}{dy^*} = C_T y^* + A_T \quad (\text{A.12})$$

$$\frac{d\Theta}{dy^*} = \frac{\{1 + \beta_T(h^* - y^*)\}(C_T y^* + A_T) \text{Pr}}{1 + \alpha_T(y^* - y_v^*) + \beta_T(h^* - y^*) \mu} \quad (\text{A.13})$$

The further integration leads to:

$$\begin{aligned} \Theta = & - \left\{ \frac{C_T \lambda_b \beta_T}{(\alpha_T - \beta_T)^2} - \frac{C_T(1 + \beta_T h^*) + \beta_T A_T}{\alpha_T - \beta_T} \right\} \frac{\text{Pr}}{\mu} y^* + \frac{\text{Pr} C_T \beta_T}{2\mu(\alpha_T - \beta_T)} y^{*2} \\ & + \left[\frac{C_T \lambda_b^2 \beta_T}{(\alpha_T - \beta_T)^3} - \frac{\lambda_b \{C_T(1 + \beta_T h^*) + \beta_T A_T\}}{(\alpha_T - \beta_T)^2} \right. \\ & \left. + \frac{(1 + \beta_T h^*) A_T}{\alpha_T - \beta_T} \right] \frac{\text{Pr}}{\mu} \ln [(\alpha_T - \beta_T) y^* + \lambda_b] + B_T \end{aligned} \quad (\text{A.14})$$

When $y^* > h^*$

$$\frac{d\Theta}{dy^*} = \frac{\text{Pr}(C_T y^* + A'_T)}{\mu \{1 + \alpha_T(y^* - y_v^*)\}} = \frac{\text{Pr}(C_T y^* + A'_T)}{\mu Y^{\alpha T}} \quad (\text{A.15})$$

$$\Theta = \frac{\text{Pr} C_T}{\mu \alpha_T} y^* + \frac{\text{Pr}}{\mu} \left\{ \frac{A'_T}{\alpha_T} - \frac{C_T Y_w^{\alpha T}}{\alpha_T^2} \right\} \ln Y^{\alpha T} + B'_T \quad (\text{A.16})$$

The integration constants A_T , B_T , A'_T and B'_T are readily obtainable by equations (A.13–A.16) imposing the boundary conditions ($\Theta|_{y=0} = \Theta_w$, $\Theta|_{y=y_n} = \Theta_n$) and a monotonic distribution condition of $d\Theta/dy^*$ and Θ at $y^* = h^*$ as in the flow field.

In the case of a constant wall temperature, the wall heat flux q_w can be calculated as:

$$q_w = -\rho c_p \left(\frac{\nu}{\text{Pr}} + \frac{\nu_t}{\text{Pr}_t} \right) \frac{k_P^{1/2}}{\nu} \left. \frac{d\Theta}{dy^*} \right|_w = -\frac{\rho c_p k_P^{1/2}}{\mu} A_T \quad (\text{A.17})$$

whose resultant form is the same as equation (28) and can be calculated with A_T .

In the other cases (b)–(d), integration can be performed by the similar manner.

Appendix B

The integrals of the functions are:

$$\int \frac{1}{\Gamma_{\theta a}} dy^* = \int \frac{1}{1 + \frac{\alpha' \Pr_t y^{*3}}{\Pr_t}} dy^* \\ = \frac{a}{3} \left\{ \frac{1}{2} \ln \frac{(y^* + a)^2}{y^{*2} - ay^* + a^2} + \sqrt{3} \tan^{-1} \frac{2y^* - a}{a\sqrt{3}} \right\} \quad (\text{B.1})$$

$$\int \frac{y^*}{\Gamma_{\theta a}} dy^* = \int \frac{y^*}{1 + \frac{\alpha' \Pr_t y^{*3}}{\Pr_t}} dy^* \\ = \frac{a^2}{3} \left\{ -\frac{1}{2} \ln \frac{(y^* + a)^2}{y^{*2} - ay^* + a^2} + \sqrt{3} \tan^{-1} \frac{2y^* - a}{a\sqrt{3}} \right\} \quad (\text{B.2})$$

$$\int \frac{1}{\Gamma_{\theta b}} dy^* = \int \frac{1}{1 + \frac{\alpha \Pr_t (y^* - y_v^*)}{\Pr_t}} dy^* = \frac{1}{\alpha_\theta} \ln |1 + \alpha_\theta(y^* - y_v^*)| \quad (\text{B.3})$$

$$\int \frac{y^*}{\Gamma_{\theta b}} dy^* = \int \frac{y^*}{1 + \frac{\alpha \Pr_t (y^* - y_v^*)}{\Pr_t}} dy^* \\ = \frac{y^*}{\alpha_\theta} - \frac{1 - \alpha_\theta y_v^*}{\alpha_\theta^2} \ln |1 + \alpha_\theta(y^* - y_v^*)| \quad (\text{B.4})$$

$$\int \frac{1}{\Gamma_{\theta d}} dy^* = \int \frac{1}{1 + \frac{\alpha \Pr_t (y^* - y_v^*)}{\Pr_t^\infty + C_h \max(0, 1 - y^*/h^*)}} dy^* \\ = -\frac{\beta_T y^*}{\alpha_T - \beta_T} + \left\{ \frac{\beta_T \lambda_b}{(\alpha_T - \beta_T)^2} + \frac{1 + \beta_T h^*}{\alpha_T - \beta_T} \right\} \\ \times \ln |(\alpha_T - \beta_T)y^* + \lambda_b| \quad (\text{B.5})$$

$$\int \frac{y^*}{\Gamma_{\theta d}} dy^* = \int \frac{y^*}{1 + \frac{\alpha \Pr_t (y^* - y_v^*)}{\Pr_t^\infty + C_h \max(0, 1 - y^*/h^*)}} dy^* \\ = -\frac{\beta_T y^{*2}}{2(\alpha_T - \beta_T)} + \left\{ \frac{\beta_T \lambda_b}{(\alpha_T - \beta_T)^2} + \frac{1 + \beta_T h^*}{\alpha_T - \beta_T} \right\} y^* \\ - \left\{ \frac{\beta_T \lambda_b^2}{(\alpha_T - \beta_T)^3} + \frac{\lambda_b(1 + \beta_T h^*)}{(\alpha_T - \beta_T)^2} \right\} \\ \times \ln |(\alpha_T - \beta_T)y^* + \lambda_b| \quad (\text{B.6})$$

where $\alpha_\theta = \alpha \text{Pr}/\text{Pr}_t$, $\alpha_T = \alpha \text{Pr}/\text{Pr}_t^\infty$, $a = (\alpha' \text{Pr}/\text{Pr}_t)^{-1/3}$

$$\beta_T = \begin{cases} C_h / (\text{Pr}_t^\infty h^*), & \text{for } y \leq h \\ 0, & \text{for } h < y \end{cases}$$

$\lambda_b = 1 - \beta_T h^* - \alpha y_v^*$. (Note that integration constants are neglected in the results.)

$$\begin{aligned} \int \frac{1}{\Gamma_{\theta c}} dy^* &= \int \frac{1}{1 + \frac{\alpha' \text{Pr}(y^* + \delta_v)^3}{\text{Pr}_t^\infty + C_h \max(0, 1 - y^*/h^*)}} dy^* \\ &= \frac{\zeta}{\alpha'_T} \left[- \left\{ \eta_c \beta_T - (\eta_a - \eta_b)(1 + \beta_T h^*) - \frac{\eta_b}{2}(1 + \beta_T [h^* + \eta_a]) \right\} \Phi(y) \right. \\ &\quad \left. - \frac{1 + \beta_T(h^* + \eta_a)}{2} \ln |y^{*2} + \eta_b y^* + \eta_c| \right. \\ &\quad \left. + \{1 + \beta_T(h^* + \eta_a)\} \ln |y^* + \eta_a| \right] \end{aligned} \quad (\text{B.7})$$

$$\begin{aligned} \int \frac{y^*}{\Gamma_{\theta c}} dy^* &= \int \frac{y^*}{1 + \frac{\alpha' \text{Pr}(y^* + \delta_v)^3}{\text{Pr}_t^\infty + C_h \max(0, 1 - y^*/h^*)}} dy^* \\ &= \frac{\zeta}{\alpha'_T} \left[\left\{ \eta_c(1 + \beta_T[h^* + \eta_a]) - \frac{\eta_b}{2}(\beta_T[\eta_a \eta_b - \eta_c] \right. \right. \\ &\quad \left. \left. + \eta_a[1 + \beta_T h^*]) \right\} \Phi(y) + \frac{1}{2} [\beta_T[\eta_a \eta_b - \eta_c] \right. \\ &\quad \left. + \eta_a(1 + \beta_T h^*)] \ln |y^{*2} + \eta_b y^* + \eta_c| \right. \\ &\quad \left. - \eta_a \{1 + \beta_T(h^* + \eta_a)\} \ln |y^* + \eta_a| \right] \end{aligned} \quad (\text{B.8})$$

where $\alpha'_T = \alpha' \text{Pr}/\text{Pr}_t^\infty$,

$$\xi = \begin{cases} (-q + \sqrt{q^2 + p^3})^{1/3} + (-q - \sqrt{q^2 + p^3})^{1/3}, & \text{if } q^2 + p^3 \geq 0, \\ 2\sqrt{-p} \cos \left[\frac{1}{3} \cos^{-1} \left(\frac{q}{p\sqrt{-p}} \right) \right], & \text{if } q^2 + p^3 < 0, \end{cases} \quad (\text{B.9})$$

$$\begin{aligned} p &= -\beta_T/(3\alpha'_T), q = \{1 + \beta_T(h^* + \delta_v)\}/(2\alpha'_T), \\ \eta_a &= \delta_v - \xi, \eta_b = 2\delta_v + \xi, \eta_c = \delta_v^2 + \delta_v \xi + \xi^2 + 3p, \zeta = 1/\{\eta_a(\eta_a - \eta_b) + \eta_c\} \end{aligned}$$

and

$$\Phi(y) = \begin{cases} -\frac{2}{\eta_b + 2y^*}, & \text{if } \eta_b^2 - 4\eta_c = 0, \\ \frac{2}{\sqrt{-\eta_b^2 + 4\eta_c}} \tan^{-1} \left(\frac{\eta_b + 2y^*}{\sqrt{-\eta_b^2 + 4\eta_c}} \right), & \text{if } \eta_b^2 - 4\eta_c < 0, \\ \frac{2}{\sqrt{\eta_b^2 - 4\eta_c}} \ln \left| \frac{\eta_b + 2y^* - \sqrt{\eta_b^2 - 4\eta_c}}{\eta_b + 2y^* + \sqrt{\eta_b^2 - 4\eta_c}} \right|, & \text{if } \eta_b^2 - 4\eta_c > 0. \end{cases} \quad (\text{B.10})$$

9.6 Nomenclature

$A_C, A_T, A_U:$	integration constants
$c, \bar{c}:$	concentration, mean concentration
$c_p:$	specific heat capacity at constant pressure
$c_\ell, c_\mu:$	model constants
$C_0-C_3, C_h:$	model coefficients
$C_p:$	pressure coefficient
$C_C:$	sum of the convection and the diffusion terms of the concentration equation
$C_T:$	sum of the convection and the diffusion terms of the energy equation
$C_U:$	sum of the convection and the diffusion terms of the momentum equation
$d_p:$	mean particle diameter
D	pipe diameter
$D_C, D_T, E_C, E_T:$	integration constants
$f:$	friction coefficient of pipe flows
$f_i:$	drag force
$F, F_{ij}:$	Forchheimer correction coefficient, Forchheimer correction tensor
$h, h^*:$	roughness height (equivalent sand grain roughness height), $h\sqrt{k_P}/v$
$h^+:$	roughness Reynolds number: hU_τ/v
$H:$	ramp height or channel height
$k, k_P:$	turbulence energy, k at node P
$K, K_{ij}:$	permeability, permeability tensor
$\ell:$	turbulent length scale
$L:$	ramp length
Nu:	Nusselt number
$P:$	pressure or cell centre of the wall-adjacent cell
$P_k:$	production term of k equation
$\text{Pr}, \text{Pr}_t, \text{Pr}_t^\infty:$	Prandtl number, turbulent Prandtl numbers

$q_w:$	wall heat flux
$q_s:$	surface concentration flux
$\text{Re}, \text{Re}_x, \text{Re}_\theta:$	Reynolds numbers: $U_b D / \nu, U_0 x / \nu, U_e \theta / \nu; \theta:$ momentum thickness
$\text{Re}_K:$	permeability Reynolds number: $U_\tau K^{1/2} / \nu$
$\text{Sc}, \text{Sc}_t, \text{Sc}_t^\infty:$	Schmidt number, turbulent Schmidt numbers
$\text{St}:$	Stanton number: $\text{Nu}/(\text{RePr})$
$S_\theta:$	source term of the energy equation
$U, U^+:$	mean velocity, U/U_τ
$U_b, U_\tau:$	bulk velocity, friction velocity: $\sqrt{\tau_w/\rho}$
$U_d:$	Darcy velocity
$U_e, U_0:$	free stream velocities
$U_n:$	velocity at the point n
$U_w:$	slip velocity
$x:$	wall-parallel coordinate
$x':$	x/L in the ramp flow
$y:$	wall normal coordinate or wall normal distance
$y_n, y_v, y_\varepsilon:$	cell height, viscous sublayer thickness, characteristic dissipation length
$y^+, y^*:$	normalised distances: $yU_\tau/\nu, y\sqrt{k_p}/\nu$
$y_b^*, y_c^*:$	connection points in two segment diffusivity models
$Y^*, Y_i^*:$	$1 + \alpha(y^* - y_v^*), 1 + \alpha(y_i^* - y_v^*); i = w, n, h; y_w = 0; y_h = h$
$Y^{\alpha T}, Y_i^{\alpha T}:$	$1 + \alpha_T(y^* - y_v^*), 1 + \alpha_T(y_i^* - y_v^*); i = w, n, h; y_w = 0; y_h = h$
$Y^{\beta T}, Y_i^{\beta T}:$	$1 + \beta_T(y^* - y_v^*), 1 + \beta_T(y_i^* - y_v^*); i = w, n, h; y_w = 0; y_h = h$
$\alpha:$	$c_\mu c_\ell$ or heat transfer coefficient: $q_w/(\Theta_w - \Theta_{\text{inlet}})$
$\alpha', \alpha_{ct}, \alpha_{cd}:$	$\alpha(y_b^* - y_v^*)/y_b^{*3}, \alpha\beta\text{Sc}/\text{Sc}_t, \alpha_{ct}/y_c^*$
$\alpha_T, \alpha_\theta:$	$\alpha \text{Pr} / \text{Pr}_t^\infty, \alpha \text{Pr} / \text{Pr}_t$
$\beta, \beta_p:$	model coefficients
$\beta_T:$	$C_0/(h^* \text{Pr}_t^\infty)$
$\Gamma_U, \Gamma_\theta, \Gamma_t:$	normalised total viscosity, thermal diffusivity and turbulent diffusivity
$\delta_v:$	origin shift
$\Delta x, \Delta y:$	cell widths in the x, y direction
$\varphi:$	porosity
$\varepsilon:$	dissipation rate of k
$\Theta, \Theta^+:$	mean temperature, $ \Theta - \Theta_w (\rho c_p U_\tau)/q_w$
$\Theta_w, \Theta_n:$	wall temperature, temperature at the point n
$\kappa, \kappa_t:$	von Kármán constants
$\lambda_b:$	$Y_w^{\alpha T} + \beta_T h^*$
$\mu, \mu_t:$	viscosity, turbulent viscosity
$v, v_t:$	kinematic viscosity, kinematic turbulent viscosity
$\rho:$	fluid density
$\tau_w:$	wall shear stress
$\langle \phi \rangle, \langle \phi \rangle^f:$	superficial and intrinsic averaged values of ϕ
$\bar{\phi}:$	Reynolds averaged value of ϕ

References

- [1] Launder, B. E. On the computation of convective heat transfer in complex turbulent flows, *ASME J. Fluid Eng.*, 110, pp. 1112–1128, 1988.
- [2] Kim, J., Moin, P., and Moser, R. Turbulence statistics in fully developed channel flow at low Reynolds number, *J. Fluid Mech.*, 177, pp. 133–166, 1987.
- [3] Ahmed, A., and Demoulin, M. Turbulence modelling in the automotive industry, in: *Engineering Turbulence Modelling and Measurements–5*, W. Rodi and N. Fueyo (Eds.), Elsevier Science Ltd., Amsterdam, pp. 29–42, 2002.
- [4] Launder, B. E., and Spalding, D. B. The numerical computation of turbulent flows, *Comp. Meth. Appl. Mech. Eng.*, 3, pp. 269–289, 1974.
- [5] Chieng, C. C., and Launder, B. E. On the calculation of turbulent heat transport downstream from an abrupt pipe expansion, *Numerical Heat Transfer*, 3, pp. 189–207, 1980.
- [6] Amano, R. S. Development of turbulence near wall model and its application to separated and reattached flows, *Numerical Heat Transfer*, 7, pp. 59–76, 1984.
- [7] Launder, B. E. Numerical computation of convective heat transfer in complex turbulent flows: time to abandon wall functions?, *Int. J. Heat Mass Transfer*, 27, pp. 1485–1490, 1984.
- [8] Craft, T. J., Gerasimov, A. V., Iacovides, H., and Launder, B. E. Progress in the generalization of wall-function treatments, *Int. J. Heat Fluid Flow*, 23, pp. 148–160, 2002.
- [9] Cabot, W., and Moin, P. Approximate wall boundary conditions in the large-eddy simulation of high Reynolds number flow, *Flow Turb. Combust.*, 63, pp. 269–291, 1999.
- [10] Suga, K., Craft, T. J., and Iacovides, H. An analytical wall-function for turbulent flows and heat transfer over rough walls, *Int. J. Heat Fluid Flow*, 27, pp. 852–866, 2006.
- [11] Suga, K., and Nishiguchi, S. Computation of turbulent flows over porous/fluid interfaces, *Fluid Dyn. Res.*, 41, pp. 012401:1–15, 2009.
- [12] Suga, K. Computation of high Prandtl number turbulent thermal fields by the analytical wall-function, *Int. J. Heat Mass Transfer*, 50, pp. 4967–4974, 2007.
- [13] Suga, K., and Kubo, M. Modelling turbulent high Schmidt number mass transfer across underformable gas-liquid interfaces, *Int. J. Heat Mass Transfer*, 53, pp. 2989–2995, 2010.
- [14] Patankar, S. V. *Numerical Heat Transfer and Fluid Flow*. Hemisphere/McGraw Hill, Washington, 1980.
- [15] Rhie, C. M., and Chow, W. L. Numerical study of the turbulent flow past an airfoil with trailing edge separation, *AIAA J.*, 21, pp. 1525–1532, 1983.
- [16] Lien, F. S., and Leschziner, M. A. Upstream monotonic interpolation for scalar transport with application in complex turbulent flows, *Int. J. Num. Meth. Fluids*, 19, pp. 527–548, 1994.
- [17] Gerasimov, A. V. Development and validation of an analytical wall-function strategy for modelling forced, mixed and natural convection phenomena, Ph.D. Thesis, UMIST, Manchester, UK, 2003.
- [18] Suga, K., and Basara, B. Application of the analytical wall-function of turbulence to IC engine flows, Proc. 43rd Nat. Heat Transfer Symp., Japan, Nagoya, CD ROM, 2006.
- [19] Suga, K. A study toward improving accuracy of large scale industrial turbulent flow computations, in: *Frontiers of Computational Science*, Y. Kaneda, H. Kawamura, M. Sasai (Eds.), Springer, pp. 99–105, 2007.

- [20] Nikuradse, J. Strömungsgesetze in rauhen rohren. *VDI-Forschungsheft*, 361, 1933.
- [21] Cebeci, T., and Bradshaw, P. Momentum Transfer in Boundary Layers, Hemisphere Pub. Co, Washington, 1977.
- [22] Krogstad, P. A., Antonia, R. A., and Browne, L. W. B. Comparison between rough- and smooth-wall turbulent boundary layers, *J. Fluid Mech.*, 245, pp. 599–617, 1992.
- [23] Tachie, M. F., Bergstrom, D. J., and Balachandar, R. Roughness effects in low-Re₀ open-channel turbulent boundary layers, *Expt. Fluids*, 35, pp. 338–346, 2003.
- [24] Nagano, Y., Hattori, H., and Houra, T. DNS of velocity and thermal fields in turbulent channel flow with transverse-rib roughness, *Int. J. Heat Fluid Flow*, 25, pp. 393–403, 2004.
- [25] Launder, B. E., and Sharma, B. I. Application of the energy-dissipation model of turbulence to the calculation of flow near a spinning disc, *Lett. Heat Mass Transfer*, 1, pp. 131–138, 1974.
- [26] Craft, T. J., Launder, B. E., and Suga, K. Development and application of a cubic eddy-viscosity model of turbulence, *Int. J. Heat Fluid Flow*, 17, pp. 108–115, 1996.
- [27] Moody, L. F. Friction factors for pipe flow, *Trans. ASME*, 66, pp. 671–678, 1944.
- [28] Turner, A. B., Hubbe-Walker, S. E., and Bayley, F. J. Fluid flow and heat transfer over straight and curved rough surfaces, *Int. J. Heat Mass Transfer*, 43, pp. 251–262, 2000.
- [29] van Mierlo, M. C. L. M., and de Ruiter, J. C. C. Turbulence measurements above artificial dunes, Delft Hydraulics Lab. Rep., Q789, Delft, The Netherlands, 1988.
- [30] Iacovides, H., and Raishee, M. Recent progress in the computation of flow and heat transfer in internal cooling passages of gas-turbine blades, *Int. J. Heat Fluid Flow*, 20, pp. 320–328, 1999.
- [31] Song, S., and Eaton, J. K. The effects of wall roughness on the separated flow over a smoothly contoured ramp, *Expt. Fluids*, 33, pp. 38–46, 2002.
- [32] Zippe, H. J., and Graf, W. H. Turbulent boundary-layer flow over permeable and non-permeable rough surfaces, *J. Hydraul. Res.*, 21, pp. 51–65, 1983.
- [33] Breugem, W. P., Boersma, B. J., and Uittenbogaard, R.E. The influence of wall permeability on turbulent channel flow, *J. Fluid Mech.*, 562, pp. 35–72, 2006.
- [34] Beavers, G. S., Joseph, D. D. Boundary conditions at a naturally permeable wall, *J. Fluid Mech.*, 30 pp. 197–207, 1967.
- [35] Whitaker, S. The Forchheimer equation: A theoretical development, *Transp. Porous Media*, 25, pp. 27–61, 1996.
- [36] Ergun, S. Fluid flow through packed columns, *Chem. Eng. Progr.*, 48, pp. 88–94, 1952.
- [37] Macdonald, I. F., El-Syed, M. S., Mow, K., and Dullien, F. A. Flow through porous media: The Ergun equation revised, *Ind. Eng. Chem. Fundam.*, 18, pp. 199–208, 1979.
- [38] Brinkman, H. C. A calculation of viscous force exerted by a flowing fluid on a dense swarm of particles, *Appl. Sci. Res. A*, 1, pp. 27–34, 1948.
- [39] Craft, T. J., and Launder, B. E. Principles and performance of TCL-based second-moment closures *Flow, Turb. Combust.*, 66, pp. 355–372, 2001.
- [40] Rogers, M. M., Mansour, N. N., and Reynolds, W. C. An algebraic model for the turbulent flux of a passive scalar, *J. Fluid Mech.*, 203, pp. 77–101, 1989.
- [41] Suga, K., and Abe, K. Nonlinear eddy viscosity modelling for turbulence and heat transfer near wall and shear-free boundaries, *Int. J. Heat Fluid Flow*, 21(1), pp. 37–48, 2000.
- [42] So, R. M. C., and Sommer, T. P. A near-wall eddy conductivity model for fluids with different Prandtl numbers, *ASME J. Heat Transfer*, 116, pp. 844–854, 1994.

- [43] Kasagi, N., and Shikazono N. Contribution of direct numerical simulation to understanding and modelling turbulent transport, *Proc. R. Soc. Lond. A*, 451, pp. 257–292, 1995.
- [44] Kader, B. A. Temperature and concentration profiles in fully turbulent boundary layers, *Int. J. Heat and Mass Transfer*, 24, pp. 1541–1544, 1981.
- [45] Kays, W. M., and Crawford, M. E. in: Convective Heat and Mass Transfer, 3rd ed., McGraw-Hill, New York, pp. 298–301, 1993.
- [46] Calmet, I., and Magnaudet, J. High-Schmidt number mass transfer through turbulent gas-liquid interfaces *Int. J. Heat Fluid Flow*, 19, pp. 522–532, 1998.
- [47] Hasegawa, Y., and Kasagi, N. Effects of interfacial velocity boundary condition on turbulent mass transfer at high Schmidt numbers, *Int. J. Heat Fluid Flow*, 28, pp. 1192–1203, 2007.

IV. Advanced Simulation Modeling Technologies

This page intentionally left blank

10 SPH – a versatile multiphysics modeling tool

Fangming Jiang¹ and Antonio C.M. Sousa^{2,3}

¹*The Pennsylvania State University, USA*

²*Universidade de Aveiro, Portugal*

³*University of New Brunswick, Canada*

Abstract

In this chapter, the current state-of-the-art and recent advances of a novel numerical method – the Smoothed Particle Hydrodynamics (SPHs) will be reviewed through case studies with particular emphasis to fluid flow and heat transport. To provide sufficient background and to assess its engineering/scientific relevance, three particular case studies will be used to exemplify macro- and nanoscale applications of this methodology. The first application deals with magnetohydrodynamic (MHD) turbulence control. Effective control of the transition to turbulence of an electrically conductive fluid flow can be achieved by applying a stationary magnetic field, which is not simply aligned along the streamwise or transverse flow direction, but along a direction that forms an angle with the main fluid flow in the range of 0° (streamwise) to 90° (transverse). The SPH numerical technique is used to interpret this concept and to analyze the magnetic conditions. The second application deals with non-Fourier ballistic-diffusive heat transfer, which plays a crucial role in the development of nanotechnology and the operation of submicron- and nanodevices. The ballistic-diffusive equation to heat transport in a thin film is solved numerically via the SPH methodology. The third application deals with mesoscopic pore-scale model for fluid flow in porous media. SPH simulations enable microscopic visualization of fluid flow in porous media as well as the prediction of an important macroscopic parameter – the permeability.

Keywords: CFD, Numerical methods, Smoothed particle hydrodynamics

10.1 Introduction

A novel numerical method – the smoothed particle hydrodynamic (SPH) offers a relatively flexible tool for heat and fluid flow computations, as it can cope with a wide range of space scales and of physical phenomena. SPH is a meshless particle-based Lagrangian fluid dynamics simulation technique, in which the fluid flow is

represented by a collection of discrete elements or pseudo-particles. These particles are initially distributed with a specified density distribution and evolve in time according to the constitutive conservation equations (e.g., mass, momentum, energy). Flow properties are determined by an interpolation or smoothing of the nearby particle distribution using a special weighting function – the smoothing kernel. SPH was first proposed by Gingold and Monaghan [1] and by Lucy [2] in the context of astrophysical modeling. The method has been successful in a broad spectrum of problems, among others, heat conduction [3, 4], forced and natural convective flow [5, 6], low Reynolds number flow [7, 8], and interfacial flow [9–11]. Many other applications are given by Monaghan [12] and Randle and Libersky [13], which offer comprehensive reviews on the historical background and some advances of SPH. In comparison to the Eulerian-based computational fluid dynamics (CFD) methods, SPH is advantageous in what concerns the following aspects: (a) particular suitability to tackle problems dealing with multiphysics; (b) ease of handling complex free surface and material interface; and (c) relatively simple computer codes and ease of machine parallelization. These advantages make it particularly well suited to deal with transient fluid flow and heat transport.

This chapter is organized in six main sections, namely: 1. Introduction; 2. SPH theory, formulation, and benchmarking; 3. Control of the onset of turbulence in MHD fluid flow; 4. SPH numerical modeling for ballistic-diffusive heat conduction; 5. Mesoscopic pore-scale SPH model for fluid flow in porous media, and 6. Concluding remarks.

10.2 SPH Theory, Formulation, and Benchmarking

SPH, despite its promise, is far from being at its perfected or optimized stage. Problems with the SPH, such as handling of solid boundaries [14], choice of smoothing kernel and specification of the smoothing length [15], treatment of heat/mass conducting flow [5, 6], and modeling of low Reynolds number flow, where the surface/interface effects act as a dominating role relative to the inertial effect [10], have prevented its application to some practical problems. This section reports an effort to enhance the SPH methodology and develop appropriate computer programmable formulations.

10.2.1 SPH theory and formulation

In the SPH formulation, the continuous flow at time t is represented by a collection of N particles each located at position $\mathbf{r}_i(t)$ and moving with velocity $\mathbf{v}_i(t)$, $i = 1, 2, \dots, N$. The “smoothed” value of any field quantity $q(\mathbf{r}, \mathbf{v})$ at a space point \mathbf{r} (bold text designates vector or tensor) is a weighted sum of all contributions from neighboring particles, namely:

$$\langle q(\mathbf{r}, \mathbf{v}) \rangle = \sum_j \frac{m_j}{\rho(\mathbf{r}_j)} q(\mathbf{r}_j, \mathbf{v}_j) w(|\mathbf{r} - \mathbf{r}_j|, h) \quad (1)$$

$$\langle \nabla q(\mathbf{r}, \mathbf{v}) \rangle = \sum_{j \neq i} \frac{m_j}{\rho(\mathbf{r}_j)} q(\mathbf{r}_j, \mathbf{v}_j) \nabla w(|\mathbf{r} - \mathbf{r}_j|, h) \quad (2)$$

where, m_j and $\rho(\mathbf{r}_j)$, respectively, denote the mass and density of particle j . $w(|\mathbf{r}|, h)$ is the weight or smoothing function with h being the smoothing length.

Density update

Based on equation (1), the local density at position $\mathbf{r}_i(t)$ is:

$$\rho_i = \sum_j m_j w_{ij} \quad (3)$$

The summation is over all neighboring particles j including particle i itself. Equation (1) conserves mass accurately if the total SPH particles are kept constant. In terms of equation (2), another formulation for density calculation can be derived from the continuity equation, namely

$$\frac{d\rho_i}{dt} = \sum_{j \neq i} m_j \mathbf{v}_{ij} \cdot \nabla_i w_{ij} \quad (4)$$

The summation is over all neighboring particles j with exception of particle i itself. Here, $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$, $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, and $w_{ij} = w(|\mathbf{r}_i - \mathbf{r}_j|, h)$; the subscript in the operator ∇_i indicates the gradient derivatives are taken with respect to the coordinates at particle i

$$\nabla_i w_{ij} = \frac{\mathbf{r}_{ij}}{|\mathbf{r}_{ij}|} \frac{\partial w_{ij}}{\partial r_i} \quad (5)$$

Equation (4) is the Galilean invariant, which is more appropriate for interface reconstruction in free surface flow simulations. The use of this equation has the computational advantage over equation (3) of calculating all rates of change in one pass over the particles; whereas the use of equation (3) requires one pass to calculate the density, then another one to calculate the rates of change for velocity, and different scalars, such as temperature.

Equation (4) is employed to perform the density update in the case studies to be presented.

Incompressibility

A quasi-compressible method is implemented to determine the *dynamic* pressure p [8], in which an artificial equation of state is used

$$p = p_0 \left[\left(\frac{\rho}{\rho_0} \right)^\gamma - 1.0 \right] \quad (6)$$

where, p_0 is the magnitude of the pressure for material state corresponding to the reference density ρ_0 . The ratio of the specific heats γ is artificially a large value

(say, 7.0) to guarantee the incompressibility of the fluid. The speed of sound, c_s , can be formulated as:

$$c_s^2 = \frac{\gamma p_0}{\rho_0} = \alpha U^2 \quad (7)$$

where, U is the characteristic or maximum fluid velocity. The choice of α is a compromise: it should be an adequate value to avoid making p_0 (or c_s) too large, which will force the time advancement of the simulation to become prohibitively slow, and also to avoid yielding Mach numbers that can violate the incompressibility condition of the fluid; α takes 100 in the case studies to be presented, which ensures the density variation be less than 1.0%.

Velocity rate of change

The SPH formulation of viscous effects in fluid flow requires careful modeling to satisfy its adaptivity and robustness. In this respect, Morris et al. [8] presented the following expression, equation (8), to simulate viscous diffusion.

$$\left\{ \left(\frac{1}{\rho} \nabla \cdot \mu \nabla \right) \mathbf{v} \right\}_i = \sum_{j \neq i} \frac{m_j (\mu_i + \mu_j) \mathbf{v}_{ij}}{\rho_i \rho_j} \left(\frac{1}{|\mathbf{r}_{ij}|} \frac{\partial w_{ij}}{\partial r_i} \right) \quad (8)$$

This expression has proven to be highly accurate in particular for the simulations of low Reynolds number planar shear flows [7, 8]. The variable μ denotes the dynamic viscosity of the fluid.

In what concerns equation (2), the pressure gradient terms in the Navier-Stokes (N-S) equations can be formulated using the following *symmetrized* [6] SPH version:

$$-\left(\frac{1}{\rho} \nabla p \right)_i = -\sum_{j \neq i} m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} \right) \nabla_i w_{ij} \quad (9)$$

Therefore, the SPH momentum equation yields

$$\frac{d\mathbf{v}_i}{dt} = -\sum_{j \neq i} m_j \left[\begin{array}{l} \left(\frac{p_j}{\rho_j^2} + \frac{p_i}{\rho_i^2} \right) \nabla_i w_{ij} - \frac{1.0}{\rho_i \rho_j} \\ (\mu_i + \mu_j) \frac{\nabla_i w_{ij} \cdot \mathbf{r}_{ij}}{r_{ij}^2 + 0.01 h^2} \mathbf{v}_{ij} \end{array} \right] + \mathbf{F}_i \quad (10)$$

where, F is the body force per unit mass (m/s^2). The quantity $0.01 h^2$ is used to prevent singularities for $r_{ij} \approx 0$.

Position update

To assure the SPH particles move with a velocity consistent with the average velocity of its neighboring particles it used the variant proposed by Morris [10]

$$\frac{d\mathbf{r}_i}{dt} = \mathbf{v}_i + \varepsilon \sum_j \frac{m_j \mathbf{v}_{ij}}{\bar{\rho}_{ij}} w_{ij} \quad (11)$$

Here, $\bar{\rho}_{ij} = (\rho_i + \rho_j)/2.0$ and the factor ε is chosen as $\varepsilon = 0.5$.

Rate of change of temperature

Cleary et al. [16] derived a SPH heat transfer equation for solidifying metals, which can be rewritten for a fluid of constant heat capacity as:

$$c_p \frac{dT_i}{dt} = \sum_{j \neq i} \frac{\frac{4.0m_j}{\rho_i \rho_j} \frac{\lambda_i \lambda_j}{\lambda_i + \lambda_j} \frac{\mathbf{r}_{ij} \cdot \nabla_i w_{ij}}{r_{ij}^2 + 0.01h^2} T_{ij} - \frac{2.0m_j}{\rho_i \rho_j} \frac{\mu_i \mu_j (\mathbf{v}_{ij} \cdot \mathbf{r}_{ij})^2}{\mu_i + \mu_j} \cdot \frac{\mathbf{r}_{ij} \cdot \nabla_i w_{ij}}{(r_{ij}^2 + 0.01h^2)^2}}{(12)}$$

where, λ and c_p are the heat conductivity and the specific heat at constant pressure, respectively. Due to the incompressibility of the fluid, pressure work was neglected and the second term on the right-hand side denotes the contribution from viscous heat dissipation.

Time advancement

The 4th order Runge–Kutta integration algorithm is used to perform the integrations of equations (4), (10), (11), and (12). The time step Δt is adaptive and determined by the Courant–Friedrichs–Lewy (CFL) condition [8] given by:

$$\Delta t = \min \left(\frac{c_1 h}{\max_i (15.0U, |\mathbf{v}_i|)}, c_2 \sqrt{\frac{h}{\max \text{ acceleration}}}, \frac{c_3 h}{\max \text{ sound speed}} \right) \quad (13)$$

The Courant “safety” factors c_1 , c_2 , and c_3 are given as $c_1 = 0.125$, $c_2 = 0.25$, and $c_3 = 0.25$, respectively. In the right-hand side of equation (13), a quantity $15.0U$ is introduced in the first constraint so that the three constraints may have similar influence upon the calculation.

Smoothing kernel

High accuracy three-splines quintic kernel is used

$$w_{ij} = n_d \frac{1.0}{h^\beta} \begin{cases} (3.0 - s)^5 - 6.0 (2.0 - s)^5 + 15.0 (1.0 - s)^5 & \text{if } 0 \leq s < 1.0 \\ (3.0 - s)^5 - 6.0 (2.0 - s)^5 & \text{if } 1.0 \leq s < 2.0 \\ (3.0 - s)^5 & \text{if } 2.0 \leq s < 3.0 \\ 0 & \text{if } s \geq 3.0 \end{cases} \quad (14)$$

where, $s = |\mathbf{r}_{ij}|/h$, β is the dimensionality of the problem, and n_d a normalization constant, which is determined by the normalization property [15] of the smoothing kernel

$$\int w(|\mathbf{r} - \mathbf{r}'|, h) d\mathbf{r}' = 1.0 \quad (15)$$

For the present implementation, $\beta = 2$ and $n_d = 7.0/478.0\pi$.

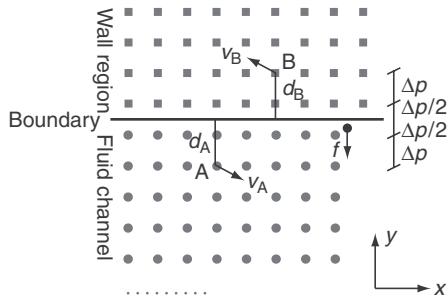


Figure 10.1. Illustration of initial particle arrangement and treatment of solid wall boundary.

The smoothing length h is given as:

$$h = 1.001 \times \sqrt{(\Delta x)^2 + (\Delta y)^2} \quad (16)$$

where, Δx and Δy are the initial spacing in the x and y direction, respectively, between two neighboring particles, and $\Delta x = \Delta y$. Thus, the interpolation of the field quantity with respect to one fluid particle extends to a region where about 61 neighboring particles are included.

Solid wall boundary

The treatment of solid wall boundaries is illustrated in Figure 10.1. Taking into account the influence region of the employed quintic smoothing kernel, four layers of ‘dumb’ wall particles are arranged outside of the fluid channel and parallel to the real wall boundary with the first layer $\Delta p/2$ away from the wall plane. These ‘dumb’ particles have the same spatial separation Δp as fluid particles. The density of a ‘dumb’ particle is initially set equal to the reference fluid density, ρ_0 , and it remains unchanged. Boundary particles interact with the fluid particles by contributing to their density variations, and by prescribing viscous forces on the nearby fluid particles.

To ensure the non-penetrating condition of solid wall boundary, the calculations are made with boundary particles exerting central forces on fluid particles [17]. For a pair of boundary and fluid particle separated by a distance r , the force per unit mass $f(r)$ is assumed to have the 12-6 Lennard–Jones form [18]:

$$f(r) = D \left[\left(\frac{r_0}{r} \right)^{12} - \left(\frac{r_0}{r} \right)^6 \right] \frac{r}{r^2} \quad (17)$$

$f(r)$ is set to zero if $r > r_0$ so that the force is purely repulsive. r_0 is taken to be $0.45\Delta p$. During the calculations, only the component of $f(r)$ normal to the solid wall is considered; D has the units of $(\text{m/s})^2$ and is taken as $10.0F \times \text{Characteristic-Geometric-Length}$. Varying the magnitude of D has no particular influence on the calculation results.

Nonslip wall boundary condition is guaranteed by using the method proposed by Morris et al. [8]. For each fluid particle A, the normal distance d_A to the wall is

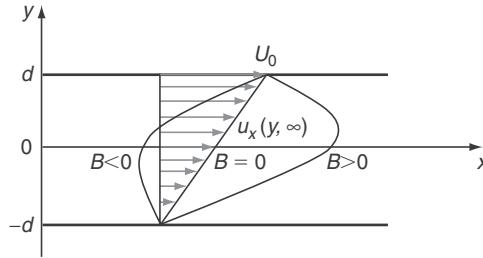


Figure 10.2. Combined Poiseuille and Couette flow.

calculated and for each boundary particle B , d_B is obtained. Then an artificial velocity $\mathbf{v}_B = -(d_B/d_A)\mathbf{v}_A$ is applied to the boundary particle B with an assumption of zero velocity condition on the boundary plane itself. This artificial velocity \mathbf{v}_B is used to calculate the viscous force applied on the nearby fluid particles, but not used to locate the boundary particle position. Boundary particles are motionless or moving at specified velocities. The relative velocity between fluid and boundary particles is:

$$\mathbf{v}_{AB} = \mathbf{v}_A - \mathbf{v}_B = \chi \mathbf{v}_A \quad (18)$$

where

$$\chi = \min \left(\chi_{\max}, 1.0 + \frac{d_B}{d_A} \right) \quad (19)$$

χ_{\max} is used to exclude the extremely large values when fluid particles get too close to the wall. In Morris et al. [8] good results were obtained for $\chi_{\max} = 1.5$ when simulating low Reynolds number planar shear flows. In the present study, $\chi_{\max} = 7.0$ is specified, which is the largest value of χ for the initial regular particle arrangement.

10.2.2 Benchmarking

The validity of the above-outlined SPH formulation is examined for two benchmarks dealing with heat and fluid flow, namely: combined transient Poiseuille and Couette flow, and steady heat convection for combined Poiseuille and Couette flow conditions; these benchmarks were selected because analytical solutions for them are available.

Transient combined Poiseuille and Couette flow

This test case is the fluid flow between two parallel infinite plates, located at $y = -d$ and $y = d$, respectively, as shown in Figure 10.2. The system is initially at rest. At time $t = 0$, the upper plate starts to move at constant velocity U_0 along the positive x -axis. Concurrently, a body force F is aligned parallel to the x -axis. The resultant velocity profile is dependent on a non-dimensional quantity, designated as B

$$B = \frac{d^2}{\nu U_0} F \quad (20)$$

and its time-dependent series solution is:

$$\begin{aligned} u_x(y, t) = & \frac{F}{2\nu}(d^2 - y^2) + \frac{U_0}{2d}(y + d) \\ & - \sum_{n=0}^{\infty} (-1)^n \frac{16Fd^2}{\nu\pi^3(2n+1)^3} \cos\left(\frac{\pi y}{2d}(2n+1)\right) \exp\left(-\frac{(2n+1)^2\pi^2\nu}{4d^2}t\right) \\ & + \sum_{n=1}^{\infty} (-1)^n \frac{2U_0}{n\pi} \sin\left(\frac{n\pi}{2d}(y+d)\right) \exp\left(-\frac{n^2\pi^2\nu}{4d^2}t\right) \end{aligned} \quad (21)$$

The first and third terms in the right-hand side of equation (21) refer to the contribution by the applied body force F and the other two terms result from the motion of the upper plate. When time t approaches infinite, the steady-state solution takes the form:

$$u_x(y, \infty) = \frac{B}{2}U_0\left(1 - \frac{y^2}{d^2}\right) + \frac{1}{2}U_0\left(1 + \frac{y}{d}\right) \quad (22)$$

The physical properties of the fluid are specified as, $\nu = 1.0 \times 10^{-6}$ m²/s and $\rho = 1000.0$ kg/m³. The value of d is taken as 0.0005 m and $U_0 = 5.0 \times 10^{-5}$ m/s. Fifty SPH fluid particles are set to span the channel. The body force F is adjusted to give different values of B . The results are presented in Figures 10.3a and b.

The overall agreement between simulations and exact analytical solutions is excellent: the largest deviation between SPH simulation and series solution is observed at the position close to the motionless wall, and it is less than 2.0%.

Steady heat convection for combined Poiseuille and Couette flow conditions

The second test case is conducted on the steady convective heat transfer between two parallel infinite plates, which are kept at constant temperatures: T_e and T_w ($T_e > T_w$), respectively. The upper plate is moving at a constant velocity U_0 yielding the Couette flow condition, as shown in Figure 10.4. Concurrently, a body force F may be applied too.

The exact solution for the steady-state temperature profile is given as:

$$T^*(y) = \frac{1}{2}(1 + y^*) + \frac{Br}{8}(1 - y^{*2}) - \frac{Br}{6}B(y^* - y^{*3}) + \frac{Br}{12}B^2(1 - y^{*4}) \quad (23)$$

where, Br stands for the Brinkman number which is the product of the Prandtl number Pr by the Eckert number Ec ($Br = Pr.Ec$), y^* and T^* are the normalized position and temperature, respectively, as defined below:

$$Br = \frac{\mu c_p}{\lambda} \frac{U_0^2}{c_p(T_e - T_w)} \quad (24)$$

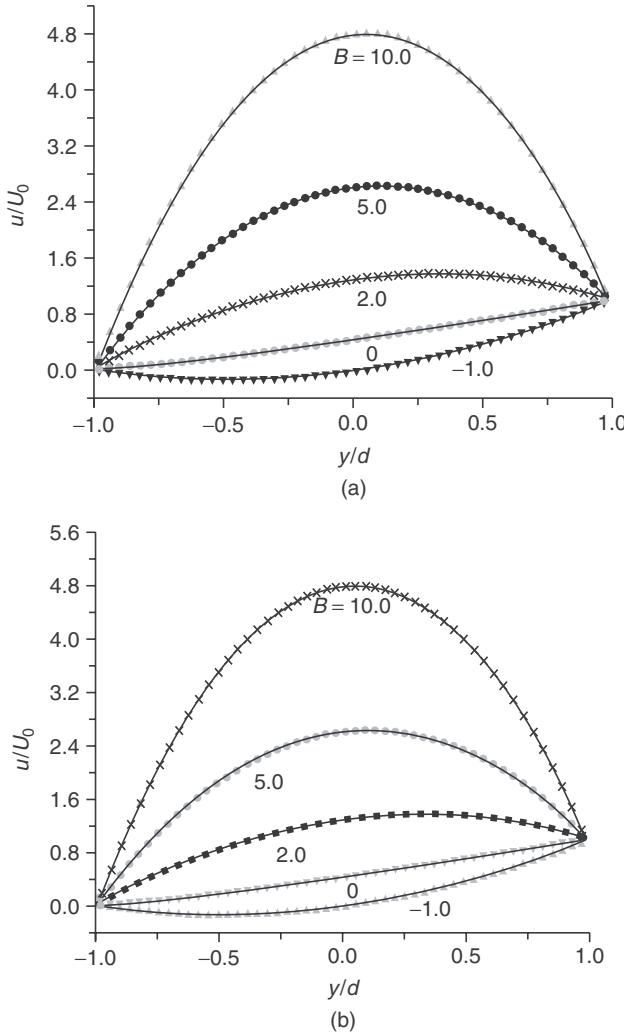


Figure 10.3. Results for combined Poiseuille and Couette flow at: (a) time = 0.212 seconds, and (b) time approaching infinite. (Symbols represent the SPH predictions and the full lines the analytical solutions).

$$y^* = \frac{y}{d} \quad (25)$$

$$T^*(y) = \frac{T(y) - T_w}{T_e - T_w} \quad (26)$$

The parameter B is the ratio between the contributions from the body force to the fluid flow and the motion of upper plate, as expressed in equation (20). The first term in the right-hand side of equation (23) is due to the heat conduction between the two

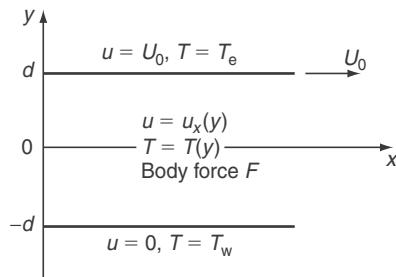


Figure 10.4. Steady heat convection for combined Poiseuille and Couette flow condition.

parallel plates, whereas the last three terms describe the viscous heat dissipation, which for moderate values of B and small values of Br is negligible; however, for high viscosity fluids, as expected, may be significant. In the following calculations, to test the ability of the SPH methodology in predicting heat dissipation, the fluid is assumed to have a relatively large viscosity of 1.0 Pa·s. Moreover, the geometric dimension is increased to $d = 0.05$ m and the moving velocity of the upper plate is $U_0 = 0.05$ m/s. Other parameters specified are: heat conductivity, $\lambda = 0.1$ W/m·K; heat capacity, $c_p = 4,200$ J/kg·K; density, $\rho = 1,000.0$ kg/m³; and the temperature of the lower plate, $T_w = 0^\circ\text{C}$. By varying the magnitude of F and the upper plate temperature T_e , different B and Br values are obtained.

As before, 50 SPH fluid particles are set to span the channel. The results for this particular case are depicted in Figures 10.5a and b, and, as observed, the SPH predictions for the non-dimensional temperature compare well with the analytical solution; the maximum deviation is less than 1%.

The heat convection process and the heat dissipation characteristics are also reproduced well by the SPH simulations.

10.3 Control of the Onset of Turbulence in MHD Fluid Flow

Magnetohydrodynamic (MHD) flow control finds growing importance in many industrial settings, in particular in the metallurgical industry. The electromagnetic force imposed by a stationary magnetic field may suppress the flow instabilities of electrically conducting fluids and creates favorable flow conditions in fluid flow devices, such as tundishes, as described by Szekley and Illegbusi [19]. The work reported by Hartmann and Lazarus [20] was the first one to experimentally identify the change in drag and the suppression of turbulence when a magnetic field is applied to turbulent liquid metal flows. From then on, numerous studies have addressed the interactions between the applied transverse (wall-normal) magnetic field and the fluid flow, as reported in Lee and Choi [21], Krasnov et al. [22], Ji and Gardner [23]. A magnetic field applied in the transverse flow direction is effective in reducing the turbulence fluctuations and suppressing the near-wall streamwise vorticity; the mean velocity profile is modified, therefore yielding a drag increase. In contrast to

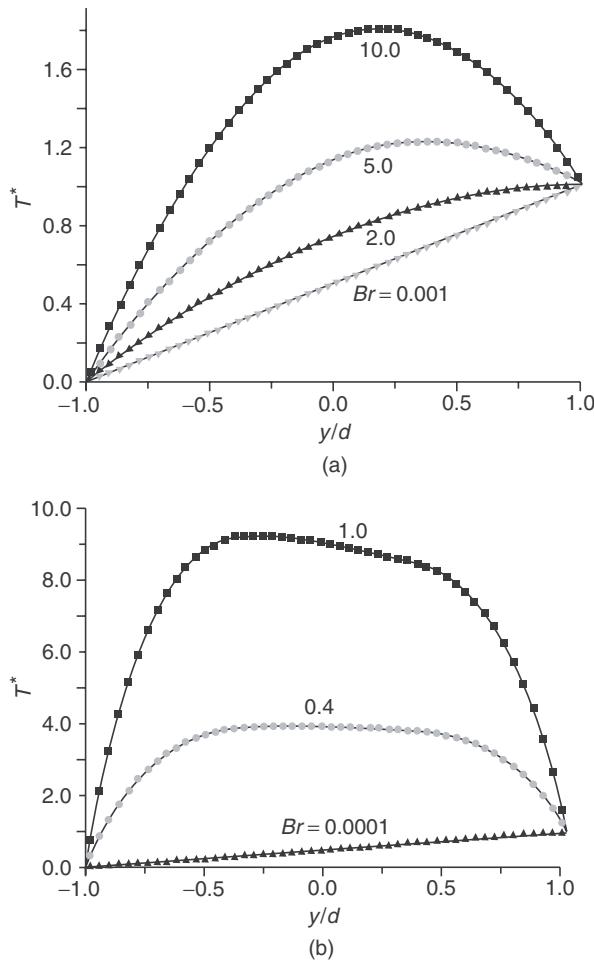


Figure 10.5. Results for temperature profiles when (a) $B = 0$ and (b) $B = 10.0$. (Symbols represent the SPH predictions and the full lines the analytical solutions).

the extensive research effort dealing with MHD flows with an applied transverse magnetic field, little attention was given to the flows interacting with a streamwise magnetic field. The application of a streamwise magnetic field, to a flow, however, is able to restrain the velocity fluctuations in the transverse plane of that flow and the transition to turbulence can be distinctly delayed, or even suppressed in the flow. The ability of a streamwise magnetic field of controlling the onset of turbulence in an electrically conducting fluid flow was recently corroborated by Jiang et al. [24]. To achieve the most effective flow control with the least amount of energy consumption is the goal of an MHD flow control strategy, which can be realized by combining the two above-mentioned magnetic fields.

10.3.1 MHD flow control

Motion of electrically conducting fluids across a magnetic field induces the so-called Lorentz force, which, when properly handled, may suppress the flow instability and restrain the flow turbulence. A magnetic field applied in the transverse (wall-normal) direction of a fluid flow creates a ponderomotive body force and changes the mean velocity profile due to the formation of the Hartmann layers at electrically insulating boundaries normal to the magnetic field. A magnetic field applied to the flow in the streamwise direction restrains the velocity fluctuations in the transverse plane of the fluid flow and prevents the onset of turbulence. In comparison to a streamwise magnetic field, a magnetic field applied in the transverse direction of the flow acts directly on the main flow and may prove to be more flexible in controlling the flow; however, it increases the flow resistance and needs additional energy consumption. Therefore, a reasonable control strategy may be realized by bringing an angle between the magnetic field and the flow direction within the range of the two above-mentioned magnetic supply conditions, i.e., 0° (streamwise) and 90° (transverse). In this way, both velocity fluctuations in the transverse and streamwise direction are restrained concurrently and a more effective control effect may be achieved with some advantage in what concerns external energy usage.

The transition to turbulence of electrically conducting fluid flow in a two-dimensional (2-D) channel constrained by two parallel infinite plates, which are assumed electrically insulated, as depicted in Figure 10.6, is chosen to illustrate the underlying concept. Fluid enters the channel with a uniform velocity U_0 , initially.

For a Reynolds number exceeding a critical value ($\approx 2,000$), the turbulence, after a short entry length, sets in. For the sake of restraining the transition to turbulence, a uniform magnetic field \mathbf{B} is applied at an angle θ to the main fluid flow direction. The present study, by adjusting the angle θ , aims to achieve the best restraining effect. To compare the results obtained for different values of θ , a body force F_B is prescribed in the x direction to balance the produced ponderomotive electromagnetic force so as to maintain the maximum flow velocity ($\approx U_0$) approximately equal for the cases of interest. The restraining effect is quantitatively evaluated by the time when the turbulence sets in, or by measuring the magnitude of the “turbulence-related” quantities – the mean velocity \bar{u} and the two velocity correlations in the x and y direction, i.e., $\bar{u' u'}$ and $\bar{v' v'}$ at a specified time.

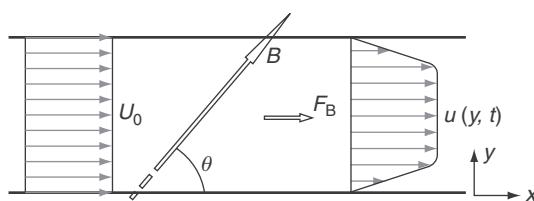


Figure 10.6. MHD flow control for 2-D channel flow.

10.3.2 MHD modeling

The MHD flow is governed by a set of coupled partial differential equations that express the conservation of mass, momentum and the interaction between the flow and the applied magnetic field. In the present work, it is assumed that the hydrodynamic Reynolds number Re is much larger than the magnetic Reynolds number Re_m , namely

$$\text{Re} = \frac{UL}{\nu} \gg \text{Re}_m = \frac{U_m L_m}{\eta} \quad (27)$$

Here, $\eta = (\sigma \mu_0)^{-1}$, where ν , σ , and μ_0 are the kinematic viscosity, electrical conductivity, and magnetic permeability of free space ($\mu_0 = 4\pi \times 10^{-7}$ H/m), respectively; U and L are the characteristic velocity and length of the fluid flow, respectively; the subscript m denotes the corresponding magnetic quantities. This assumption guarantees the fluctuations of the magnetic field due to the fluid motion are very small as compared to the applied field B ; therefore, the total magnetic field can be taken as uniform and time-independent, and the induced electrical current \mathbf{j} is given by:

$$\mathbf{j} = \sigma(\nabla\phi + \mathbf{V} \times \mathbf{B}) \quad (28)$$

where, \mathbf{V} is the fluid velocity vector and ϕ is the induced electrical potential, which is produced by the interaction between the applied magnetic field, and the flow vorticity \mathbf{w} ($= \nabla \times \mathbf{V}$). The present study does not aim to investigate the turbulence pattern or structure of MHD turbulent flows [21–23], of interest here is to investigate the time duration prior to the onset of turbulence of the flow. The vorticity itself, \mathbf{w} , is considered to be negligible, and the induced electrical potential is assumed to be zero, which yields [25]

$$\mathbf{j} = \sigma(\mathbf{V} \times \mathbf{B}) \quad (29)$$

The Lorentz force defined per unit of volume, \mathbf{F} takes the form

$$\mathbf{F} = \mathbf{j} \times \mathbf{B} = \sigma(\mathbf{V} \times \mathbf{B}) \times \mathbf{B} \quad (30)$$

Accordingly, the MHD equations for the above-mentioned transient 2-D ($x - y$) planar channel flow can be written as follows:

$$\frac{D\rho}{Dt} = -\rho \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) \quad (31)$$

$$\rho \frac{Du}{Dt} = -\frac{\partial p}{\partial x} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) - \sigma(B \sin \theta)^2 u + F_B \quad (32)$$

$$\rho \frac{Dv}{Dt} = -\frac{\partial p}{\partial y} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) - \sigma(B \cos \theta)^2 v \quad (33)$$

where, u and v denote the velocity components in the streamwise (x) and transverse (y) direction, respectively. In the present study, u is the total velocity in the x direction: $u = \bar{u} + u'$, and v is the velocity fluctuation in the y direction, i.e., $v = v'$. ρ , and μ stand for the density and dynamic viscosity, respectively. D/Dt is the material derivative. F_B (N/m^3) is the applied body force, which is used to balance the electromagnetic force in the x direction, and is given by:

$$F_B = \sigma(B \sin \theta)^2 U_0 \quad (34)$$

10.3.3 SPH analysis of magnetic conditions to restrain the transition to turbulence

SPH simulations are performed with respect to the channel flow configuration shown in Figure 10.6. The separation distance between the two parallel infinite plates (L) is taken as 1.0 m. The computational domain is a unit square where solid boundaries confine the flow in the transverse direction (y) and periodic boundaries are specified in the flow direction (x). The simulation is started with the fluid moving to the right (positive x) at a velocity, $U_0 = 1.0$ m/s. The Reynolds number is defined as $Re = UL/\nu$, with U , L being the characteristic velocity (1.0 m/s) and length (1.0 m), respectively, and the kinematic viscosity, ν , is adjusted to give different Reynolds numbers. The initial arrangement of the SPH particles and treatment of solid wall boundaries were already illustrated in Figure 10.1. A 50 square lattice of fluid particles is used.

The intensity of the electromagnetic force can be quantified with a dimensionless quantity, namely, Stuart number St , which is defined as:

$$St = Re_m Al = \frac{\sigma B^2 L_m}{\rho U_m} \quad (35)$$

where, Al denotes the Alfvén number. The Stuart number St designates the ratio of the electromagnetic force to the inertial force. It is assumed that the magnetic characteristic velocity U_m and length L_m have the same magnitudes of their hydrodynamic counterparts, i.e., 1.0 m/s for U_m and 1.0 m for L_m . The magnitude of the applied magnetic field B is adjusted to give different values of St .

Transitional organization of the SPH fluid particles versus the onset of turbulence

For SPH simulations, the onset of turbulence is accompanied by an *ordered-disordered* transition of the SPH fluid particles [14, 24], so the transitional organization of the SPH fluid particles indicates the time when the onset of turbulence takes place. For fluid flow of $Re = 10^4$ and $St = 40$ with the magnetic field aligned along the streamwise flow direction, i.e., $\theta = 0^\circ$, at the time of 1.89 seconds the SPH particles display an ordered flow pattern, which takes on a plug flow profile with very thin boundary layers. At 1.95 seconds the transitional organization of SPH fluid particles begins to appear, which indicates that the flow turbulence sets

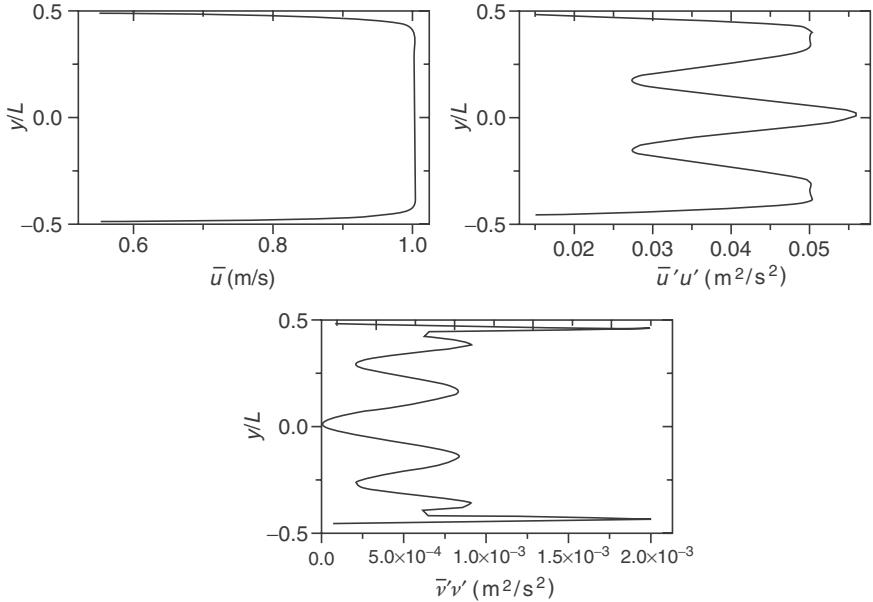


Figure 10.7. Representation of \bar{u} , $\bar{u}'u'$, and $\bar{v}'v'$ for MHD channel flow of $Re = 10^4$, $St = 40$, and $\theta = 0^\circ$ at time ~ 1.9 seconds.

in. The calculation is terminated at this point because the ensuing MHD turbulence is of no immediate interest to the present study.

Further insight into the flow state can be achieved by examining the turbulence-related quantities, the mean velocity \bar{u} and the velocity correlations $\bar{u}'u'$ and $\bar{v}'v'$. This is done by dividing the flow domain into 50 horizontal strips and by using a smaller time step to redo the calculations around a specified instant of time, afterwards the results are averaged both spatially and temporally over each sub-region. Shortly before the onset of turbulence is detected, at time ~ 1.9 seconds, the turbulence-related quantities of the above-described MHD fluid flow are shown in Figure 10.7. The flow takes on some turbulent flow characteristics, namely, the mean velocity profile is approaching to that of plug flow with the occurrence of very thin boundary layers.

Enhanced effect of the tilting angle θ

The MHD flow to be considered in this section has $Re = 10^4$ and $St = 40$. The tilting angle θ of the applied magnetic field \mathbf{B} to the main fluid flow direction is varied to examine the acquired restraining effect to the transition to turbulence. When $\theta = 0^\circ$, i.e., a streamwise magnetic field is applied, the flow turbulence takes place at time ~ 1.95 seconds. The turbulence-related quantities at time ~ 1.9 seconds, as shown in Figure 10.7, reveal the velocity correlations in the y direction take smaller values than those in the x direction: the values of $\bar{u}'u'$ are in the range of 0.015 to 0.06 m 2 /s 2

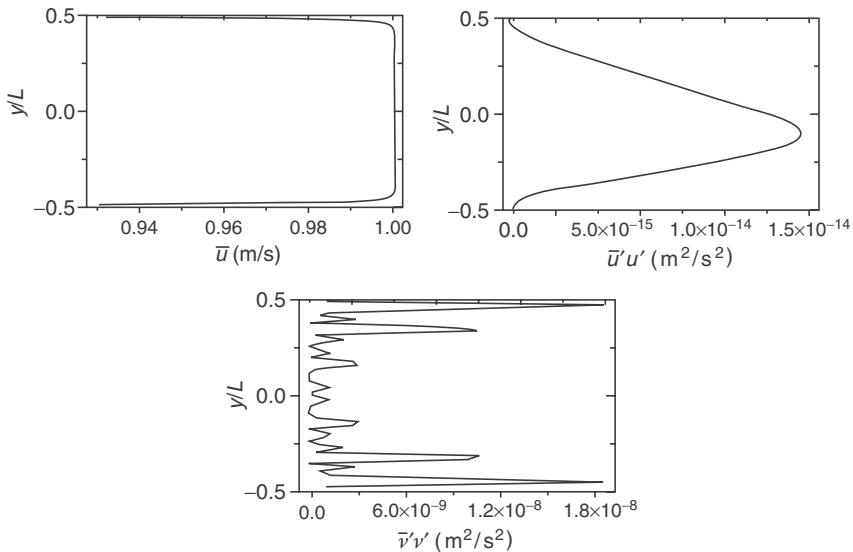


Figure 10.8. Representation of \bar{u} , \bar{u}' , and $\bar{v}'v'$ for MHD channel flow of $Re = 10^4$, $St = 40$, and $\theta = 20^\circ$ at time ~ 5.24 seconds.

while $v'v'$ are within 0 to $0.002\text{ m}^2/\text{s}^2$, which implies that the velocity fluctuations in the transverse plane of the flow are restrained by the applied streamwise magnetic field. The largest velocity fluctuation in the y direction is detected to occur in the region close to the walls, whereas that in the x direction takes place in the central region of the fluid flow. This indicates the velocity fluctuation in the x direction is initiated from the large bulk flow velocity, while the one in the y direction is caused by the interactions between the fluid flow and the boundary walls. It is the large velocity fluctuations freely developing in the x direction that cause the velocity fluctuation in the y direction and yield the transitional organization of SPH particles. An effective MHD method to restrain the transition to turbulence is enhanced when the applied magnetic field is capable of controlling the flow concurrently in the x and y direction.

At an angle of $\theta = 20^\circ$, plug flow with very thin boundary layers develops gradually; although the displacement in the streamwise direction between particles in the boundary layers and those in the main flow is cumulative with time, the flow is laminated and the SPH particles remain ordered till the end of the simulation ($t = 6.79$ seconds). The corresponding turbulence-related quantities for this condition at time ~ 5.24 seconds are displayed in Figure 10.8. The representation of the mean velocity profile takes on a nearly perfect plug flow: even for the fluid particles close to the wall boundaries, their velocities approach the bulk velocity, 1.0 m/s. Compared to those shown in Figure 10.7, both the velocity fluctuations in the x and y direction are restrained, even when the time is extended by approximately 3.3 seconds beyond that presented in Figure 10.7; for these conditions the

fluctuations are at a much lower level, namely: $u'v'$ is within 0 to $1.5 \times 10^{-14} \text{ m}^2/\text{s}^2$ and $v'v'$ within 0 to $2.0 \times 10^{-8} \text{ m}^2 \text{ s}^2$. The velocity fluctuation in either the x or y direction gets effectively damped, and although the electromagnetic force prescribed in the y direction diminishes, the restraining effect exerted on the velocity fluctuations in this direction seems to become stronger, not weaker. This is seemingly unreasonable; however, it may be explained on the basis of what is observed in Figure 10.7: the velocity fluctuations in the x direction are not constrained by the applied streamwise magnetic field, thus they have very large values; these velocity fluctuations in the x direction transfer “turbulent” energy to the velocity fluctuations in the y direction, which, in turn, also acquire large values. This supports the viewpoint advanced above: an effective MHD method for restraining the transition to turbulence needs to suppress simultaneously the velocity fluctuations in the x and y direction.

Ten SPH simulations were performed for $\theta = 0, 10, 20, 30, 40, 50, 60, 70, 80,$ and 90° , respectively. All the results are listed in Table 10.1. An applied streamwise magnetic field ($\theta = 0^\circ$) is more effective in restraining the velocity fluctuations in the y direction, but it has no influence on the velocity fluctuations in the x direction.

Table 10.1. Restraining effect upon the transition to turbulence by the tilting angle θ ($\text{Re} = 10^4$, $\text{St} = 40$)

θ°	Onset of turbulence at time equal to (seconds)	Turbulence-related quantities		
		\bar{u}^b	$u'v'$	$v'v'$
^c 90	0.96	0.991	$0\text{--}4.3 \times 10^{-11}$	0–0.068
^c 80	1.02	0.99	$0\text{--}1.1 \times 10^{-11}$	0–0.014
^c 70	1.23	0.989	$0\text{--}4.2 \times 10^{-12}$	0–0.011
^c 60	1.85	0.988	$0\text{--}1.1 \times 10^{-12}$	0–0.006
^a 50	–	0.985	$0\text{--}5.4 \times 10^{-17}$	$0\text{--}9.2 \times 10^{-9}$
^a 40	–	0.978	$0\text{--}5.1 \times 10^{-18}$	$0\text{--}1.2 \times 10^{-8}$
^a 30	–	0.965	$0\text{--}5.7 \times 10^{-18}$	$0\text{--}2.2 \times 10^{-8}$
^a 20	–	0.93	$0\text{--}1.5 \times 10^{-14}$	$0\text{--}2.0 \times 10^{-8}$
^a 10	–	0.78	$0\text{--}8.0 \times 10^{-7}$	$0\text{--}3.5 \times 10^{-8}$
^c 0	1.95	0.55	0.015–0.06	$0\text{--}2.1 \times 10^{-3}$

^a Up to the end of simulation time, 6.79 seconds, turbulence is practically not detected. The turbulence-related quantities are calculated at time ~ 5.24 seconds.

^b In the central region of the fluid flow, the average velocity is about 1.0 m/s. The datum shown here is the velocity of the fluid particle closest to the wall.

^c The turbulence-related quantities are calculated shortly before the onset of turbulence. For instance, the turbulence quantities shown in the last row are calculated around at 1.90 seconds.

It is the fast growth of the velocity fluctuations in the x direction that leads to the early turbulence onset (1.95 seconds). With increasing θ ($\theta = 10$ to 50°), the mean flow velocity profile approaches the plug flow more closely; the velocity fluctuations in the x direction are effectively restrained due to the increasing electromagnetic force produced in this direction; although the y -component of the electromagnetic force decreases, the restraining effect exerted on the velocity fluctuations in the y direction benefits from the effective constraint of its counterparts in the x direction. Further increase of θ ($\theta = 60$ to 80°) yields the renewed appearance of turbulence, largely due to the increase of the velocity fluctuations in the y direction. An applied transverse magnetic field ($\theta = 90^\circ$) is more effective in controlling the mean flow profile, but it does not restrain well the transition to turbulence. Turbulence occurs at an earlier time (0.96 seconds), when the applied magnetic field is aligned with the transverse direction.

Enhanced conditioning effect of the transition to turbulence is obtained when the oblique angle θ is within the range 20 to 50° . In this situation, the mean flow velocity profile is close to a perfect plug flow; the velocity fluctuations in both the x and y direction are kept at very low levels, namely: $u' u'$ is below $1.5 \times 10^{-14} \text{ m}^2/\text{s}^2$ and $v' v'$ less than $2.2 \times 10^{-8} \text{ m}^2/\text{s}^2$; and, effectively, the onset of flow turbulence does not take place up to the end of the simulation time (6.79 seconds).

SPH simulations were also conducted with a magnetic field, B , increased by a factor of 10 to give a Stuart number, St, of 4,000 for the MHD flow of $\text{Re} = 10^4$ with the θ varying in the range of 0 to 90° . For the range $10^\circ \leq \theta \leq 80^\circ$ the transition to turbulence gets effectively suppressed; even so, for the magnetic supply conditions of $\theta = 0$ and 90° with this increased magnetic field applied, the flow turbulence still appears, which supports the viewpoint advanced above; an effective MHD method for restraining the transition to turbulence requires a strategy that restrains all velocity fluctuation components simultaneously.

10.4 SPH Numerical Modeling for Ballistic-Diffusive Heat Conduction

Design of nanoscale systems, such as semiconductor devices based on the GaAs MESFETs or Si MOSFETs, and ultrafast (picoseconds or even femtoseconds) pulsed lasers have forced a fresh look into fundamental heat transfer issues [26]. At nanoscale level, the classical Fourier heat diffusion is not valid due to: (a) the mean free path of the energy carriers becomes comparable to or larger than the characteristic length scale of the particular device/system, and/or (b) the time scale of the processes becomes comparable to or smaller than the relaxation time of the energy carriers. Numerical solutions either using the Boltzmann transport equation (BTE) or using atomic-level simulations such as the molecular dynamics simulation (MDS) and Monte Carlo simulation (MCS) are helpful in understanding the physics of heat transfer in this regime; however, these options require large computational resources, which make them not very effective analytical tools for the design/management of devices in practical nanoscale engineering.

The Cattaneo-Vernotte (C-V) hyperbolic equation [27], for a while, seemed to be a convenient alternative approach, since it excludes the so-called infinite thermal propagation speed assumption implied in Fourier law. The work reported by Haji-Sheikh et al. [28] was the first one to point out certain anomalies existing in the hyperbolic solutions. Further on, recent work by Jiang and Sousa [29] also demonstrated the presence of hyperbolic anomalies in a hollow sphere with the Dirichlet boundary condition prescribed. Although may exist some remedies [28] to the C-V equation, a newly developed heat conduction formulation, known as the ballistic-diffusive equation [30, 31], seems to be a viable candidate for applications in this particular field.

The key concept of the model, on which the ballistic-diffusive heat conduction equation is formulated, is to split the heat carriers inside the medium into two components – ballistic and diffusive. The ballistic component is determined from the prescribed boundary condition and/or nanoscale heat sources, and it only experiences outscattering; the transport of the scattered and excited heat carriers inside the medium is treated as a diffusive component. This approach has its origin in methodologies used to deal with radiative heat transfer [32, 33].

The ballistic-diffusive equation is derived on the basis of the BTE under the relaxation time approximation; it differs, in appearance, from the C-V equation mainly by having the additional ballistic term, namely

$$\tau \frac{\partial^2 u_m}{\partial t^2} + \frac{\partial u_m}{\partial t} = \nabla \left(\frac{\lambda}{C} \nabla u_m \right) - \nabla \cdot \mathbf{q}_b + \left(S + \tau \frac{\partial S}{\partial t} \right) \quad (36)$$

where, C refers to the volumetric specific heat ($\text{J}/\text{m}^3 \text{ K}$), \mathbf{q}_b to the ballistic heat flux vector (W/m^2), S to the heat generation (W/m^3), t to time (seconds), u_m to the internal energy per unit volume (J/m^3), λ to the heat conductivity ($\text{W}/\text{m K}$), and τ to the thermal relaxation time (seconds). The heat generation S is assumed to have a much larger feature size than the mean free path of the heat carriers; otherwise it should be treated as a ballistic term also. In the original work of Chen [30, 31], the following general boundary condition for the diffusive components is as follows:

$$\tau \frac{\partial u_m}{\partial t} + u_m = \frac{2\Lambda}{3} \nabla u_m \cdot \mathbf{n} \quad (37)$$

where, \mathbf{n} (bold text designates vector) is the inward unit vector perpendicular to the boundary, and Λ is the mean free path of heat carrier (m). The ballistic heat flux q_b is given by an algebraic analytical expression [31, 34]; however, the full solution of equation (36) still has to resort to an appropriate numerical method, in general, the finite difference or finite element method.

The present section aims to show the viability and accuracy of the SPH solution of the one-dimensional (1-D) model proposed by Chen [31]. It should be mentioned that SPH was already employed to solve the Fourier heat conduction equations with satisfying results [3, 4, 35], and a corrective SPH method [35] is used by Chen and Beraun [36] to solve the non-classic heat transfer process in thin metal films heated by ultrashort laser pulse [37].

10.4.1 Transient heat conduction across thin films

In Chen [31] it is reported that the application of the ballistic-diffusive model to a transient heat conduction process in thin films. The same thermal case is considered in the present study: a thin (nanoscale) film of thickness L with constant thermal properties initially at ambient temperature T_0 ; at the time $t \geq 0$, one boundary surface emits phonons at temperature T_1 ; both boundary surfaces are black body emitters and there is no heat generation inside the film. The dimensionless ballistic-diffusive equation and associated initial and boundary restrictions can be written as in Chen [31].

$$\frac{\partial^2 \theta_m}{\partial t^{*2}} + \frac{\partial \theta_m}{\partial t^{*}} = \frac{Kn^2}{3} \frac{\partial^2 \theta_m}{\partial \eta^2} - Kn \frac{\partial q_b^*}{\partial \eta} \quad (38)$$

$$t^* = 0, \quad \theta(\eta, 0) = 0 \left. \frac{\partial \theta(\eta, t^*)}{\partial t^*} \right|_{t^*=0} = 0 \quad (39)$$

$$\eta = 0, \quad \left(\frac{\partial \theta_m}{\partial t^*} + \theta_m \right)_{\eta=0} = \frac{2Kn}{3} \left(\frac{\partial \theta_m}{\partial \eta} \right)_{\eta=0} \quad (40)$$

$$\eta = 0, \quad \theta_{b1} = 1 \quad (41)$$

$$\eta = 1, \quad \left(\frac{\partial \theta_m}{\partial t^*} + \theta_m \right)_{\eta=1} = -\frac{2Kn}{3} \left(\frac{\partial \theta_m}{\partial \eta} \right)_{\eta=1} \quad (42)$$

$$\eta = 1, \quad \theta_{b2} = 0 \quad (43)$$

where, η and θ_m are the dimensionless coordinate and diffusive temperature, respectively.

The boundary heat flux, its derivative, and the temperature of the ballistic components have their algebraic analytical expressions [31] given in non-dimensional form as:

$$q_b^*(\eta, t^*) = \begin{cases} \frac{1}{2} \int_{\mu_t}^1 \mu e^{-(\eta/\mu Kn)} d\mu & (0 \leq \mu_t = \frac{x}{vt} \leq 1) \\ 0 & \text{other } \mu_t \end{cases} \quad (44)$$

$$\frac{\partial q_b^*(\eta, t^*)}{\partial \eta} = \begin{cases} -\frac{1}{2Kn} \int_{\mu_t}^1 e^{-(\eta/\mu Kn)} d\mu + \frac{\eta}{Kn t^{*2}} e^{-t^*} & (0 \leq \mu_t \leq 1) \\ 0 & \text{other } \mu_t \end{cases} \quad (45)$$

$$\theta_b^*(\eta, t^*) = \begin{cases} \frac{1}{2} \int_{\mu_t}^1 e^{-(\eta/\mu Kn)} d\mu & (0 \leq \mu_t \leq 1) \\ 0 & \text{other } \mu_t \end{cases} \quad (46)$$

where, $\mu_t = x/(vt)$, v refers to the carrier group velocity (m/s), x to the coordinate (m), and μ to the directional cosine.

The non-dimensional parameters are defined as follows:

$$\theta_m = \frac{u_m - u_{m0}}{C\Delta T}, \theta_b = \frac{u_b - u_{b0}}{C\Delta T}, q_m^* = \frac{q_m - q_{m0}}{Cv\Delta T}, q_b^* = \frac{q_b - q_{b0}}{Cv\Delta T}, t^* = t/\tau, \eta = x/L \quad (47)$$

with

$$\text{Kn} = \Lambda/L, \quad \Delta T = T_1 - T_0 \quad (48)$$

where, Kn refers to the phonon Knudsen number, L to the thickness of the nanoscale thin film (m), T_0 and T_1 are the initial temperature (K) and the phonon emission temperature at the left boundary (K), respectively.

Thus, the total temperature and heat flux are defined by:

$$\theta = \frac{u - u_0}{C\Delta T} = \frac{T - T_0}{\Delta T}, \quad q^* = \frac{q - q_0}{Cv\Delta T} \quad (49)$$

The diffusive component of the heat flux is governed by the following equation.

$$\frac{\partial \mathbf{q}_m^*}{\partial t^*} + \mathbf{q}_m^* = -\frac{\text{Kn}}{3} \nabla \theta_m \quad (50)$$

where, \mathbf{q}_m^* is the dimensionless diffusive heat flux.

In Chen [31] the above set of equations were solved numerically with the finite difference method (FDM) and the results were compared against those obtained using: (a) the direct solution of the BTE, (b) the Cattaneo hyperbolic heat conduction equation [27], and (c) the classic Fourier heat diffusion equation. Relative to the BTE results, the ballistic-diffusion heat conduction equation gives consistent temperature and heat flux profiles, whereas the Cattaneo and Fourier heat conduction model both lead to erroneous representations of the processes. In the present study, the ballistic-diffusive equation is to be solved with the SPH method.

10.4.2 SPH modeling

In the implementation of the SPH method to solve the heat conduction, the thermal action at time t is represented by a collection of N particles located at a position \mathbf{r}_i , $i = 1, 2, \dots, N$. The “smoothed” value of any field quantity $f(\mathbf{r}, t)$ at a space point \mathbf{r} and at a time t is a weighted sum of all contributions from neighboring particles

$$\langle f(\mathbf{r}, t) \rangle = \sum_{j=1}^N V_j f(\mathbf{r}_j, t) w(|\mathbf{r} - \mathbf{r}_j|, h) \quad (51)$$

where, N refers to the total SPH particle number.

Since the particle density remains constant here, the weight is the volume of SPH particle instead of the quotient of mass and density [9]. The first order derivative of $f(\mathbf{r}, t)$ is written as:

$$\langle \nabla f(\mathbf{r}, t) \rangle = \sum_{j=1}^N V_j f(\mathbf{r}_j, t) \nabla w(|\mathbf{r} - \mathbf{r}_j|, h) \quad (52)$$

Equation (52) is not symmetric; however, by using the following mathematical operator

$$\nabla f = \frac{\nabla (\chi f) - f \nabla \chi}{\chi} \quad (53)$$

where, χ is unity, equation (52) can be rewritten in the following symmetric form:

$$\langle \nabla f(\mathbf{r}, t) \rangle = \sum_{j=1}^N V_j [f(\mathbf{r}, t) - f(\mathbf{r}_j, t)] \nabla w(|\mathbf{r} - \mathbf{r}_j|, h) \quad (54)$$

In terms of equation (54), the “smoothing” formulation of the first-order derivative of the temperature (θ_m) is:

$$\nabla \theta_m(\eta_i, t^*) = \sum_j V_j [\theta_m(\eta_i, t^*) - \theta_m(\eta_j, t^*)] \nabla_i w_{ij} \quad (55)$$

where, ∇_i denotes the gradient taken with respect to the coordinates at particle i

$$\nabla_i w_{ij} = \frac{\boldsymbol{\eta}_{ij}}{|\boldsymbol{\eta}_{ij}|} \frac{\partial w_{ij}}{\partial \eta_i} \quad (56)$$

and $\boldsymbol{\eta}_{ij} = \boldsymbol{\eta}_i - \boldsymbol{\eta}_j$ and $w_{ij} = w(|\boldsymbol{\eta}_i - \boldsymbol{\eta}_j|, h)$.

One of the key issues for the SPH modeling of the ballistic-diffusive heat conduction is how to construct the “smoothing” formulation for the second derivative of the diffusive temperature (θ_m) in equation (36). A formulation, which combines the standard first-order SPH derivative (equation (55)) and the finite difference concept, was first used by Brookshaw [38] and later on by Cleary and Monaghan [3]. This procedure is computationally efficient, since it only involves the first-order derivative of the smoothing kernel. Although it does not conserve the angular momentum accurately, its use in simulations of Fourier heat conduction problems [3] has been very successful. It is employed in the present study and has the following evolution form:

$$\frac{\partial^2 \theta_m(\eta_i, t^*)}{\partial \eta^2} = \sum_j 2V_j [\theta_m(\eta_i, t^*) - \theta_m(\eta_j, t^*)] \frac{1}{|\boldsymbol{\eta}_{ij}|} \frac{\partial w_{ij}}{\partial \eta_i} \quad (57)$$

Five hundred SPH particles are uniformly distributed along the heat transfer path. The separation distance between neighboring particles is Δp ($= L/500$). The

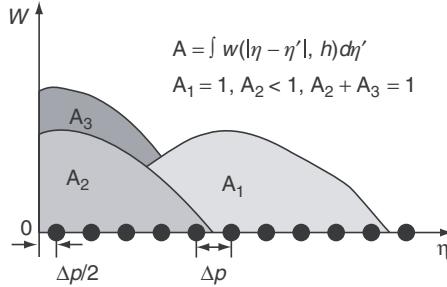


Figure 10.9. Illustration of SPH “near-boundary deficiency” and the boundary treatment devised.

inner-node approach is used, which means that either the first or the last SPH particle is not set on the boundary surfaces, but has half Δp distance away from the respective adjoining boundary. The time derivatives are approximated by the explicit central (for the second-order derivative) or by the forward (for the first-order derivative) scheme. The time step is adjusted to satisfy the numerical stability requirements.

In this implementation, the high order 3-splines quintic kernel is also used.

10.4.3 Boundary treatment

In the present ballistic-diffusive heat conduction case, the temperature specified at the boundary does not represent the equivalent equilibrium temperature of the boundary; the boundary temperature and the boundary flux are calculated from the summation of the corresponding ballistic and diffusive components. Therefore, the temperature values of all the SPH particles depend on the smoothing interpolation. This is the reason why the boundary condition requires special treatment.

On the basis of the properties of the smoothing kernel, a simple boundary treatment is devised, as illustrated in Figure 10.9.

One of the properties the SPH smoothing kernel must have is the normalization property [15], that is

$$\int w(|\eta - \eta'|, h) d\eta' = 1.0 \quad (58)$$

Clearly, for the particles close to the boundary the normalization property is not satisfied, which is the reason for the SPH “near-boundary deficiency”. An expedient boundary treatment can be devised by varying the normalization constant n_d values to meet the requirements set by equation (58) for all the particles. A general form for the SPH smoothing kernel is:

$$w(s, h) = n_d F(s, h) \quad (59)$$

where, $s = |\eta_{ij}|/h$, and F is a fitting function.

By substituting equation (59) into equation (58), the following expression, which is used to adjust the values of the normalization constant of the smoothing function, n_d , yields:

$$n_d = \frac{1.0}{\int F(s, h) d\eta'} \quad (60)$$

Equation (60) is essentially equivalent to the corrective interpolation formulation for a field quantity of zero-th order proposed by Liu et al. [15], which gives for the 1-D 3-spline quintic kernel, n_d [39]:

$$n_d = \frac{1}{120} \times \begin{cases} 1.4632 & \text{1st closest particle to the boundary} \\ 1.0781 & \text{2nd closest particle to the boundary} \\ 1.0054 & \text{3rd closest particle to the boundary} \\ 1.0 & \text{other particles} \end{cases} \quad (61)$$

The particle closest to the boundary gets the largest modification factor, 1.4632, which implies that the “near-boundary deficiency” stays primarily with the closest particle to the boundary.

10.4.4 Results

Figures 10.10 and 10.11 display the non-dimensional diffusive temperature and heat flux distributions in thin films of different thicknesses yielding for Kn values of 10 and 1.0, respectively. The overall agreement between the SPH and FDM calculations is excellent and more accurate predictions can be given by SPH for the heat conduction processes in solid films with smaller values of Kn.

The distributions for the total (summation of the ballistic and diffusive components) temperature and heat flux were also calculated; the ballistic component dominates and due to its analytical formulation yields enhanced agreement between the SPH and FDM results, as shown in Figure 10.12.

As seen from Figure 10.12, the calculated boundary temperature is not constant, but increases as time advances. In Chen [31] this artificial temperature jump is attributed to the inconsistency between the temperatures obtained in the ballistic-diffusive calculation and the imposed boundary temperature. As mentioned previously, the imposed boundary temperature is the temperature of the emitted phonon, not the equivalent equilibrium temperature at the boundary; however, the calculated temperature is the temperature of equilibrium phonons that can be understood as a measure of the local internal energy. The so-called equivalent equilibrium concept can be visualized as the adiabatic thermalization equilibrium state of the local phonons. In what concerns the SPH calculation, the temperature of the particle closest to the boundary cannot be determined directly from the specified temperature of the emitted phonons at the boundary, but depends on the SPH smoothing interpolation. This fact explains why further treatment of the boundary

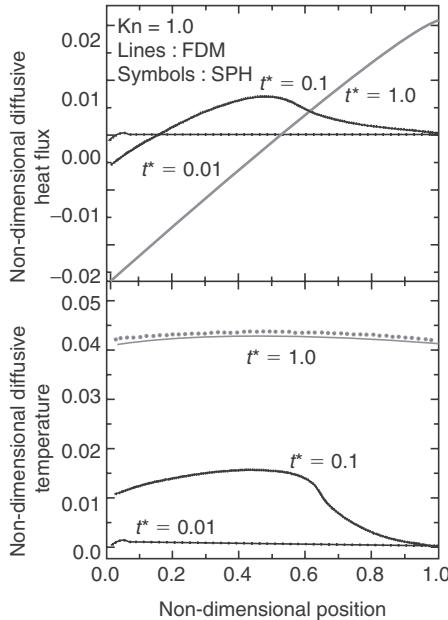


Figure 10.10. Comparison between the SPH and the FD predictions for non-dimensional diffusive temperature and heat flux with $\text{Kn} = 10$.

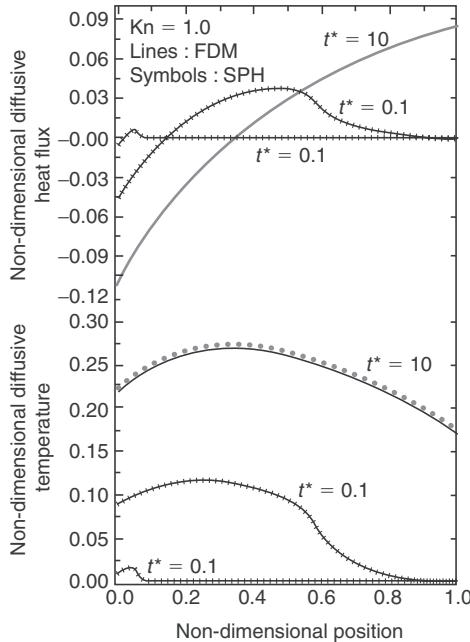


Figure 10.11. Comparison between the SPH and the FD predictions for non-dimensional diffusive temperature and heat flux with $\text{Kn} = 1.0$.

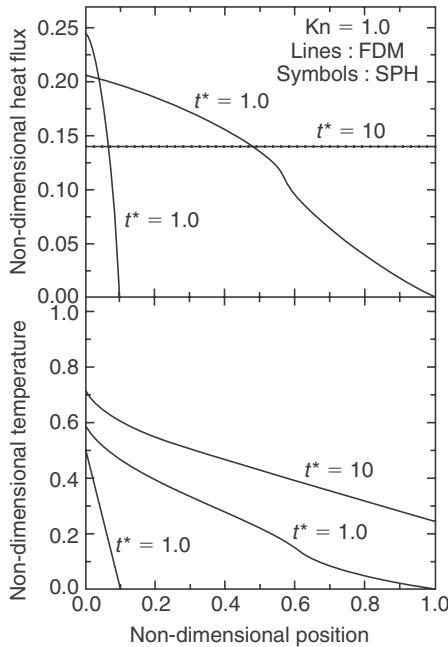


Figure 10.12. Comparison between the SPH and the FD predictions for non-dimensional total temperature and heat flux with $\text{Kn} = 1.0$.

is required. Figure 10.13 displays the contributions of the ballistic and diffusive components to the total temperature (internal energy) and heat flux. The SPH calculations reproduce the characteristics of the ballistic-diffusive heat conduction reported in Chen [31].

Close observation of the calculation source data for all the 500 SPH particles reveals an anomaly for the SPH heat flux result for the second particle away from the boundary. The heat flux of this particle is inconsistent with the overall trend of the heat flux profile and clearly deviates from the FDM predictions. This particle was not shown in Figure 10.10 due to the limited amount of particles (only 50) selected. The implementation of equation (61) can affect the symmetry of the SPH formations, in particular that given by equation (55), which may explain the cause for this anomaly. A test simulation demonstrates a possible remedy for this anomaly is the use of a *ad hoc* operation: artificially increase the n_d of the second particle close to the boundary to 1.234 (mid value of the n_d -s for the first and third particle away from the boundary), which yields [39]

$$n_d = \frac{1}{120} \times \begin{cases} 1.4632 & \text{1st closest particle to the boundary} \\ 1.234 & \text{2nd closest particle to the boundary} \\ 1.0054 & \text{3rd closest particle to the boundary} \\ 1.0 & \text{other particles} \end{cases} \quad (62)$$

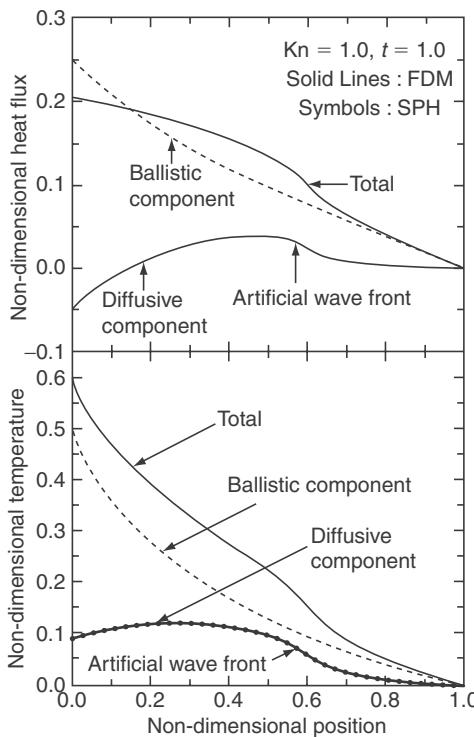


Figure 10.13. Contribution of the ballistic and diffusive components to the non-dimensional total temperature and heat flux.

Increasing the n_d value of the second closest particle to the boundary improves the symmetry contributions among the particles in the vicinity of the boundary region, in particular the symmetry between the first and second closest particles to the boundary. By plotting all the SPH particles near the left boundary, Figure 10.14 clearly indicates the limitation of the treatment formulated by equation (61).

The use of equation (62) provides an effective *ad hoc* modification for the n_d by eliminating the observed heat flux anomaly and it does not influence the overall calculation accuracy.

10.5 Mesoscopic Pore-Scale SPH Model for Fluid Flow in Porous Media

Many industrial applications involve flow in porous media, such as oil exploration, groundwater purification from hazardous wastes, and packed bed chemical reactors. One application area of particular interest is in liquid composite molding

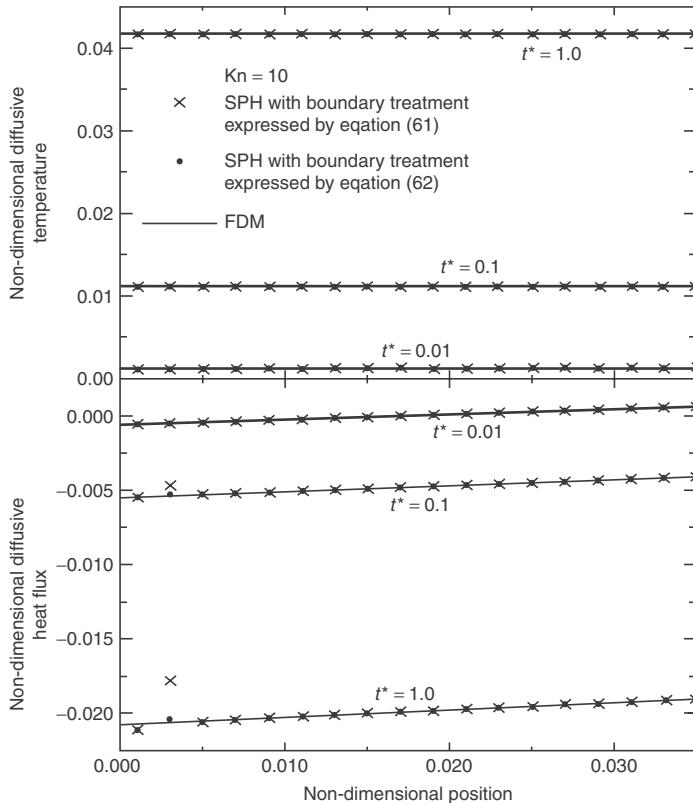


Figure 10.14. Error resulting from the boundary treatment expressed by equation (61) and the *ad hoc* correction effect by using equation (62).

(LCM), a process being widely used to manufacture polymer composite components in the civil, aerospace, automotive and defense industries [40]. Important LCM technologies are the resin transfer molding (RTM) and the structural reaction injection molding (SRIM), and for both cases the understanding of the physics of liquid flow in the porous preform is a requirement for the design of the mold and the choice of the operation procedure, which are critical to the quality of the final product. To this goal CFD numerical modeling can effectively reduce the experimental parametric investigation effort needed for the optimization of the industrial process. Therefore, it can play an important role in practice. CFD modeling of fluid flow in porous media is in general conducted by using Darcy's law [41, 42], which is expressed in the following form:

$$\langle v \rangle = -\frac{\mathbf{K}}{\mu} \nabla P \quad (63)$$

where, $\langle \mathbf{v} \rangle$ and P are the volume averaged (superficial) velocity and the pressure of the fluid (bold text denotes vector or tensor); \mathbf{K} is the permeability tensor of the porous medium; and μ is the dynamic viscosity of the fluid.

Darcy's law is a macroscopic phenomenological model, which is largely based on experimental observations. In this law, the complex interactions between fluid and microscopic porous structures are all lumped in a macroscopic physical quantity – the permeability tensor, \mathbf{K} . As the permeability tensor is of paramount importance to the fluid flow in porous media, numerous research efforts [42–45] were devoted to establish theoretical relations between the permeability and other characteristic material properties with the purpose of avoiding time-consuming physical experiments. These theoretical models are generally based on the analysis of the microscopic porous structure of the medium and the subsequent extraction of the desired macroscopic information. The complicated, diverse pore structures in reality present a major challenge to pure (or semiempirical) theoretical models because they are all constructed on the basis of artificially simple, regular pore arrangements.

Obviously, the modeling of fluid flow in fibrous porous media directly from the mesoscopic pore structure level provides refined fluid flow information and it does not resort to Darcy's law. Moreover, the simulated fluid field is amenable to the determination of \mathbf{K} as well. Over the last few decades, rapid advances in computer capabilities and computational algorithms enabled this kind of modeling work. For example, the lattice-gas-automaton (LGA) or lattice Boltzmann method (LBM) [46], based upon a micro- or mesoscopic model of kinetic formulations or the Boltzmann equation, respectively, can bridge the gap between microscopic structures and macroscopic phenomena. These models have the advantage of allowing parallelism in a straightforward manner for large-scale numerical calculations, and they have the attractive feature of nonslip bounce-back solid boundary treatment for the simulation of fluid flow in porous media. The simulations based on LGA [47, 48] and LBM [49–51] have demonstrated that Darcy's law can be reproduced by these methods. The simulated porous media in the work by Koponen et al. [49] describe three-dimensional (3-D) random fiber webs that closely resemble fibrous sheets such as paper and non-woven fabrics. The computed permeability of these webs presents an exponential dependence on the porosity over a large range of porosity and is in good agreement with experimental data [52, 53]. Spaid and Phelan [51] investigated the resin injection process encountered in RTM. The cell permeability obtained for transverse flow through regularly arranged porous tows of circular or elliptical cross-section agrees well with the semi-analytical solution. In contrast to LBM or LGA, SPH is a meshless particle-based method and offers more freedom in dealing with complicated geometries. Application of SPH to fluid flow in porous media has the potential of providing a mesoscopic/microscopic pore-scale insight into the relevant physics [54–57]. In the present work, SPH is employed to construct a mesoscopic pore-scale model for fluid flow in porous media, in particular, for the transverse fluid flow in randomly aligned fibrous porous media.

10.5.1 Modeling strategy

SPH formulation and methodology

SPH formulation is directly based on the resolution of the macroscopic governing equations of fluid flow. Derivation of the SPH formulation from traditional Eulerian-based equations is routine, as described in Section 2; therefore, in this section is only briefly surveyed.

In SPH, the continuous flow at time t is represented by a collection of N particles located at position $\mathbf{r}_n(t)$ and moving with velocity $\mathbf{v}_n(t)$, $i = 1, 2, \dots, N$. The “smoothed” value of any field quantity $q(\mathbf{r}, t)$ at a space point \mathbf{r} is a weighted sum of all contributions from the neighboring particles

$$\langle q(\mathbf{r}, t) \rangle = \sum_{j=1}^N \frac{m_j}{\rho(\mathbf{r}_j)} q(\mathbf{r}_j, t) w(|\mathbf{r} - \mathbf{r}_j|, h) \quad (64)$$

where, m_j and $\rho(\mathbf{r}_j)$ denote the mass and density of particle j , respectively. $w(|\mathbf{r}|, h)$ is the weight or smoothing function with h being the smoothing length. In the present SPH implementation, the high-order 3-splines quintic kernel [15] is used and the smoothing length h is equal to $\sqrt{2}\Delta S$, with ΔS being the average separation of the SPH particles. In the SPH formulation the gradient of $q(\mathbf{r}, t)$ is determined as:

$$\langle \nabla q(\mathbf{r}, t) \rangle = \sum_{j=1}^N \frac{m_j}{\rho_j} [q(\mathbf{r}_j, t) - q(\mathbf{r}, t)] \nabla w(|\mathbf{r} - \mathbf{r}_j|, h) \quad (65)$$

Equation (65) is already symmetrized [24], and for the second order viscous diffusive term, the SPH theory gives the following formulation [8, 24]:

$$\frac{1}{\rho} \nabla \cdot (\mu \nabla q) = \sum_j \frac{m_j}{\rho_n \rho_j} (\mu_n + \mu_j) q_{nj} \frac{1}{|\mathbf{r}_{nj}|} \frac{\partial w_{nj}}{\partial \mathbf{r}_n} \quad (66)$$

where, $q_{nj} = q_n - q_j$, $\mathbf{r}_{nj} = \mathbf{r}_n - \mathbf{r}_j$ and $w_{nj} = w(|\mathbf{r}_n - \mathbf{r}_j|, h)$.

In terms of equation (65), the “smoothed” version of the continuity equation is:

$$\frac{d\rho_n}{dt} = \sum_j m_j \frac{\mathbf{v}_{nj} \cdot \mathbf{r}_{nj}}{|\mathbf{r}_{nj}|} \frac{\partial w_{nj}}{\partial \mathbf{r}_n} \quad (67)$$

The summation is over all neighboring particles j with exception of particle n itself and $\mathbf{v}_{nj} = \mathbf{v}_n - \mathbf{v}_j$.

The transformation of the N–S equations into their SPH version is relatively involved, because it requires the symmetrization of the pressure terms [24] to satisfy momentum conservation, and the adequate treatment of the viscous terms. In SPH,

the symmetrization version of pressure gradient can be formulated as:

$$\frac{1}{\rho_n} \nabla p_n = \sum_j m_j \left(\frac{p_n}{\rho_n^2} + \frac{p_j}{\rho_j^2} \right) \frac{\mathbf{r}_{nj}}{|\mathbf{r}_{nj}|} \frac{\partial w_{nj}}{\partial \mathbf{r}_n} \quad (68)$$

From equations (66) and (68), the standard SPH form of N–S equation is derived [8]

$$\frac{dv_n}{dt} = - \sum_j m_j \left(\frac{p_n}{\rho_n^2} + \frac{p_j}{\rho_j^2} \right) \frac{\mathbf{r}_{nj}}{|\mathbf{r}_{nj}|} \frac{\partial w_{nj}}{\partial \mathbf{r}_n} + \sum_j \frac{m_j (\mu_n + \mu_j) v_{nj}}{\rho_n \rho_j} \left(\frac{1}{|\mathbf{r}_{nj}|} \frac{\partial w_{nj}}{\partial \mathbf{r}_n} \right) + \mathbf{F}_B \quad (69)$$

where, \mathbf{F}_B denotes an external body force. In this work, the following SPH expression for the momentum equations is used.

$$\begin{aligned} \frac{dv_n}{dt} = & - \sum_j m_j \left(\frac{p_n}{\rho_n^2} + \frac{p_j}{\rho_j^2} + RG_{nj}^\eta \right) \frac{\mathbf{r}_{nj}}{|\mathbf{r}_{nj}|} \frac{\partial w_{nj}}{\partial \mathbf{r}_n} \\ & + \sum_j \frac{m_j (\mu_n + \mu_j) v_{nj}}{\rho_n \rho_j} \left(\frac{1}{|\mathbf{r}_{nj}|} \frac{\partial w_{nj}}{\partial \mathbf{r}_n} \right) + \mathbf{F}_B + \sum f \end{aligned} \quad (70)$$

Equation (70) was proposed by Jiang et al. (2007) and it differs from the standard SPH N–S equation in what concerns two terms on the right-hand side of this equation, namely: (a) RG_{nj}^η , an artificial pressure, which is used to restrain the tensile instability [58]; (b) $\sum f$, an additional force, which is introduced with the purpose of correctly mimicking the no-penetrating restraint prescribed by the solid pore structure. The term RG_{nj}^η is calculated as follows:

$$G_{nj} = \frac{w_{nj}}{w(\Delta S, h)} \quad (71)$$

$$R = \begin{cases} \varphi_1 \frac{|p_n|}{\rho_n^2} & \text{if } p_n < 0 \\ \varphi_1 \frac{|p_j|}{\rho_j^2} & \text{if } p_j < 0 \\ \varphi_2 \left(\frac{|p_n|}{\rho_n^2} + \frac{|p_j|}{\rho_j^2} \right) & \text{if } p_n > 0 \text{ and } p_j > 0 \end{cases} \quad (72)$$

The factors φ_1 and φ_2 are taken to be 0.2 and 0.05, respectively. The exponential η is dependent on the smoothing kernel as $\eta \geq w(0, h)/w(\Delta S, h)$; 3.0 is adopted for η in this work. The repulsive force f (per unit mass) is initiated from a porous material particle and is applied to the neighboring fluid particles within a distance

less than a threshold value r_0 . In these conditions, for most of the fluid particles, $\sum f = 0$, f is assumed to be of the 12-6 Lennard-Jones pair potential form, namely

$$f(r) = \phi F_B \left[\left(\frac{r_0}{r} \right)^{12} - \left(\frac{r_0}{r} \right)^6 \right] \frac{r}{r} \quad (r < r_0) \quad (73)$$

Determination of the constants: ϕ and r_0 were discussed in detail in Jiang et al. [55]; the values $r_0 = 0.6\Delta S$ and $\phi = 15.0$ were found to be suitable for the present SPH implementation.

The quasi-compressible method, presented in Section 2 is used to comply with the incompressibility condition for the fluid. The values of p_0 and c_s are taken in accordance with the following relation

$$c_s^2 = \frac{\gamma p_0}{\rho_0} = \max \left(0.01U^2, 0.01F_B L, \frac{0.01U\mu}{\rho_0 L} \right) \quad (74)$$

where, U and L are the characteristic (or maximum) fluid velocity and the characteristic length of the geometry, respectively.

The positions of SPH particles are determined directly from the flow field using

$$\frac{dr_n}{dt} = v_n \quad (75)$$

A direct numerical integration algorithm is used to perform the integration of equations (67), (70), and (75). The time step Δt is adaptive and determined by the CFL condition [8]

$$\Delta t = \min \left(\frac{0.125h}{\max_n (\sim 100.0U, |v_n|)}, 0.25 \sqrt{\frac{h}{\left| \frac{dv_n}{dt} \right|}}, \frac{0.25h}{c_{s,n}} \right) \quad (76)$$

On the right-hand side of equation (76), the quantity $\sim 100.0U$ is introduced in the denominator of the first constraint to restrict it to a small value so that the calculation advances mostly in a fixed small time step. In this way the accuracy of the integration procedure can be enhanced and the calculation stability is also improved.

Mesoscopic pore structures

The porous matrix is established by randomly packing a certain number of fibers. Only the transverse flow across the fibers is simulated in the present implementation. Typical 2-D porous microstructures are illustrated in Figure 10.15. The cross-sectional shape of the fibers is either circular (Figure 10.15a) or square (Figure 10.15b), and the fibers can be either overlapped with no preconditioning (Figures 10.15a and b) or be isolated from each other (Figure 10.15c). Each porous system is formed by packing fibers of fixed cross-section (shape and area).

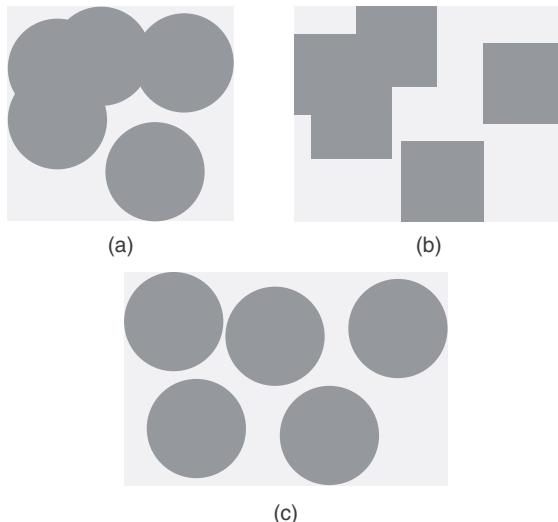


Figure 10.15. Typical 2-D porous microstructures formed by randomly packing a number of fibers: (a) fibers are of circular cross-section with unlimited overlapping between fibers; (b) fibers are of square cross-section with unlimited overlapping between fibers; (c) fibers are of circular cross-section and the fibers are isolated from each other. Void space represents the fluid region.

Different porous systems have fibers of different cross-sectional shapes, different cross-sectional areas or different relative arrangement conditions.

Both the resin fluid and the fibers are represented by SPH particles. The positions of the SPH particles used to represent the solid pore matrix are fixed throughout the simulation. Thus, the porosity ε of a porous system is determined as:

$$\varepsilon = 1 - \frac{\text{number of fixed particles}}{\text{total number of particles}} \quad (77)$$

Fixed porous material particles contribute to the density variations and exert viscous forces upon the nearby fluid particles, while their densities are kept constant. The no-penetrating restraint prescribed by solid pore structure is ensured by activating a repulsive force, as expressed by equation (73), on a fluid particle once it is away from a porous material particle by a distance less than r_0 . With this additional force, the fluid particles are directed to pass by the pore structures in physically acceptable paths [55].

Other relevant conditions

The simulated domain is a 2-D square of dimensions 10 mm \times 10 mm with periodic boundary conditions applied in both the streamwise and transverse flow directions.

Initially, 100×100 SPH particles are distributed uniformly within the simulated domain (i.e. $\Delta x = \Delta y = \Delta S = 0.1$ mm) and are all motionless; the density for all the SPH particles is specified to be ρ_0 .

The fluid flow is driven by a constant body force (F_{Bx}) imposed in the streamwise flow direction. The density, velocity, and position of each fluid particle evolve in time by integrating equations (67), (70), and (75) respectively; while all the porous material particles have a fixed position and constant density (ρ_0). The given physical properties of the resin fluid are $\mu = 0.1$ Pa·s and $\rho_0 = 1,000$ kg/m³. The applied body force per unit of mass F_{Bx} is 10 m/s² for most of the simulations; however, when the porosity is very high, a much smaller body force ($F_{Bx} = 1.0$ or 0.5 m/s²) is imposed to keep the fluid flow in the creeping flow regime.

10.5.2 Results and discussion

Flow visualization

Practical LCM procedures require sufficient wetting of all the fibers by the resin. An identified serious hurdle for LCM process is the so-called preferential flow or edge flow, which is caused by inhomogeneity of the porous preform or unavoidable clearances existing between the preform edges and mold walls. Preferential flow results in the maldistribution of injected resin and makes part(s) of the fibrous preform not to be saturated or fully saturated, i.e., dry spot(s). To exactly predict the preferential flow so as to optimize the resin injection process is crucial to the quality of the final product. Pore-scale SPH model can provide fluid flow information at microscopic/mesoscopic pore level and it has the potential of being very useful in practice. Figure 10.16 presents the visualization of the fluid motion across the aligned fibers due to the applied body force (F_{Bx}). Preferential flow paths are identified by depicting the local velocity vectors. The fluid meanders in the porous microstructures and its path has a tortuosity determined by the location of the embedded fibers. The mesoscopic flows involved in porous systems of different microstructures do not exhibit substantially different patterns and the fluid always flows through the path of least resistance among the porous structures.

Derivation of permeability

The permeability is an important macroscopic property for fluid flow in fibrous porous media. In terms of the simulated steady state flow field, the permeability for the transverse flow through the fibrous porous medium can be calculated as:

$$k = \frac{\mu}{\rho_0 F_{Bx}} \langle u \rangle \quad (78)$$

and this formulation is based on equation (63). To facilitate the comparison between the present results and existing numerical results/experimental data, the permeability k (of a square domain) must be normalized. The normalization parameter is taken to be the square of the characteristic dimension, d^2 , of the embedded fibers; d is defined as the hydrodynamic radius of the embedded fiber's cross-section: for

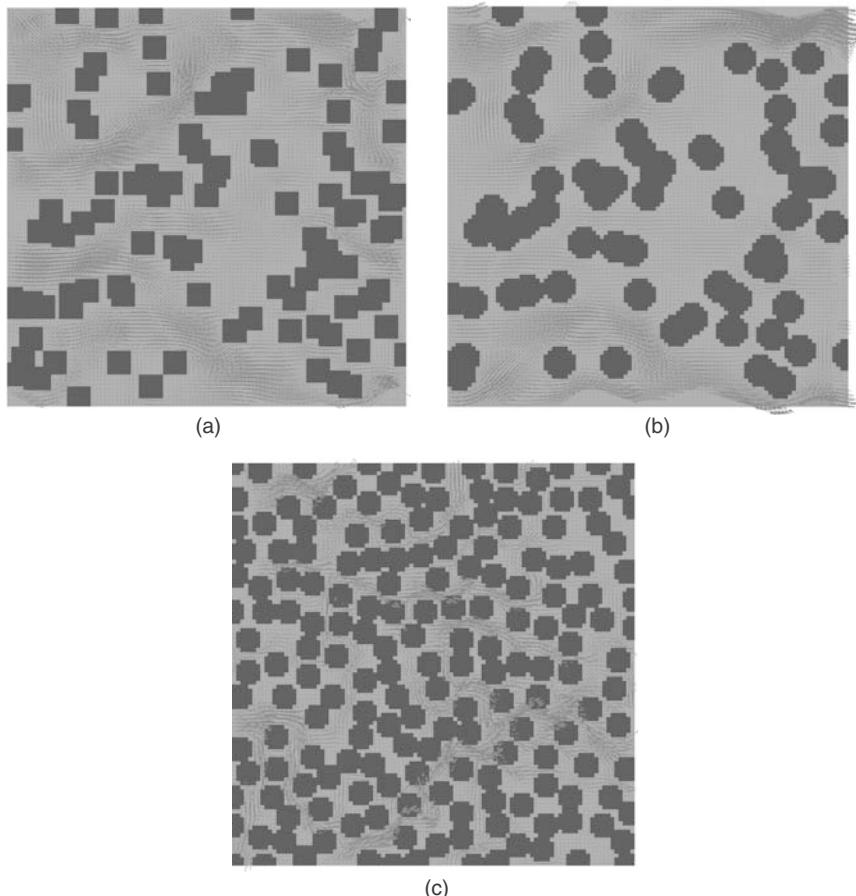


Figure 10.16. Flow visualization in randomly aligned fibrous porous systems. (Note: the length scales for the velocity vectors in the three plots are not the same and resin fluid flows from the left to the right side.) The background of the plot depicts the location of the fibers. (a) Unlimited overlapping between fibers of 0.6×0.6 mm square cross section, $\varepsilon = 0.705$; (b) unlimited overlapping between circular fibers with cross-section of 0.8 mm diameter, $\varepsilon = 0.7061$; (c) isolated circular fibers with cross-section of 0.6 mm diameter, $\varepsilon = 0.4661$.

fibers of circular cross-section, d is the radius; while for fibers of square cross-section, d should be the half side-length of the square. It is worth pointing out that due to the finite numerical discretization, the circular fiber cross-section cannot be approximated accurately and d is calculated exactly in terms of the definition of hydrodynamic radius, instead of simply specifying it equal to the radius of the desired circular cross-section.

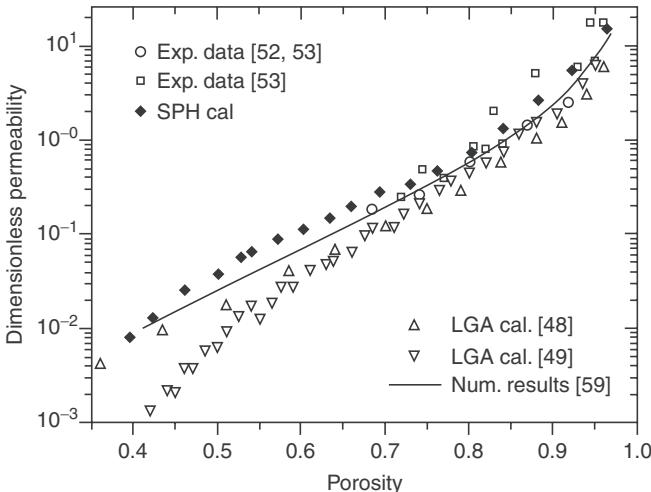


Figure 10.17. A comparison between the calculated permeability for the porous system formed by circular fibers of 0.6 mm diameter cross-section (with unlimited overlapping between fibers) and existing numerical results/experimental data.

The calculated dimensionless permeability for the porous system formed by packing circular fibers of 0.6 mm diameter cross-section with unlimited overlapping between fibers is depicted in Figure 10.17, as a function of porosity.

The comparison between the SPH simulation results and existing experimental data [52, 53] indicates the present SPH calculation technique captures the main features of the fluid flow in fibrous porous media. Computational results derived from other numerical techniques [48, 59] are also consistent, at least in trend, with the present SPH simulation predictions. It should be noted that there are at least two free parameters present in this comparison. First, different porous systems were studied, namely, the LGA results [48] were obtained for 2-D porous systems with rectangular obstacles either randomly distributed (depicted by downward open triangles in Figure 10.17) or placed in a regular square lattice (upward open triangles in Figure 10.17); the physical experiments [52, 53] were performed with respect to 3-D compressed fiber mats and fibrous filters; the (Eulerian-based) numerical results [59] were reported for a 3-D porous system of a face-centered-cubic (fcc) array of fibers. As a second free parameter for comparison, different physical properties of the fluid were assumed. For example, in the present study the viscosity of the fluid is specified as, $\mu = 0.1 \text{ Pa} \cdot \text{s}$, which may be different from that used by other researchers, since in many cases this parameter is not reported. A nearly linear dependence of permeability on viscosity (with fixed porosity) was reported in the work by Koponen et al. [49].

From Figure 10.17, it is clearly seen that when the porosity is smaller than 0.7, the SPH-calculated dimensionless permeability is slightly larger than the FEM result, but is noticeably larger than that obtained by LGA technique. Besides the above-described free parameters between different numerical efforts, there is another reason present. For the porous systems of unlimited overlapping fibers, the embedded fibers may be connected and adjoined to each other, and they form larger obstacles (as shown in Figure 10.15a). This holds true especially when the porosity is low. Therefore, assuming the hydrodynamic radius as the characteristic dimension of the individual fibers, to a great extent, underestimates the actual characteristic dimension of the porous system, which leads to larger dimensionless permeability values. Moreover, for porous systems established by randomly distributed inclusions, the randomness can affect the interior non-homogeneity of the system, which might cause a large variation of the permeability. Koponen et al. [48] stated that the non-homogeneity could increase the permeability as much as 50% as compared to a porous system with homogeneous pore structure. Notably, the SPH-calculated permeability has large departure from the LGA prediction in very low porosity range ($\varepsilon < 0.5$). Non-homogeneity in the structure has significant influence on the effective porosity when the porosity is approaching the percolation threshold. Slight variation of the effective porosity may lead to an order of magnitude change of the permeability. This additional uncertainty makes it extremely difficult to accurately predict the permeability of a porous system with very low porosity by using any numerical technique.

The microstructures inside the porous media may have some effect on the permeability–porosity relation. However, as displayed by Figure 10.18, for the investigated 2-D isotropic porous media, none of the examined participating factors: the inclusion shape (circular or square), the inclusion size and the relative arrangement condition between inclusions (isolated or unlimited overlapping inclusions) brings significant effect on the relation of the dimensionless permeability versus porosity.

There are some distinct features displayed by the permeability – porosity relation shown in Figure 10.18. First, the permeability has a sharp increase, as expected, when the porosity ε approaches one (k is infinite for $\varepsilon = 1$). Second, the permeability seems to follow an exponential dependence on the porosity within the intermediate porosity range, $0.5 \leq \varepsilon \leq 0.8$. A regression fit to the calculated data yields the following exponential relationship:

$$\ln \left(\frac{K}{d^2} \right) = -3.13 + 9.43\varepsilon \quad (79)$$

The correlation coefficient between the simulated points and the exponentially fitted curve is 0.99. Third, the SPH simulations indicate the percolation threshold of the investigated porous media is smaller than 0.4. A value of 0.33 for the percolation threshold was used [48] for a 2-D porous system consisting of randomly distributed rectangular obstacles.

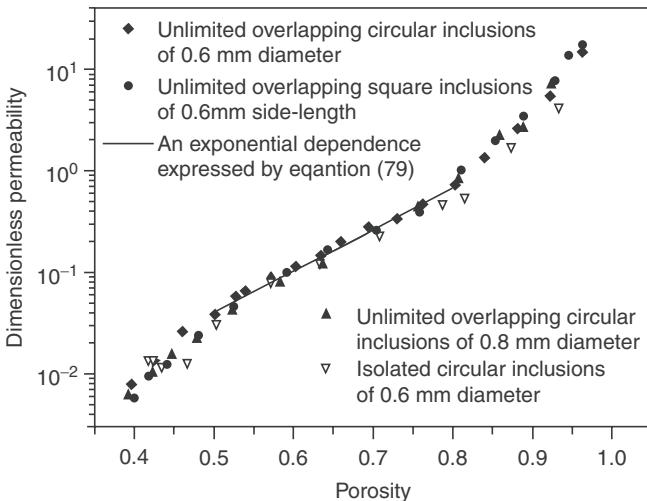


Figure 10.18. Dependence of the dimensionless permeability on pore structure.

10.6 Concluding Remarks

In Section 2 of this chapter, the SPH benchmarking conducted for flow and heat transfer situations with analytical solutions proves the reliability and feasibility of this methodology.

In Section 3, SPH simulations with respect to 2-D channel flow of electrically conducting fluid were used to demonstrate that a MHD methodology to conditioning the transition to turbulence is more effective when the magnetic field is applied at an angle θ , with respect to the main fluid flow direction, within the range of 0° (i.e., streamwise flow direction) to 90° (transverse flow direction). In what concerns the flow at $Re = 10^4$, an applied magnetic field that gives a Stuart number of 40 can effectively stop the onset of turbulence, when it forms an angle θ between 20° and 50° with respect to the main fluid flow direction. For the purpose of obtaining an effective restraining effect a sufficiently strong magnetic field is required, and the applied direction (θ) for a magnetic field of larger Stuart number also gets its effective range enhanced. For a turbulent flow, augmented velocity fluctuations in one direction will transfer turbulent energy to the other two directions with a consequent cascading effect, which may spread over the whole flow region. Therefore, it can be reasoned that an effective MHD turbulence control strategy should consider restraining the velocity fluctuations in all flow directions simultaneously.

In Section 4, the viability of the SPH method to solve the ballistic-diffusive heat conduction equation is demonstrated. It is reported a simple SPH procedure to give accurate temperature and heat flux representations for the ballistic-diffusive heat conduction processes in solid thin films. This approach is primarily based on a particular boundary treatment, which is constructed in terms of the normalization

property of the SPH smoothing kernel. It is shown that this boundary treatment can affect the symmetry of the SPH formulations. The SPH heat flux calculation, based on the first-order derivative of the smoothing kernel, may lead to an erroneous heat flux value at the second particle closest to the boundary. An *ad hoc* modification of the normalization coefficients of the smoothing kernel is capable of giving correct results for all the SPH particles.

Finally, in Section 5, in what concerns the transverse fluid flow across randomly aligned fibrous porous media, a 2-D SPH numerical model is constructed. The developed model fully utilizes the meshless characteristic of SPH methodology and thus offers great flexibility in the construction of complicated pore structures. The proposed modeling approach enables a mesoscopic insight into the fluid flow. Pore-scale preferential flow paths are visualized, the fluid meanders in the porous microstructures and, as expected, always flows along the paths of least resistance. The macroscopic features of the fluid flow are fully captured by this model. For instance, the dimensionless permeability (normalized by the squared characteristic dimension of the fiber cross-section) is found to exhibit an exponential dependence on the porosity in the intermediate porosity range. In addition, it is found that neither the relative arrangement between fibers nor the fiber cross-section (shape or area) has significant effect on this dimensionless permeability–porosity relation.

Acknowledgment

The authors gratefully acknowledge the financial support received by this project from FCT (Foundation for Science and Technology, Portugal) – the research grant POCTI/EME/59728/2004 (ACMS) and the Post-doctoral Fellowship, SFRH/BPD/20273/2004 (FJ), and from NSERC (Natural Sciences and Engineering Research Council of Canada) – Discovery Grant 12875 (ACMS)

References

- [1] Gingold R. A., and Monaghan, J. J. Smoothed particle hydrodynamics: Theory and application to non-spherical stars, *Mon. Not. R. Astron. Soc.*, 181, pp. 375–398, 1977.
- [2] Lucy, L. B. A numerical approach to the testing of the fission hypothesis, *Astron. J.*, 82, pp. 1013–1024, 1977.
- [3] Cleary, P. W., and Monaghan, J. J. Conduction modelling using smoothed particle hydrodynamics, *J. Comput. Phys.*, 148, pp. 227–264, 1999.
- [4] Jeong, J. H., Jhon, M. S., Halow, J. S., and van Osdol, J. Smoothed particle hydrodynamics: Applications to heat conduction, *Comput. Phys. Commun.*, 153, pp. 71–84, 2003.
- [5] Cleary, P. W. Modelling confined multi-material heat and mass flows using SPH, *Appl. Math. Model.*, 22, pp. 981–993, 1998.
- [6] Chaniotis, A. K., Poulikakos, D., and Koumoutsakos, P. Remeshed smoothed particle hydrodynamics for the simulation of viscous and heat conducting flow, *J. Comput. Phys.*, 182, pp. 67–90, 2002.

- [7] Sigalotti, L. D. G., Klapp, J., Sira, E., Melean, Y., and Hasmy, A. SPH simulations of time-dependent Poiseuille flow at low Reynolds numbers, *J. Comput. Phys.*, 191, pp. 622–633, 2003.
- [8] Morris, J. P., Fox, P. J., and Zhu, Y. Modelling low Reynolds number incompressible flows using SPH, *J. Comput. Phys.*, 136, pp. 214–226, 1997.
- [9] Nugent, S., and Posch, H. A. Liquid drops and surface tension with smoothed particle applied mechanics, *Phys. Rev. E*, 62, pp. 4968–4975, 2000.
- [10] Morris, J. P. Simulating surface tension with smoothed particle hydrodynamics, *Int. J. Numer. Meth. Fluids*, 33, pp. 333–353, 2000.
- [11] Colagrossi, A., and Landrini, M. Numerical simulation of interfacial flows by smoothed particle hydrodynamics, *J. Comput. Phys.*, 191, pp. 448–475, 2003.
- [12] Monaghan, J. J. Smoothed particle hydrodynamics, *Annual Rev. Astron. Astrophys.*, 30, pp. 543–574, 1992.
- [13] Randle, R. W., and Libersky, L. D. Smoothed particle hydrodynamics: Some recent improvements and applications, *Appl. Mech. Eng.*, 139, pp. 375–408, 1996.
- [14] Cleary, P. W., and Monaghan, J. J. Boundary interactions and transition to turbulence for standard CFD problems using SPH. *Proc. of the 6th International Computational Techniques and Applications*, Canberra, pp. 157–165, 1993.
- [15] Liu, M. B., Liu, G. R., and Lam, K. Y. Constructing smoothing functions in smoothed particle hydrodynamics with applications, *J. Comput. Appl. Math.*, 155, pp. 263–284, 2003.
- [16] Cleary, P. W., Ha, J., Mooney, J., and Ahuja, V. Effect of heat transfer and solidification on high pressure die casting, *Proc. of the 13rd Australasian Fluid Mechanics Conference*, M. C. Thompson and K. Hourigan, eds., Monash University, pp. 679–682, 1998.
- [17] Monaghan, J. J. Simulating free surface flow with SPH, *J. Comput. Phys.*, 110, pp. 399–406, 1994.
- [18] Allen, M. P., and Tildesley, D. J. Computer simulation of liquids (Chapter 1.4). *Constructing an intermolecular potential*, Oxford: Clarendon, 1989.
- [19] Szekley, J., and Ilegbusi, O. J. *The physical and mathematical modelling of tundish operations*, Springer: Berlin, 1989.
- [20] Hartmann, J., and Lazarus, F. Experimental investigations on the flow of mercury in a homogeneous magnetic field, *K. Dan. Vidensk. Selsk. Mat. Fys. Medd.*, 15(7), pp. 1–45, 1937.
- [21] Lee, D., and Choi, H. Magnetohydrodynamic turbulence flow in a channel at low magnetic Reynolds number, *J. Fluid Mech.*, 439, pp. 367–394, 2001.
- [22] Krasnov, D. S., Zienicke, E., Zikanov, O., Boeck, T., and Thess, A. Numerical study of the instability of the Hartmann layer, *J. Fluid Mech.*, 504, pp. 183–211, 2004.
- [23] Ji, H. C., and Gardner, R. A. Numerical analysis of turbulence pipe flow in a transverse magnetic field, *Int. J. Heat Mass Transfer*, 40, pp. 1839–1851, 1997.
- [24] Jiang, F., Oliveira, M. C. A., and Sousa, A. C. M. SPH simulation of transition to turbulence for planar shear flow subjected to a streamwise magnetic field, *J. Comput. Phys.*, 217, pp. 485–501, 2006.
- [25] Holman, J. P. Heat transfer (7th ed.) (Chapter 12). *Special topics in heat transfer*, McGraw-Hill: New York, 1990.
- [26] Flik, M. I., Choi, B. I., and Goodson, K. E. Heat transfer regimes in microstructures, *ASME J. Heat Transf.*, 114, pp. 666–674, 1992.
- [27] Cattaneo, C. A. Form of heat conduction equation which eliminates the paradox of instantaneous propagation, *Compt. Rend.*, 247, pp. 431–433, 1958.

- [28] Haji-Sheikh, A., Minkowycz, W. J., and Sparrow, E. M. Certain anomalies in the analysis of hyperbolic heat conduction, *ASME J. Heat Trans.*?, 124, pp. 307–319, 2002.
- [29] Jiang, F., and Sousa, A. C. M. Analytical solution to hyperbolic heat propagation in a hollow sphere, *J. Thermophys. Heat Transfer*, 19, pp. 595–598, 2005.
- [30] Chen, J. K., and Beraun, J. E. Numerical study of ultrashort laser pulse interactions with metal films, *Numer. Heat Transfer A*, 40, pp. 1–20, 2001.
- [31] Chen, G. Ballistic-diffusive heat conduction equations, *Phys. Rev. Lett.*, 86, pp. 2297–2300, 2001.
- [32] Olfe, D. B. A modification of the differential approximation for radiative transfer, *AIAA J.*, 5, pp. 638–643, 1967.
- [33] Modest, M. F. The modified differential approximation for radiative transfer in general three-dimensional media, *J. Thermophys. Heat Transfer*, 3, pp. 283–288, 1989.
- [34] Yang, R., Chen, G., Laroche, M., and Taur, Y. Simulation of nanoscale multidimensional transient heat conduction problems using ballistic-diffusive equations and phonon boltzmann equation, *ASME J. Heat Trans.*, 127, pp. 298–306, 2005.
- [35] Chen, J. K., Beraun, J. E., and Camey, T. C. A corrective smoothed particle method for boundary value problems in heat conduction, *Int. J. Numer. Meth. Eng.*, 46, pp. 231–252, 1999.
- [36] Chen, J. K., and Beraun, J. E. Numerical study of ultrashort laser pulse interactions with metal films, *Numer. Heat Transfer A*, 40, pp. 1–20, 2001.
- [37] Qiu, T. Q., and Tien, C. L. Heat transfer mechanisms during short-pulse laser heating of metals, *ASME J. Heat Trans.*, 115, pp. 835–841, 1993.
- [38] Brookshaw, L. A method of calculating radiative heat diffusion in particle simulations, *Publ. Astron. Soc. Aust.*, 6, pp. 207–210, 1985.
- [39] Jiang, F., and Sousa, A. C. M. SPH numerical modelling for ballistic-diffusive heat conduction, *Numer. Heat Transfer B*, 50, pp. 499–515, 2006.
- [40] Potter, K. *Resin transfer moulding*, Springer: New York, 1997.
- [41] Bickerton, S., and Advani, S. G. Characterization and modelling of race-tracking in liquid composite molding processes, *Compos. Sci. Technol.*, 59, pp. 2215–2229, 1999.
- [42] Phelan Jr., F. R., and Wise, G. Analysis of transverse flow in aligned fibrous porous media, *Composites A*, 27, pp. 25–34, 1996.
- [43] Sangani, A. S., and Acrivos, A. Slow flow past periodic arrays of cylinders with application to heat transfer, *Int. J. Multiphase Flow*, 8, pp. 193–206, 1982.
- [44] Katz, A. J., and Thompson, A. H. Quantitative prediction of permeability in porous rock, *Phys. Rev. B*, 34, pp. 8179–8181, 1986.
- [45] Bear, J. *Dynamics of fluids in porous media*, Dover: New York, 1972.
- [46] Chen, S., and Doolen, G. D. Lattice Boltzmann method for fluid flows, *Annu. Rev. Fluid Mech.*, 30, pp. 329–364, 1998.
- [47] Chen, S., Diemer, K., Doolen, G. D., Eggert, K., Fu, C., Gutman, S., and Travis, B. J. Lattice gas automata for flow through porous media, *Physica D*, 47, pp. 72–84, 1991.
- [48] Koponen, A., Kataja, M., and Timonen, J. Permeability and effective porosity of porous media, *Phys. Rev. E*, 56, pp. 3319–3325, 1997.
- [49] Koponen, A., Kandhai, D., Hellen, E., Alava, M., Hoekstra, A., Kataja, M., Niskanen, K., Sloot, P., and Timonen, J. Permeability of three-dimensional fiber webs, *Phys. Rev. Lett.*, 80, pp. 716–719, 1998.
- [50] Cancelliere, A., Chang, C., Foti, E., Rothman D. H., and Succi, S. The permeability of a random medium: Comparison of simulation with theory, *Phys. Fluids A*, 2, pp. 2085–2088, 1990.

- [51] Spaid, M. A. A., and Phelan, F. R. Lattice Boltzmann methods for modelling microscale flow in fibrous porous media, *Phys. Fluids*, 9, pp. 2468–2474, 1997.
- [52] Ingmanson, W. L., Andrews, B. D., and Johnson, R. C. Internal pressure distributions in compressible mats under fluid stress, *Tappi J.*, 42, pp. 840–849, 1959.
- [53] Jackson, G. W., and James, D. F. The permeability of fibrous porous media. *Can. J. Chem. Eng.*, 64, pp. 364–374, 1986.
- [54] Zhu, Y., Fox, P. J., and Morris, J. P. A pore scale numerical model for flow through porous media, *Int. J. Numer. Anal. Meth. Geomech.*, 23, pp. 881–904, 1999.
- [55] Jiang, F., Oliveira, M. C. A., and Sousa, A. C. M. Mesoscale SPH modelling of fluid flow in isotropic porous media, *Comput. Phys. Commun.*, 176, pp. 471–480, 2007.
- [56] Jiang, F., and Sousa, A. C. M. Effective thermal conductivity of heterogeneous multi-component materials: An SPH implementation, *Heat Mass Transfer* 43, pp. 479–491, 2007.
- [57] Jiang, F., and Sousa, A. C. M. Smoothed particle hydrodynamics modelling of transverse flow in randomly aligned fibrous porous media, *Transport Porous Media*, 75, pp. 17–33, 2008.
- [58] Monaghan, J. J. SPH without a tensile instability, *J. Comput. Phys.*, 159, pp. 290–311, 2000.
- [59] Higdon, J. J. L., and Ford, G. D. Permeability of three-dimensional models of fibrous porous media, *J. Fluid Mech.*, 308, pp. 341–361, 1996.

11 Evaluation of continuous and discrete phase models for simulating submicrometer aerosol transport and deposition

Philip Worth Longest¹ and Jinxiang Xi²

¹*Virginia Commonwealth University, Richmond, VA, USA*

²*University of Arkansas, Little Rock, AR, USA*

Abstract

Submicrometer aerosols are widely prevalent throughout the environment. Examples of these aerosols include combustion byproducts, ions from radioactive decay, and respiratory-specific viruses. Predicting the transport and fate of these aerosols is important in a number of applications including large-scale environmental assessments, determining pollutant exposure levels in microenvironments, and evaluating the lung deposition and potential health effects of inhaled particles. Numerical simulations of submicrometer aerosols are challenging due to low deposition efficiencies, the action of concurrent inertial and diffusive transport mechanisms, and stiff equation sets. As with other two-phase flow systems, both continuous and discrete phase models have been used to simulate the transport and deposition of submicrometer aerosols. However, each of these approaches has inherent strengths and weaknesses. In this review, the performance of continuous and discrete phase models is assessed in a series of case studies that focus on submicrometer aerosol transport and deposition in the respiratory tract. The numerical methods considered include a chemical species approach that neglects particle inertia, Lagrangian particle tracking, and a recently proposed drift flux model that utilizes a near-wall analytic solution. The performance of these models is evaluated through comparisons with experimental data sets of deposition in a representative double bifurcation geometry, a realistic model of the upper tracheobronchial region, and a realistic model of the nasal cavity. Comparisons of the mass transport methods considered with experimental data highlight advantages and disadvantages of each approach. These results are intended to provide guidance in selecting appropriate two-phase models for simulating submicrometer aerosol transport and deposition in a variety of applications including exposure analysis, respiratory dosimetry, and respiratory drug delivery.

Keywords: Modeling nanoaerosols, Submicrometer aerosol dynamics, Drift flux model, Lagrangian particle tracking

11.1 Introduction

Fine respiratory aerosols can be characterized as having a particle diameter in the range of approximately 100 nm to 1 μm . Inhalable fine aerosols include diesel exhaust products, environmental and mainstream tobacco smoke, and airborne bioaerosols such as viruses and bacteria. Morawska et al. [1] reported average count median diameters of sample diesel exhaust and environmental tobacco smoke to be 125 and 183 nm, respectively. The particle fraction of diesel exhaust can range from 5 to 500 nm [2]. Bernstein [3] summarized a number of studies and reported a consistent value of cigarette smoke count median diameter of approximately 260 nm surrounded by a wide distribution of values ranging from 18 nm to 1.6 μm . Respiratory-specific viruses such as Avian flu and SARS typically range from 20 to 200 nm [4].

Fine aerosols deposit in the respiratory tract at a reduced rate in comparison with smaller ultrafine (<100 nm) and larger coarse ($>1 \mu\text{m}$) aerosols. This reduction in deposition for fine aerosols is due to a minimum in the net sum of diffusion, sedimentation, and impaction effects [5]. In vivo experiments of whole-lung aerosol deposition in humans indicate a minimum in retention for fine aerosols with a diameter of approximately 400 nm [6–8]. For environmental tobacco smoke and diesel exhaust, the in vivo deposition study of Morawska et al. [1] reported deposition fractions of 36 and 30%, respectively. While deposition rates are typically low for fine respiratory aerosols, environmental pollutants within this size range are often carcinogenic [9] and inflammatory [10]. As a result, understanding the transport and deposition characteristics of fine respiratory aerosols is critical in order to make accurate toxicology risk assessments, to establish appropriate environmental standards, and to determine safe exposure limits.

Localized accumulations of particles have been widely observed at branching sites in bifurcating respiratory geometries. These focal depositions, or hot spots, are often most pronounced for coarse aerosols due to inertial driven impaction [11–17]. However, significant increases in local deposition have also been observed for dilute submicrometer respiratory aerosols and vapors [17,18–21]. The quantity of particles depositing within a specific localized area will affect the transport of particles and chemical species (CS) across the mucus barrier and ultimately the cellular-level dose received [18]. As a result, evaluations of localized deposition values are needed to better predict cellular-level dose and response [22].

One advantage of computational fluid dynamics (CFD) predictions compared with in vivo and in vitro studies is that modeling results can readily determine localized deposition characteristics in three dimensions. However, due to numerical challenges only few computational studies have quantified the local deposition of fine respiratory aerosols. Zhang et al. [17] compared regional and local depositions of coarse and ultrafine particles, but fine aerosols were not considered. Longest and Oldham [23] modeled the effect of an upstream aerosol delivery system on the deposition of 1 μm aerosols in a double bifurcation geometry. Robinson et al. [24] considered the total deposition of fine respiratory aerosols in a model of

respiratory generations G3–G8. Longest and Oldham [25] were the first to compare CFD-predicted local depositions of fine aerosols with in vitro data.

Numerical models for simulating particle transport and deposition typically employ either a Eulerian–Lagrangian [18,24,26] or a Eulerian–Eulerian [17,21,27,28] approach for submicrometer aerosol transport and deposition. The Eulerian model that is typically used for respiratory aerosols treats the particle phase as a dilute CS in a multicomponent mixture. This approach neglects particle inertia and only considers deposition due to diffusional effects. However, Longest and Xi [20] illustrated a significant influence of inertia on the regional and local deposition of fine and ultrafine aerosols. In contrast to the CS Eulerian model, the Lagrangian approach tracks individual particles within the flow field and can account for a variety of forces on the particle including inertia, diffusion, gravity effects, and near-wall interactions [29]. Disadvantages of the Lagrangian model include an excessively large number of particles required to establish convergent local deposition profiles and a stiff equation set for fine and ultrafine particles [30]. In the study of Longest and Oldham [23], a Lagrangian model was used to match the overall in vitro deposition rate of $1 \mu\text{m}$ particles in a double bifurcation geometry; however, the model was not able to match the local deposition characteristics.

In contrast to the CS Eulerian model that is typically applied to ultrafine and fine respiratory aerosols, other continuous field two-phase flow methods are available [31,32]. Respiratory aerosols are typically dilute, such that constant flow field properties can be assumed. However, finite particle inertia has been shown to strongly affect the deposition of fine and ultrafine particles [20]. Of the available two-phase methods, the drift flux (DF) model is an effective approach that can be applied to dilute systems with low Stokes numbers St_k and can account for finite particle forces [33]. Wang and Lai [34] applied a DF model to the deposition of ultrafine through coarse respiratory aerosols to account for gravity and electrostatic effects. However, no previous studies have considered a DF model to approximate the inertia of respiratory aerosols. A potential problem with the DF model applied to fine aerosols is the estimation of particle inertia in the near-wall region.

This chapter will discuss (i) the effect of inertia on the deposition of submicrometer particles and the appropriate bounds for the application of a CS Eulerian model; (ii) development of a new drift flux model that effectively takes into account finite particle inertia; and (iii) evaluation of this new drift flux velocity correction (DF-VC) model by comparisons with experimental data and other model predictions in human respiratory airways. Direct comparisons between numerical and experimental deposition results are made on a regional and highly localized basis. Computational models evaluated include a standard CS mass fraction approximation, Lagrangian particle tracking, and the DF approach to account for finite particle inertia. The airway geometries considered in this study include a idealized bifurcation model, a cast-based tracheobronchial (TB) geometry, and an in vivo scan-based nasal cavity model. Agreement between computational and experimental results will help to establish an effective approach for simulating fine respiratory aerosol deposition where both inertial and diffusional effects may be significant transport mechanisms.

11.2 Models of Airflow and Submicrometer Particle Transport

Flow fields in the respiratory tract are typically assumed to be isothermal and incompressible. Depending on activity conditions, the flow regime under normal breathing can be laminar, transitional, or fully turbulent. Turbulence models have different levels of complexity and include direct numerical simulation (DNS), large eddy simulation (LES), the Reynolds stress model (RSM), and the Reynolds-averaged Navier–Stokes (RANS) two-equation approach. Selection of an appropriate model for the flow of interest mainly depends on the desired accuracy and the available computational resources. DNS is the most accurate approach and resolves turbulent eddies at all scales, but it also requires the most computational resources. Large eddy simulations resolve large-scale energy-containing eddies, and the RSM approach captures the anisotropic turbulence, which is significant near the wall. Even though the lower-order RANS two-equation models cannot account for turbulence anisotropy, these models are still often shown to adequately capture the main features of the flow, provided a very fine mesh is employed in the near-wall boundary region [35]. In this study, the low Reynolds number (LRN) $k - \omega$ two-equation model was selected based on its ability to accurately predict pressure drop, velocity profiles, and shear stress for transitional and turbulent flows [36,37]. This model was demonstrated to accurately predict particle deposition profiles for transitional and turbulent flows in models of the oral airway [38,39], nasal passages [40], and multiple bifurcations [41,42]. Moreover, the LRN $k - \omega$ model was shown to provide an accurate solution for laminar flow as the turbulent viscosity approaches zero [37]. Transport equations governing the turbulent kinetic energy (k) and the specific dissipation rate (ω) are provided by Wilcox [37] and were previously reported in Longest and Xi [30].

11.2.1 Chemical species model for particle transport

There is an ever-increasing literature base on particle dynamics in turbulent flows, as reviewed by Crowe et al. [43]. Depending on the need to simulate particle inertia and coupling with the continuous phase, Eulerian-based particle transport models can be categorized as CS, DF, or mixture models. In the respiratory system, dilute suspensions of nanoparticles are often treated as a CS with a Eulerian mass transport model. This model often neglects particle inertia and the effects of the particle phase on the flow field, i.e., one-way coupled particle transport. The transport relation governing the convective–diffusive motion of ultrafine aerosols in the absence of particle inertial effects can be written on a mass fraction basis as:

$$\frac{\partial c}{\partial t} + \frac{\partial(u_j c)}{\partial x_j} = \frac{\partial}{\partial x_j} \left[\left(\tilde{D} + \frac{v_T}{Sc_T} \right) \frac{\partial c}{\partial x_j} \right] \quad (1)$$

In the above equation, c represents the mass fraction of nanoparticles, \tilde{D} is the molecular or Brownian diffusion coefficient, and Sc_T the turbulent Schmidt number

taken to be 0.9. Assuming dilute concentrations of spherical particles, the Stokes–Einstein equation was used to determine the diffusion coefficients for various sized particles and can be expressed as:

$$\tilde{D} = \frac{k_B T C_c}{3\pi\mu d_p} \quad (2)$$

where $k_B = 1.38 \times 10^{-16}$ cm²/g/s is the Boltzmann constant expressed in cgs units. The Cunningham correction factor has been computed using the expression of Allen and Raabe [44]

$$C_c = 1 + \frac{\lambda}{d_p} \left(2.34 + 1.05 \exp \left(-0.39 \frac{d_p}{\lambda} \right) \right) \quad (3)$$

where λ is the mean free path of air, assumed to be 65 nm. The above expression is reported to be valid for all particle sizes [45]. To approximate particle deposition on the wall, the boundary condition for the Eulerian transport model is assumed to be $c_{\text{wall}} = 0$.

11.2.2 Discrete phase model

One-way coupled trajectories of monodisperse ultrafine particles ranging in diameter (d_p) from 1 to 1,000 nm are often calculated on a Lagrangian basis by integrating an appropriate form of the particle trajectory equation. Aerosols in this size range have very low Stokes numbers ($St_k = \rho_p d_p^2 C_c U / 18\mu D \ll 1$), where U is the mean fluid velocity and D a characteristic diameter of the system. Other characteristics of the aerosols considered include a particle density $\rho_p = 1.00$ g/cm³, a density ratio $\alpha = \rho/\rho_p \approx 10^{-3}$, and a particle Reynolds number $Re_p = \rho|u - v|d_p/\mu \ll 1$. The appropriate equations for spherical particle motion under these conditions can be expressed as:

$$\frac{dv_i}{dt} = \alpha \frac{Du_i}{Dt} + \frac{f}{\tau_p C_c} (u_i - v_i) + f_{i, \text{Brownian}} \quad \text{and} \quad \frac{dx_i}{dt} = v_i(t) \quad (4a \text{ and } 4b)$$

In the above equations, v_i and u_i are the components of the particle and local fluid velocity, respectively. The characteristic time required for particles to respond to changes in the flow field, or the momentum response time, is $\tau_p = C_c \rho_p d_p^2 / 18\mu$. The pressure gradient or acceleration term is often neglected for aerosols due to small values of the density ratio. However, it has been retrained here to emphasize the significance of fluid element acceleration in biofluid flows [29]. The drag factor f , which represents the ratio of the drag coefficient C_D to Stokes drag, is assumed to be one for submicrometer aerosols. The effect of Brownian motion on the particle trajectories is included as a separate force per unit mass term at each time-step. The amplitude of the Brownian force is of the form [46]

$$f_{i, \text{Brownian}} = \varsigma_i \sqrt{\frac{\pi S_o}{\Delta t}} \quad (5)$$

where ξ_i is a zero mean variant from a Gaussian probability density function, S_o a spectral intensity function directly related to the diffusion coefficient, and Δt the time-step for particle integration. The influence of anisotropic fluctuations in the near-wall region is considered by implementing an anisotropic turbulence model proposed by Matida et al. [47], which is described as:

$$u'_n = f_v \xi \sqrt{2k/3} \quad \text{and} \quad f_v = 1 - \exp(-0.002y^+) \quad (6)$$

In this equation, ξ is a random number generated from a Gaussian probability density function and f_v a damping function component normal to the wall for values of y^+ less than approximately 40.

11.2.3 Deposition factors

For Lagrangian tracking, the deposition efficiency (DE) is defined as the ratio of particles depositing within a region to the particles entering that region. For the Eulerian model, the mass deposition rate on a wall can be expressed as

$$\dot{m}_{w,i} = \sum_{j=1}^N -\rho_m A_j \left(\tilde{D} + \frac{\nu_T}{Sc_T} \right) \frac{\partial c}{\partial n} \Big|_{w,j} \quad (7)$$

where the summation is performed over region of interest i , and n is the wall-normal coordinate pointing out of the geometry. Localized deposition can be presented in terms of a deposition enhancement factor (DEF), which quantifies local deposition as a fraction of the total or regional DE. A DEF, similar to the enhancement factor suggested by Balashazy et al. [11], for local region j can be defined as

$$DEF_j = \frac{DE_j/A_j}{\sum_{j=1}^N DE_j / \sum_{j=1}^N A_j} \quad (8)$$

where the summation is performed over the region of interest, i.e., the upper airway geometries. For the Lagrangian model, the local area A_j is assumed to be a region with a diameter of $500 \mu\text{m}$, or approximately 50 lung epithelial cells in length [48]. The definition of the DEF is for a pre-specified constant area at each sampling location. Sampling locations are taken to be nodal points. Constant areas are then assessed around each nodal point and allowed to overlap if necessary.

11.2.4 Numerical methods

To solve the governing mass and momentum conservation equations in each of the cases considered, the CFD package Fluent 6 was employed. User-supplied Fortran and C programs were implemented for the calculation of initial particle profiles, wall mass flow rates, Brownian force [30], anisotropic turbulence effect [47,49],

and near-wall velocity interpolation [30]. All transport equations were discretized to be at least second order accurate in space. A segregated implicit solver was employed to evaluate the resulting linear system of equations. This solver uses the Gauss–Seidel method in conjunction with an algebraic multigrid approach to solve the linearized equations. The SIMPLEC algorithm was employed to evaluate pressure–velocity coupling. Convergence of the flow field solution was assumed when the normalized global mass residuals fell below 10^{-5} and the residual–iteration curves for all flow parameters become asymptotic.

11.3 Evaluation of Inertial Effects on Submicrometer Aerosols

A direct comparison between the Eulerian and Lagrangian transport model results can be visualized in a branching respiratory geometry by snapshots of particle locations at two selected times during a washout phase with a tracheal flow rate of 30 L/min (Figure 11.1). The initial velocity profile was parabolic resulting in a near-wall region of low flow beginning at the inlet. For each model, 5 nm particles were initially released from an inlet plane for 0.003 seconds. The simulation was then continued over time to illustrate the washout of the remaining particle fraction. For the Eulerian simulation at $t = 0.005$ seconds, a high concentration region is observed to interact with the first carina and inner walls of the daughter branches (Figure 11.1a). However, these elevated concentrations have been eliminated from the first carina by time $t = 0.015$ seconds due to the high velocities in this region (Figure 11.1b). Moderately elevated concentrations of mass fraction are observed near the second carinas at $t = 0.015$ seconds (Figure 11.1b). As the simulation continues from $t = 0.005$ to 0.015 seconds, washout of the mass fraction is observed in regions of high flow, whereas regions of low flow are dominated by mixing.

Snapshots of Lagrangian particles appear similar to the Eulerian mass transport model over time (Figure 11.1c and d). For time 0.005 seconds, a region of high concentration is again observed to interact with the first carinal ridge. However, the Lagrangian particles simulated have not progressed as far as the minimum contour level considered for the Eulerian flow (Figure 11.1a vs. c). This may be due to the discrete nature of the Lagrangian particles and the relatively small number of particles considered for this illustration. Approximately 2,000 particles were tracked for visualization. However, the Lagrangian model also directly accounts for the molecular slip correction and Taylor diffusion. At time 0.015 seconds, the Eulerian and Lagrangian models again appear similar (Figure 11.1b vs. d). It is noted that the Eulerian model is presented for a mid-plane slice, whereas the Lagrangian model must be presented in terms of particles in the 3-D field. Therefore, the mid-plane view of the Eulerian model reveals only a thin near-wall region of elevated concentration in G3 at time 0.015 seconds (Figure 11.1b). In contrast, this thin near-wall concentration layer of slow-moving particles is observed to occupy a majority of the third generation based on the 3-D view through the geometry for the Lagrangian result (Figure 11.1d).

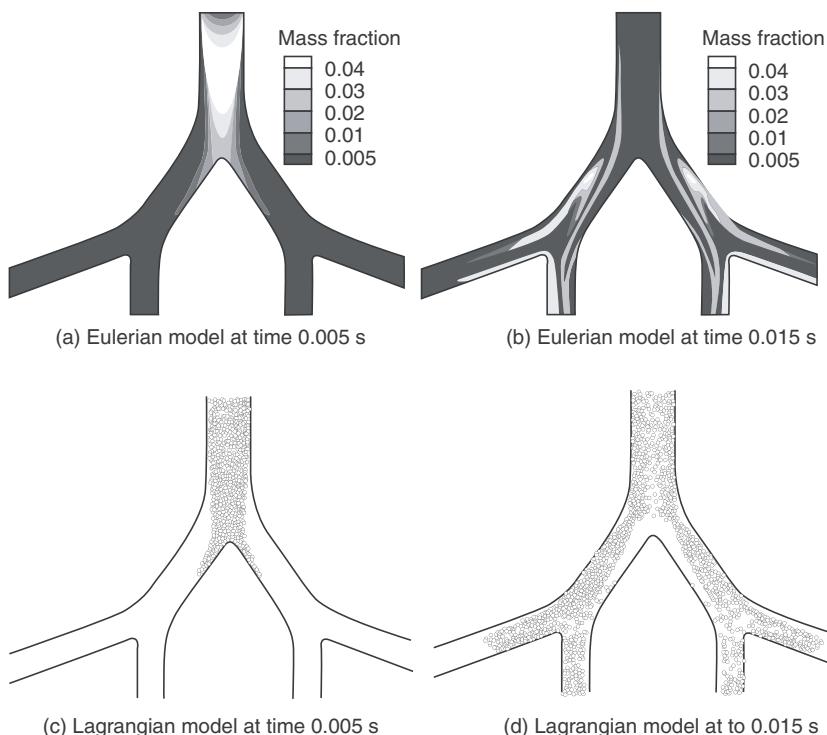


Figure 11.1. Comparison of Eulerian species mass fraction (a and b) and Lagrangian particle distributions (c and d) at various times for an inhalation flow rate of 30 L/min in the upper airway model. A constant concentration of 5 nm particles was released at the inlet plane for the first 0.003 seconds of the simulation.

Figure 11.2 shows the regional deposition fractions based on both Eulerian and Lagrangian particle tracking models in comparison to experimental and analytical results within an upper airway double bifurcation geometry (generation G3–G5) for a tracheal flow rate of 30 L/min. Predictions of the Eulerian transport model are consistent with the correlation of Cohen and Asgharian [50] as well as the analytic solution of Ingham [27] (Figure 11.2). Based on available empirical data, the correlation of Cohen and Asgharian [50] is valid for particles from 40 to 200 nm and includes a high degree of variability. Over this range, good agreement is observed between the Eulerian and empirical models (Figure 11.2). The particle size and Stokes number for which the Lagrangian particle deposition fraction exceeds Eulerian particle deposition by a difference of 20% has been marked (Figure 11.2). This particle size is considered to be the threshold where impaction plays a significant role in total deposition fraction and will be referred to as the inertial particle diameter. At 30 L/min, the inertial particle diameter is 120 nm and

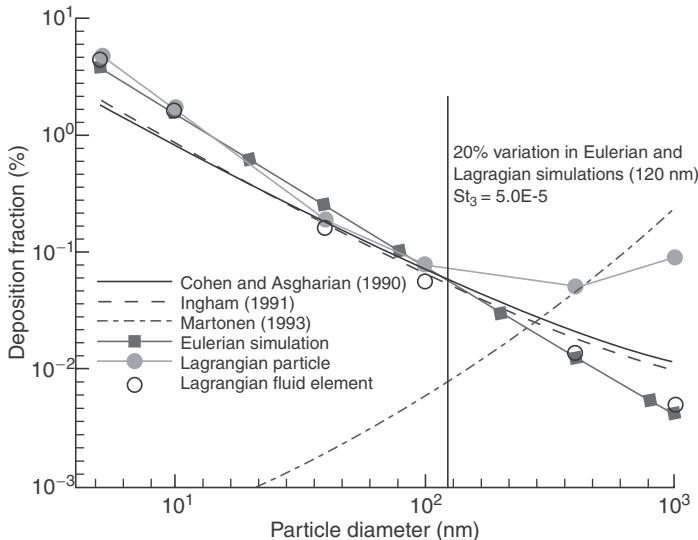


Figure 11.2. Branch-averaged deposition fraction in respiratory generations G3–G5 of the upper airway model for an inhalation flow rate of 30 L/min. Good agreement is observed between the simulated results and the empirical correlation of Cohen and Asgharian (1990) from 40 through 200 nm. Separation of Eulerian and Lagrangian simulation results indicates the onset of particle impaction. Particle impaction is observed to become significant for 120 nm particles.

the associated Stokes number for generation G3 is 5.0×10^{-5} . As the flow rate is increased, the minimum particle size where impaction becomes important grows smaller due to increasing flow inertia. The smallest observed inertial particle diameter was 70 nm ($St = 5.1 \times 10^{-5}$), which occurred for the highest tracheal flow rate considered, which was 60 L/min.

To ensure accuracy in the particle tracking routine, the deposition fraction of massless Lagrangian fluid elements has also been assessed (Figure 11.2). Fluid elements are assumed to have the density of the continuous fluid, or air, resulting in negligible inertia effects. The diffusion of fluid elements is based on the geometric element diameter and is not influenced by the density. The resulting deposition fraction of Lagrangian fluid elements is observed to be in close agreement with the Eulerian model results for all particle sizes considered (Figure 11.2). Therefore, differences in the Lagrangian and Eulerian solutions may be assumed to represent the effects of finite particle inertia and not numerical errors in the particle tracking routine.

In comparison to regional-averaged values, the effects of particle inertia on localized deposition characteristics were found to be much more dramatic [20,40,42]. As shown by Longest and Xi [20], for the upper airway bifurcation

model, inclusion of particle inertia was found to increase the maximum local microdosimetry factor by one order of magnitude for 40 nm particles at an inhalation flow rate of 30 L/min. The maximum DEF values predicted by the Lagrangian model was 103.9 in contrast to a DEF_{\max} of 5.7 predicted by the Eulerian CS model [20]. It was also observed that the Lagrangian model did not appear to resolve continuous contours of DEF below values of approximately 5.

From the above observations, it is indicated that particle inertia may be more significant in the regional and local deposition of fine and ultrafine aerosols than previously considered. Therefore, CS Eulerian models of particle transport may significantly underestimate branch-averaged and local deposition values of fine and ultrafine particle deposition. This underestimation was found to be especially significant for localized deposition patterns and microdosimetry estimates. For 40 nm particles, a one order of magnitude increase of the DEF values was observed in Lagrangian estimates in comparison to Eulerian predictions for the upper airway model at an inhalation flow rate of 30 L/min. Furthermore, the particle inertia cannot be ignored for regional deposition calculations with Stokes numbers larger than 5.0×10^{-5} in bifurcating airways. As a result, it is necessary to develop an appropriate particle transport model that can effectively resolve continuous contours of deposition and capture the influence of finite particle inertia for submicrometer aerosols in the respiratory tract.

11.4 An Effective Eulerian-Based Model for Simulating Submicrometer Aerosols

The basis of this Eulerian approach is the DF model with a novel extension to better compute wall deposition on a continuous basis. In contrast with the CS model, the DF approach includes particle inertial effects, which are taken into consideration through the convection term [25,33]

$$\frac{\partial c}{\partial t} + \frac{\partial(v_j c)}{\partial x_j} = \frac{\partial}{\partial x_j} \left(\left(\tilde{D} + \frac{v_T}{\text{Sc}_T} \right) \frac{\partial c}{\partial x_j} \right) \quad (9)$$

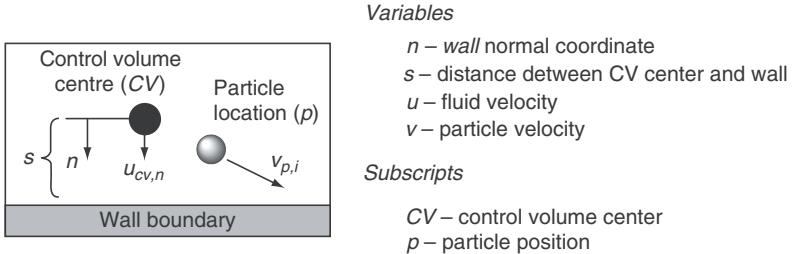
In this equation, v_j is the particle velocity, which is evaluated from the particle slip velocity (vs_j) as:

$$v_j = vs_j + u_j \quad (10)$$

For a continuous field solution, the particle slip velocity can be determined as a function of inertial and gravity forces as [33]

$$vs_j = \frac{C_c d_p^2}{18\mu_c} (\rho_p - \rho_m) \left[g_j - \frac{\partial u_j}{\partial t} - u_i \frac{\partial u_j}{\partial x_i} \right] \quad (11a)$$

where C_c is the Cunningham correction factor, ρ_p and ρ_m are the particle and mixture densities, d_p the particle diameter, and μ_c the continuous field viscosity.

*Variables*

n – wall normal coordinate
 s – distance between CV center and wall
 u – fluid velocity
 v – particle velocity

Subscripts

CV – control volume center
 p – particle position

Figure 11.3. Velocity components and notation within a near-wall control volume.

In equation (11a), the first term in brackets is the gravity vector and the next two terms represent the material derivative, which accounts for fluid element acceleration. For the small particles considered in this study ($\leq 1 \mu\text{m}$), gravity effects are neglected and Stokes flow conditions can be assumed. As a result, equation (11a) can be written in terms of the fluid pressure gradient as [33]:

$$vs_j = \frac{C_c d_p^2}{18\mu_c} \frac{(\rho_m - \rho_p)}{\rho_m} \frac{\partial p}{\partial x_j} \quad (11b)$$

As with the CS model, the DF approach approximates perfect absorption at the wall, i.e., $c_{\text{wall}} = 0$. The associated local mass deposition as a result of both diffusional and inertial effects is expressed as:

$$\dot{m}_{w,l} = -\rho_m A_l \tilde{D} \left. \frac{\partial c}{\partial n} \right|_{\text{wall}} + \rho_m A_l c v_n|_{\text{wall}} \quad (12a)$$

In the above expression, v_n represents the wall-normal particle velocity scalar

$$v_n = v_i \hat{n}_i \quad (12b)$$

where \hat{n}_i is the local wall-normal unit vector pointing out the geometry.

Differences among the two DF approaches considered in this study are based on how the particle velocity at the wall is computed and implemented in the particle deposition expression, equation (12a). Velocity values near the wall on a control volume grid and related nomenclature are indicated in Figure 11.3. For the standard DF model, slip velocities are only available at control volume center locations. As a result, the particle velocity at the wall is approximated as the value at the center of the nearest control volume, i.e.,

$$v_n|_{\text{wall}} = v_{cv,i} \hat{n}_i = v_{cv,n} \quad (13)$$

As indicated in Figure 11.3, the subscript CV indicates the value at the control volume center and the subscript n indicates the wall-normal scalar. Based on expected particle deceleration between the control volume center location and the wall

surface, the standard DF approximation is expected to significantly overestimate particle inertia at the wall and may be mesh dependent.

To improve performance of the standard DF approach, velocity corrections are proposed for particle conditions at the initial point of particle-to-wall contact. Instead of utilizing velocity at the near-wall cell center as with the standard DF model, the particle velocity at the wall is determined based on a subgrid solution of particle motion between the control volume center and wall location. The motivation behind this subgrid solution is that fully resolving near-wall finite particle inertia with a continuous model would require an excessive number of control volumes. Instead, the DF-VC model employs an analytic solution of particle velocity between the near-wall control volume center location and the wall (Figure 11.3). It is assumed that the continuous field model approximates particle diffusion within this region. Furthermore, for low particle Reynolds numbers the individual Lagrangian transport terms become linear and separable. Particle inertia between the control volume center and wall surface can then be approximated on a discrete Lagrangian basis as:

$$\frac{dv_{p,n}}{dt} = \frac{1}{\tau_p}(u_{p,n} - v_{p,n}) \quad (14)$$

To formulate an analytic solution of equation (14), the wall-normal fluid velocity is assumed to vary linearly between the control volume center and zero at the wall. The resulting equation for particle position between the near-wall control volume center and the wall as a function of time can be written as

$$\frac{d^2x_p}{dt^2} + \frac{1}{\tau_p} \frac{dx_p}{dt} + \frac{u_{cv,n}}{s\tau_p} x_p = \frac{1}{\tau_p} u_{cv,n} \quad (15)$$

where $u_{cv,n}$ is the fluid velocity at the control volume center location normal to the wall. An analytic solution to equation (15) is possible resulting in wall-normal expressions for particle position

$$x_p(t) = \frac{v_{cv,n} + \lambda_2 s}{\lambda_1 - \lambda_2} e^{\lambda_1 t} - \left(\frac{v_{cv,n} + \lambda_2 s}{\lambda_1 - \lambda_2} + s \right) e^{\lambda_2 t} + s \quad (16a)$$

and velocity

$$v_{p,n}(t) = \frac{v_{cv,n} + \lambda_2 s}{\lambda_1 - \lambda_2} \lambda_1 e^{\lambda_1 t} - \left(\frac{v_{cv,n} + \lambda_2 s}{\lambda_1 - \lambda_2} + s \right) \lambda_2 e^{\lambda_2 t} \quad (16b)$$

where

$$\lambda_1 = \frac{1}{2} \left(-\frac{1}{\tau_p} + \sqrt{\left(\frac{1}{\tau_p} \right)^2 - 4 \left(\frac{u_{cv,n}}{s\tau_p} \right)} \right) \quad (16c)$$

$$\lambda_2 = \frac{1}{2} \left(-\frac{1}{\tau_p} - \sqrt{\left(\frac{1}{\tau_p} \right)^2 - 4 \left(\frac{u_{cv,n}}{s\tau_p} \right)} \right) \quad (16d)$$

The time for initial wall contact can be determined from equation (16a). The associated particle velocity at the point of deposition is then calculated using equation (16b) and applied to calculate the local mass deposition in equation (12a).

The expressions for near-wall particle position and velocity presented as equations (16a and b) require the λ coefficients (equations 16c and d), which are roots of the characteristic equation for particle motion, to be real and unique. If these coefficients are complex numbers or repeated roots, then the general solution of near-wall particle motion will have a different form. The occurrence of real and unique values of λ_1 and λ_2 requires that

$$\left(\frac{1}{\tau_p}\right)^2 > 4 \left(\frac{u_{cv,n}}{s\tau_p}\right) \quad (17)$$

The particle response time (τ_p) is proportional to the particle diameter squared. Therefore, the condition for real and unique values of λ_1 and λ_2 is more likely to be satisfied for smaller particles. As the particle size increases, the left-hand side (LHS) of the condition decreases rapidly, which increases the probability for complex values of λ_1 and λ_2 . Considering submicrometer aerosols, the limiting maximum particle size is 1 μm , which should result in only real and unique values of λ_1 and λ_2 , as verified numerically [42].

11.5 Evaluation of the DF-VC Model in an Idealized Airway Geometry

In this section, three continuous field models were evaluated based on their ability to capture the inertial effects of submicrometer aerosols by direct comparisons with in vitro experimental data in a idealized bifurcation geometry. The first model considered was the Eulerian CS approximation, which is known to neglect particle inertia [20]. Implementation of this model is intended as a base case to capture purely diffusional effects. The remaining two models were based on a DF approach that accounts for both diffusion and particle inertia effects. Differences in the DF models arise as a result of how the near-wall inertia is approximated, as described in the previous section. The idealized airway bifurcation model was selected here as a test case because it can be mathematically described, which facilitates the generation of identical experimental and computational geometries.

The experimental method for the generation and delivery of submicrometer aerosols was previously described by Oldham et al. [15]. Briefly, a Lovelace-type compressed air nebulizer (In-Tox Products, Albuquerque, NM) was used to generate the fluorescent 400 nm and 1 μm aerosols. As shown in Figure 11.4a, a custom copper enclosure [51] was used to connect the nebulizer with the double bifurcation model. Aerosols were dried and diluted using 5% relative humidity air at an inject rate of 10 times the nebulizer output. The aerosol was discharged to Boltzmann equilibrium by passing through a ^{85}Kr discharger. Aerosols were pulled through the double bifurcation models by use of a vacuum pump. Particles that did not

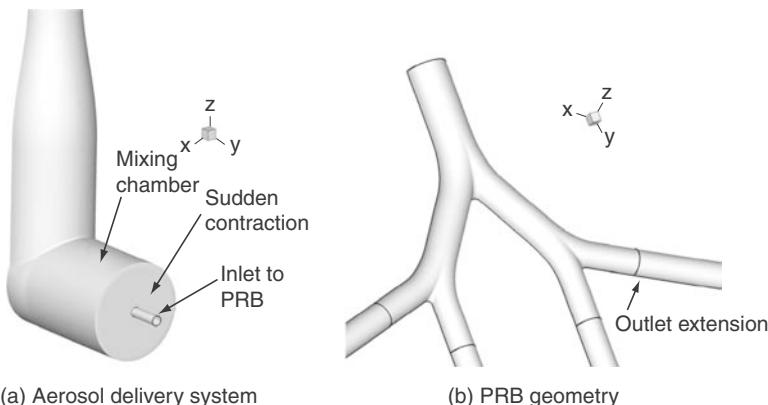


Figure 11.4. Geometries considered including (a) experimental particle delivery system, and (b) double bifurcation model of respiratory generations G3–G5.

deposit in the models were collected on a 25 mm diameter polycarbonate filter with a 0.1 μm pore size (Nucleopore Corporation, Pleasanton, CA). Local and total depositions of aerosols were determined by microscope-based counting on a grid of 1.4×0.95 mm for 1 μm particles and 1×0.69 mm for 400 nm particles [25,15]. Fluorescent particle counts in photographic microscope fields were determined using fluorescence microscopy.

Comparisons of experimental and numerical total particle deposition values in the bifurcation geometry for the two particle sizes considered are illustrated in Figure 11.5. For 400 nm particles, the experimental total deposition rate is 0.0054%. The CS model appears to closely match the experimental deposition value with a total deposition prediction of 0.0048%, resulting in a percent difference of 11.1%. In contrast, the standard DF model predicts a deposition rate that is nearly two orders of magnitude higher than the experimental findings. As a result, the standard DF model appears to significantly overpredict the effects of particle inertia. The DF-VC approximation matches the experimental results to a high degree with a total deposition value of 0.00585% and a percent difference 8.3%. As a result, both the CS and DF-VC models appear to provide adequate predictions of total deposition for 400 nm particles. Comparisons of local deposition results with experimental findings will be used to evaluate differences between the CS and DF-VC models.

Experimental deposition results for 1 μm particles indicate a total deposition fraction of 0.01% (Figure 11.5), as reported in Oldham et al. [15]. Due to a lack of particle inertia in the computational model, the CS approach under predicts the total deposition fraction for 1 μm aerosols by one order of magnitude. In contrast, the standard DF model overpredicts the total deposition by one order of magnitude. As with 400 nm particles, predictions of the DF-VC model for 1 μm particles are in close agreement with experimental results. The DF-VC model predicts a total deposition value of 0.0104%, resulting in a difference of 4.0% in comparison with the experiment.

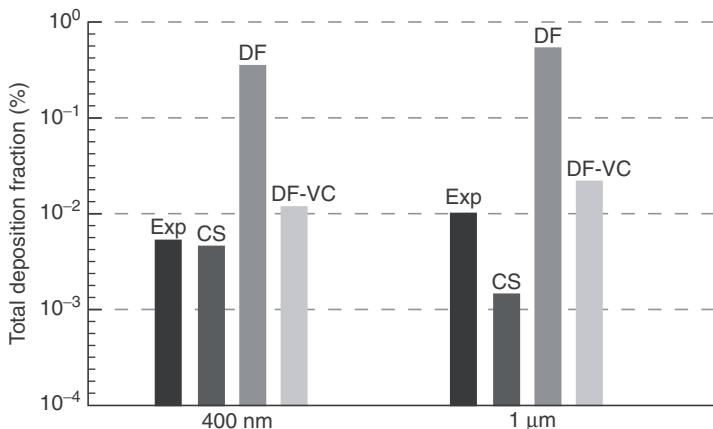


Figure 11.5. Comparison of total deposition fraction as a percentage between experimental results and model predictions for 400 nm and 1 μm particles.

Experimental results of local 400 nm particle depositions on a 1×0.69 mm grid are shown in Figure 11.6a as a percentage of total deposited particles [25]. As expected, localized increases in particle deposition are observed at the bifurcations. Deposition contours are in the range of 1 to 5% at the first carinal ridge and 0.1 to 1% at the second. A discontinuous shading pattern is observed in the first daughter branches with contours in the range of 0.01 to 1%. At the second bifurcation, higher contour values appear to extend down the inner branch in comparison with the outer branch. Some asymmetry in the deposition pattern may be due to minor variations in the inlet particle concentration and alignment of the grid field with the experimental model.

Numerical results of local particle deposition for 400 nm aerosols are reported on a two-dimensional grid with element sizes equivalent to the experimental study, i.e., 1×0.69 mm (Figure 11.6b-d). For the CS model and 400 nm particles, hot spots of local particle deposition are not observed at the carinal ridges (Figure 11.6b). As a result, the CS model appears to match the experimental total deposition value but displays significant differences from local experimental findings. Moreover, the CS model appears to predict more evenly distributed contours of deposition throughout the physiologically realistic bifurcation (PRB) in comparison with the experimental results. Considering the standard DF model, a significant increase in local particle deposition is observed at the first carinal ridge and a lesser hot spot occurs at the second bifurcation point (Figure 11.6c). However, these hot spots are at a higher contour level than observed in the experiment and surrounded by few deposited particles. The DF-VC model provides the best match to localized experimental results for the deposition of 400 nm particles (Figure 11.6d). Maximum localized particle deposition values at both carinal ridges are predicted to be within the same range as observed experimentally. Specially, the DF-VC model

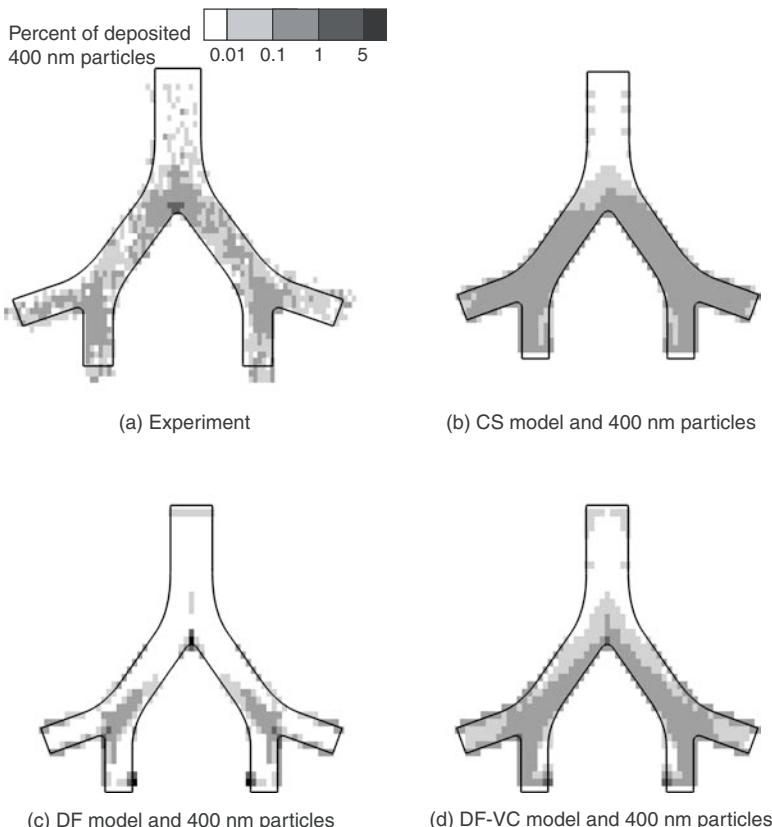


Figure 11.6. Comparison between experimental and numerically determined contours of local deposition for 400 nm particles expressed as a percentage of total deposition: (a) experimental, (b) CS, (c) DF, and (d) DF-VC models.

predicts localized deposition at the first and second carinal ridges to be 1 to 5% and 0.1 to 1%, respectively. Furthermore, the DF-VC model predicts elevated contours within the range of 0.1 to 1% extending down the inner branch of G5, as observed experimentally. Similar agreements between model and experimental results were also observed for 1 μm particles [25].

A significant finding from this direct comparison was that the CS and standard DF models did not effectively predict the deposition of fine respiratory aerosols. The CS model has been widely applied in respiratory dynamics systems for particles up to approximately 100 nm [20,21,52,53]. However, only a few available studies have applied a DF model for the simulation of respiratory aerosols [34,54]. No previous study has implemented a DF model to determine the inertial deposition of fine respiratory aerosols. Based on the available two-phase flow literature, the standard DF model should adequately model the deposition of dilute fine respiratory

aerosols based on diffusional and inertial transport mechanisms [32,33]. However, results of this study indicated a significant overprediction of respiratory aerosol deposition with use of the standard DF formulation. This overestimation was due to significant changes in particle slip velocity between the near-wall control volume center location and conditions at particle-to-wall contact.

11.6 Evaluation of the DF-VC Model in Realistic Airways

In this section, the DF-VC model was extended to transient conditions and was tested in more complex geometries of the respiratory tract for laminar and turbulent flows. Particle transport and deposition were evaluated in a computational replica of the TB geometry and an MRI-based nasal model. Numerical results were compared to existing experimental data in these models. A standard CS model was also considered to evaluate the extent to which the DF-VC model captures the effects of particle inertia. To determine the influence of time effects on particle deposition, steady and transient flow fields were evaluated. These studies are intended to further develop a highly effective method for the simulation of fine respiratory aerosols in realistic models of the upper airway.

11.6.1 Tracheobronchial region

A hollow cast of the human TB tree (Figure 11.7a) utilized by Cohen et al. [55] was digitally replicated in this study to ensure a direct comparison with experimental deposition results. The geometric parameters of the cast were in agreement with population means of a representative average adult male. The original cast used in the study of Cohen et al. [55] was scanned by a multirow-detector helical CT scanner (GE medical systems, Discovery LS) with the following acquisition parameters: 0.7 mm effective slice spacing, 0.65 mm overlap, 1.2 mm pitch, and 512×512 pixel resolution. The multislice CT images were then imported into MIMICS (Materialise, Ann Arbor, MI) to convert the raw image data into a set of cross-sectional contours that define the solid geometry. Based on these contours, a surface geometry was manually constructed in Gambit 2.3 (Ansys, Inc.) (Figure 11.7b and c). Some distal branches in the range of generations G5 and G6 were not retained in the digital model due to low resolution. Most of the digital model paths extended from the trachea to generation G4 with some paths extending to generations G5 and G6. Twenty-three outlets and a total of 44 bronchi were retained in the final computational model (Figure 11.7b). The surface geometry was then imported into ANSYS ICEM 10 (Ansys, Inc.) as an IGES file for meshing. To avoid excessive grid elements, some minor smoothing of the geometric surface was necessary.

The left-right asymmetry, which is an important feature of the human lung, was preserved in the TB model. There are three lobes in the right lung and two lobes in the left. As observed in the TB model, the left main bronchus is approximately 2.5 times longer than the right (Figure 11.7b), which is consistent with van Erbruggen et al. [56]. For conducting airways, the bifurcating pattern is typically asymmetric

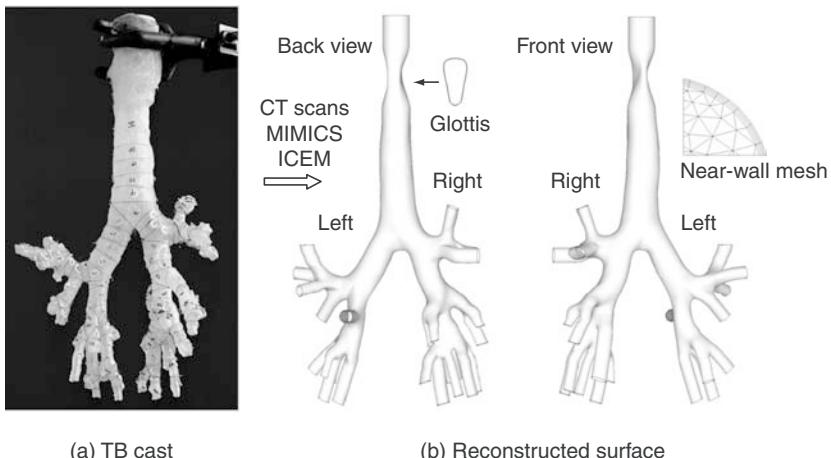


Figure 11.7. Development of a tracheobronchial airway model from a replica cast of the human upper respiratory tract: (a) the hollow cast utilized by Cohen et al. [14] showing the divisions of sub-branch segments. A mechanical larynx was connected to the cast in order to generate cyclic inspiratory flows. (The figure was provided by Dr. Beverly Cohen, NYU School of Medicine.) (b) A surface geometry was constructed based on CT scans of the cast and software packages MIMICS and Gambit. Using ANSYS ICEM 10, the computational mesh was generated and consisted of unstructured tetrahedral control volumes with a very fine near-wall grid of prism elements.

with daughter branches from the same parent often differing both in diameter and length. Additionally, the spatial orientations of the bifurcating branches are quite variable, resulting in a lung architecture that is highly out-of-plane.

A critical feature of the physiologically realistic TB model used in this study is the inclusion of a laryngeal approximation and wedge-shaped glottis, as shown in Figure 11.7b and c. Including a laryngeal approximation provides a better representation of in vivo conditions and is significant in deposition studies with TB casts [57]. Studies by Chan et al. [58], Gurman et al. [59], Martonen et al. [60], and recently Xi et al. [35] have highlighted the significance of the laryngeal jet on downstream deposition in the respiratory tract. The transient breathing conditions of Cohen et al. [55] were approximated as a sinusoidal function with the format

$$Q(t) = Q_{\text{in}} [1 - \cos(2\omega t)]; \quad u(t) = u_{\text{mean}} [1 - \cos(2\omega t)] \quad (18\text{a and b})$$

where ω is the breathing frequency in radians/s.

Compared with the idealized bifurcation model, interesting flow phenomena are observed due to the complex geometry characteristics of the TB model such as the presence of the larynx, left-right asymmetry, and non-coplanar bifurcations. The laryngeal jet that forms at the glottis aperture extends downstream through the

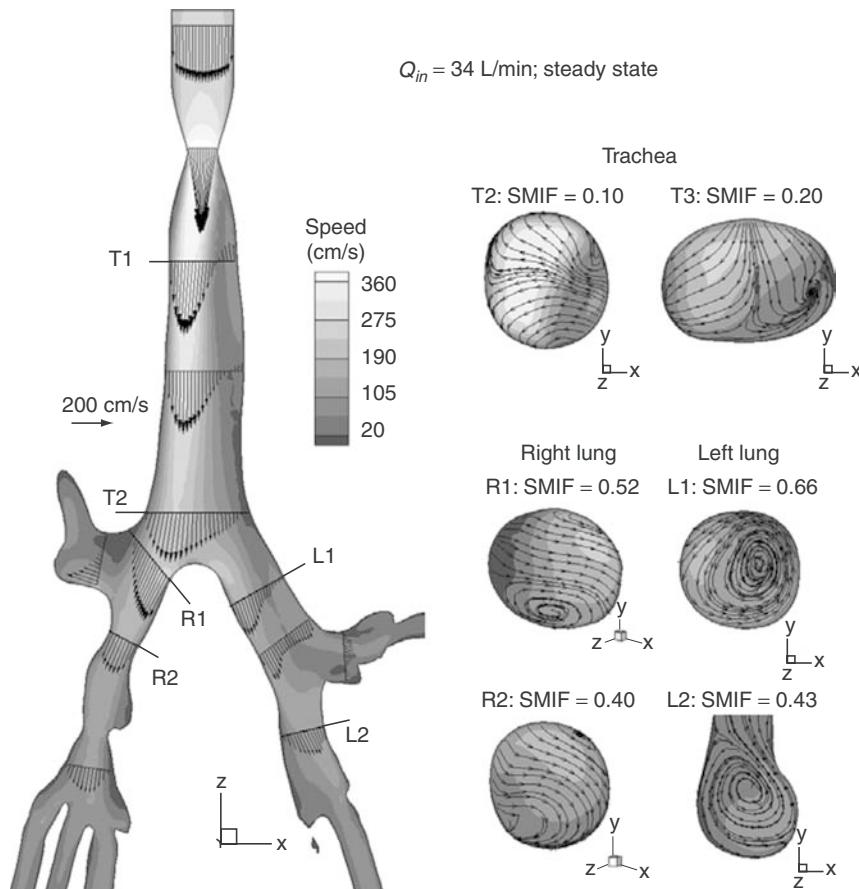


Figure 11.8. Midplane velocity vectors, contours of velocity magnitude, and in-plane streamlines of secondary motion in the tracheobronchial model under steady conditions with an inspiratory flow rate of 34 L/min. The slices are not to scale.

trachea and is skewed toward the right wall (Figure 11.8). Flow reversal occurs in the trachea due to the strong jetting effect. As a result, a large recirculation zone develops near the left tracheal wall, which significantly reduces the cross-sectional area available for expansion of the high-speed flow. Similar results have been reported by Corcoran and Chigier [61]. Using Laser Doppler Velocimetry and fluorescent dye, Corcoran and Chigier [61] also observed a skewed laryngeal jet in the right anterior trachea and a region of reverse flow in the left trachea with a cast model. The skewed jet feature may have physiological implications that facilitate efficient mixing and deeper penetration of inhaled oxygen into the lung. Conversely, higher particle deposition rates are expected on the right tracheal wall and in the right downstream branches due to impaction.

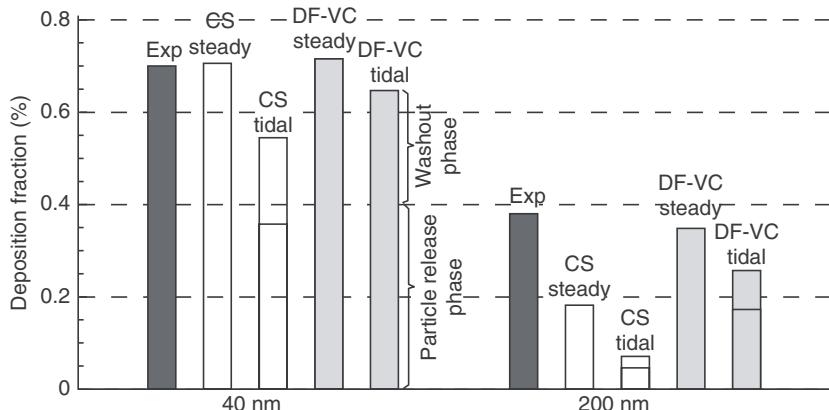


Figure 11.9. Comparison of total deposition fraction as a percentage between the experimental results of Cohen et al. [14] and model predictions under steady and transient conditions for 40 and 200 nm particles.

For transient inlet conditions of a mean flow rate of 34 L/min, the Womersley number (α) ranges from 4.2 to 0.8 in the conducting airways of G0 to G6, indicating moderate to small unsteady effects. As expected, transient flow patterns are similar to steady state conditions for a major portion of the breathing cycle (i.e., 0.15T to 0.85T), resulting in what can be considered as a quasi-steady system [42]. In contrast, during the transition phase between breathing cycles when the velocity is small, the fluid experiences dramatic changes in both mainstream flow and secondary motion [35,42]. Similarly, Kabilan et al. [62] examined the flow characteristics in a CT-based ovine lung model. Their results suggested that the onset of the transient phase of the flow might be the main cause of vorticity formation, which plays an important role in gas mixing. Li et al. [63] also reported that particle deposition has a strong dependence on individual phases of the transient waveform.

Comparisons of experimental and numerical values of total deposition in the TB model geometry for 40 and 200 nm particles are illustrated in Figure 11.9. The experimental deposition values were reported by Cohen et al. [55] for cyclic breathing with a mean flow rate of 34 L/min and a breathing frequency of 20 breaths per minute. The transient numerical deposition fraction is a cumulative value that includes both the particle-release phase (2nd cycle) and the following washout phase (3rd and 4th cycles). The first cycle is airflow only, without particles, in order to establish the transient flow field within the TB geometry. In the case of 40 nm particles, the experimental total deposition rate is 0.7%. Both the CS and DF-VC models closely match the experimental deposition value under steady conditions with a total deposition prediction of 0.706 and 0.714%, respectively. The marginally higher total deposition value of the DF-VC model indicates a negligible contribution of inertia to total deposition for 40 nm particles. In contrast, both models appear to underestimate the experimental result for transient conditions, resulting in a percent error of -48.9% for the transient CS model and -21.1% for the transient DF-VC model.

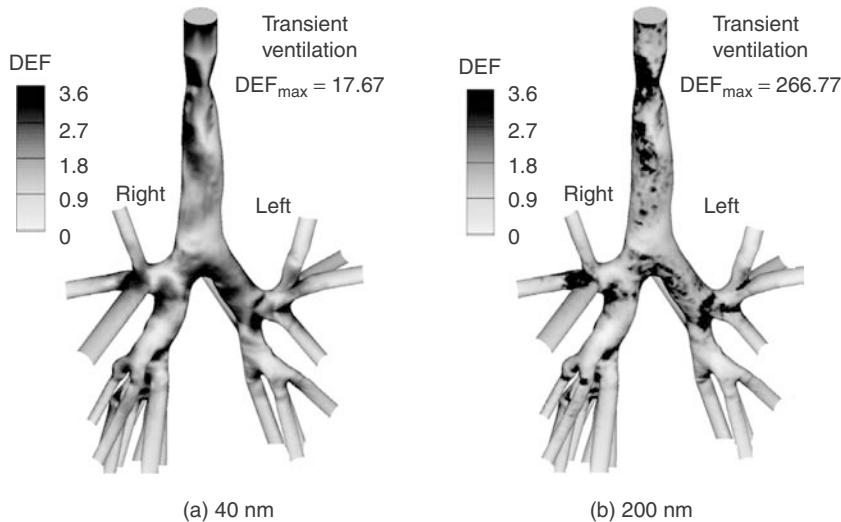


Figure 11.10. Numerically determined deposition enhancement factors (DEF) in the tracheobronchial airway under transient conditions at a mean flow rate of 34 L/min with the DF-VC model for (a) 40 nm and (b) 200 nm particles.

Experimental cyclic deposition results for 200 nm particles indicate a total deposition fraction of 0.38% (Figure 11.9). Due to a lack of particle inertia, the CS model significantly under predicts the total deposition fraction for 200 nm aerosols, resulting in percent errors of -52.1% for steady ventilation and -81.3% for cyclic ventilation. The DF-VC model provides a substantial improvement over the CS approach by incorporating the particle inertia forces in the computational model. As with 40 nm particles, predictions of the DF-VC model for 200 nm particles with steady flow are in close agreement with the experimental results, with a minor underestimation of -8.4%. For cyclic conditions, the DF-VC approach predicts a total deposition of 0.257%, resulting in a difference of -32.4% in comparison with the experiment. Comparing these two models, differences between the CS predictions and the experimental results are significant for 200 nm particles, indicating a significant contribution from particle inertia. Still, an under prediction of deposition for 200 nm particles is observed with the DF-VC model. As noted before, some distal branches of the TB cast were not included in the numerical model and a static open glottis was employed, which may contribute to this under prediction of the experimental results for both steady and transient inhalation conditions.

Figure 11.10 illustrates the DEF value for the DF-VC model under cyclic conditions with a mean flow rate of 34 L/min. For the DF-VC model, the deposition patterns under steady and cyclic conditions are similar in appearance for 40 and 200 nm particles, respectively. However, more concentrated local deposition values in the larynx and bifurcations are obtained under cyclic conditions compared with steady state. Specifically, the maximum DEF value for 200 nm aerosols with cyclic flow (i.e., 267) is about seven times higher than with the steady condition (i.e., 37),

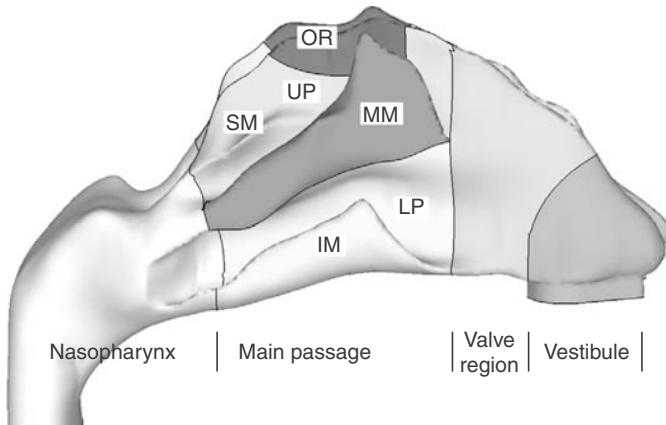


Figure 11.11. Surface geometry of a MRI-based human nasal cavity. The nasal airway was divided into different anatomical sections to examine sub-regional particle deposition. Sections include the vestibule, nasal valve region, olfactory region (OR), septum wall (SW), middle meatus (MM), as well as the upper (UP) and lower (LP) passages that include the superior (SM) and inferior (IM) meatus, respectively, and the nasopharynx.

indicating a dramatic increase of the velocity-dependent inertia effect on particle localization. For the smaller 40 nm particles, the maximum DEF value of approximately 18 for cyclic inhalation is only twice as large as the steady state value of 8.3. As a result, transient conditions have a greater effect on the deposition of a 200 nm aerosol, in comparison with 40 nm particles, due to inertial effects.

Geometric complexity, turbulence, and transient flow are all expected to significantly influence the deposition of aerosols in the respiratory tract [38,64–66]. In this context, an extended DF-VC model was evaluated by comparing with experimental results in a complex TB geometry with transient and laminar-to-turbulent flows. Through comparison with the CS model, the DF-VC approach was shown to better capture the expected effects of particle inertia on a total and local basis.

11.6.2 Nasal cavity

The DF-VC model was further tested in a human nasal airway, which was based on MRI scans of a healthy non-smoking 53-year-old male (weight 73 kg and height 173 cm). The procedure for developing the nasal airway geometry and mesh is similar to the approach applied for the TB model in the previous section. As shown in Figure 11.11, the nasal cavity is characterized by narrow, convoluted, and multilayer channels called meatus. This complex passageway generates unique aerodynamics inside the nasal cavity and helps the nose to accomplish its physiological functions. Conversely, reasons for deficient nasal function can often be traced back to abnormal

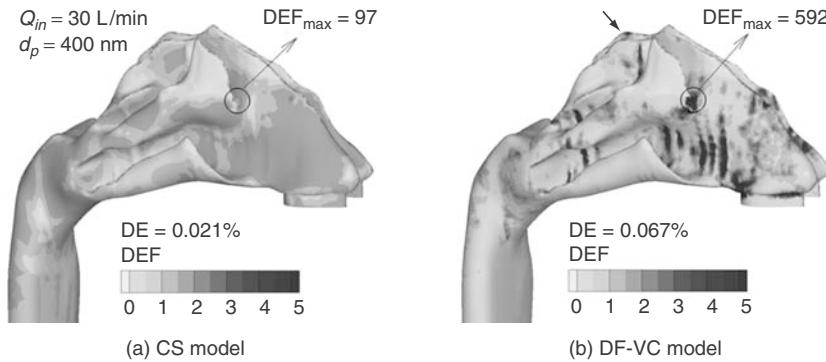


Figure 11.12. Numerically determined deposition enhancement factors (DEF) in the nasal airway for 400 nm particles at an inhalation flow rate of 30 L/min using the (a) CS and (b) DF-VC models.

global and local flow conditions and turbulent phenomena in the nasal passages. In an adult, 18,000 to 20,000 L of air pass through the nose each day. Besides warming and moistening the inhaled ambient air, the nasal cavity also houses olfactory sensory receptors, filters out airborne pollutants, drains excess sinus secretions, and balances pressure between the middle ear and atmosphere.

As in the previous section, two continuous field particle transport models, i.e., CS and DF-VC models, are considered. To highlight the effect of finite particle inertia on deposition localization, a comparison of DEFs predicted using the CS and DF-VC models are shown in Figure 11.12 for 400 nm particles at an inhalation flow rate of 30 L/min. With DEF values plotted on the same scale, the difference between these two models is striking. In contrast to the more uniformly distributed DEF values of the CS model, the deposition with the DF-VC model is significantly more heterogeneous and localized. Specifically, at the anterior junction point (solid circle) between the middle meatus and medial passage where high-speed flow and steep geometry transition occur, the hot spot of the DEF value predicted by the DF-VC model is about five times higher than the CS value. Another expected hot spot is at the superior part of the vestibule (dashed square), which is captured by the DF-VC model. In contrast, the CS model does not indicate this region as having elevated deposition. Additionally, elevated deposition accumulations are also predicted around the rear olfactory region (filled arrow) based on the DF-VC model due to trajectory deviations of large particles from curved streamlines in the main airflow. These elevated localizations may have important implications in chemical sensing applications or nasal drug delivery for neurological disorders where the olfactory region is the targeted deposition site.

The effectiveness of the DF-VC model in capturing inertial and diffusive deposition is further illustrated in Figure 11.13 in terms of deposition within specific sections of the nose. The definition of each section is depicted in Figure 11.11

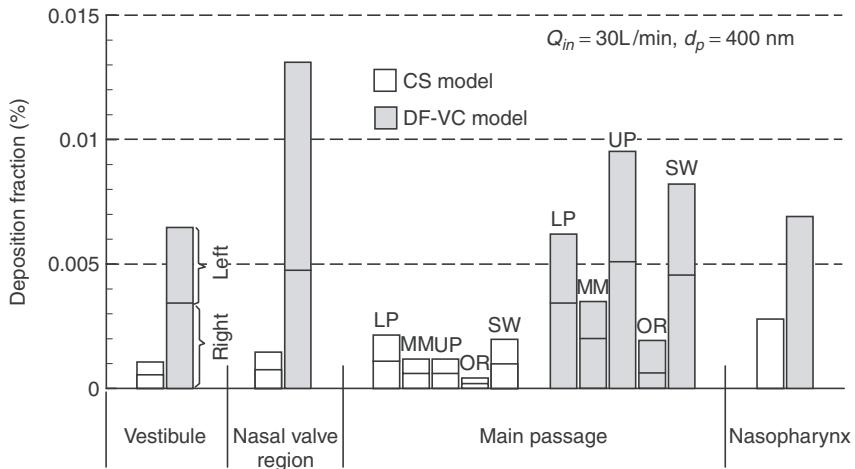


Figure 11.13. Comparison of deposition fractions between the CS and DF-VC models in different nasal sections. The magnitude of the deposition fraction values in the anterior and main nasal airway is the summation of the right (lower bar) and left (upper bar) nasal passages.

and the inhalation conditions are identical to those shown in Figure 11.12 (i.e., $Q_{in} = 30 \text{ L/min}$ and 400 nm particles). The magnitude of the deposition fraction value for each region (except for the nasopharynx) represents the summation of right (lower bar) and left (upper bar) nasal passages. The deposition fraction values for the CS model denote deposition from diffusion only, while the difference between the DF-VC and CS models can be viewed as deposition from inertial impaction. In the vestibule, where the direction of airflow changes by approximately 90° , significantly enhanced deposition is predicted by the DF-VC model, which is about 6.1 times the CS model estimate for identical conditions. An even more pronounced effect of inertial impaction (i.e., an order of magnitude increase) is found in the nasal valve region where the cross-sectional area is minimal and the airflow passages are narrowest. In this region, the valve-associated stenosis, flow acceleration, and short distances to the walls combine to effectively increase the inertial deposition of particles entrained in the flow. The inertial effect on deposition in the main passage remains significant, but to a lesser degree in comparison with the vestibule and nasal valve regions, which is likely due to the increased flow area (decreased flow speed) and less severe streamline curvatures. Within the main passage, diffusion is enhanced in the middle and inferior meatus by the slow-moving flows (and the resulting prolonged particle residence times) as well as the large surface areas of these two fin-like projections available for particle contact. As a result, the deposition fraction of the DF-VC model is only 3.0 times that of the CS model for the middle meatus and 2.9 times larger for the lower passage.

Figure 11.14a shows the DF-VC simulation results of *inspiratory* deposition fractions in comparison to replica measurements as a function of a diffusion

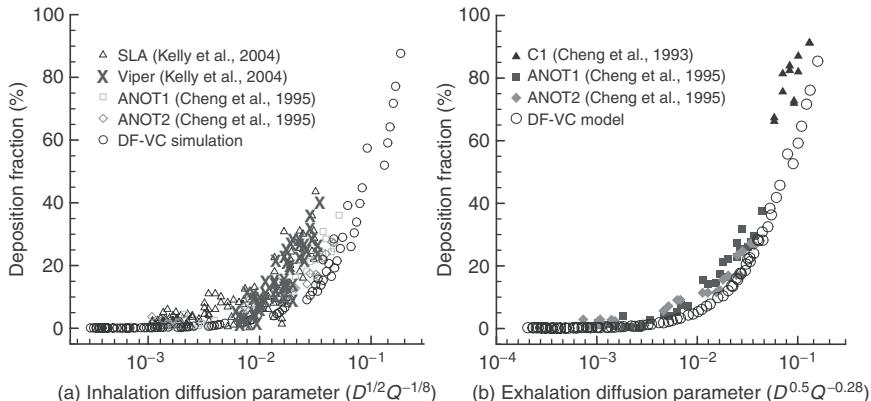


Figure 11.14. Predicted deposition rates with the DF-VC model in comparison with existing in vitro experimental data for particles ranging from 1 to 1,000 nm and inhalation flow rates ranging from 4 to 45 L/min: (a) inspiratory, (b) expiratory. Units: Q [L/min] and D [cm^2/s].

parameter suggested by Cheng et al. [67]. Inhalation flow rates that were considered included 4, 7.5, 10, 15, 20, and 30 L/min with particle sizes ranging from 1 to 1,000 nm. The correlated association of deposition fraction with the parameter ($D^{1/2}Q^{-1/8}$) indicates a stronger dependence of nasal deposition on particle diffusivity (exponent of 1/2) than on flow rate (exponent – 1/8) for the submicrometer particles considered.

Figure 11.14b shows the DF-VC simulation results of *expiratory* deposition fractions in comparison with existing in vitro deposition data. Simulated expiratory deposition rates with the discrete Lagrangian tracking is also superimposed in this figure, which closely matches the continuous-phase DF-VC results. From Figure 11.14b, reasonably good agreement is observed between simulations and in vitro measurements for the three replica casts considered with slight underestimation to the in vitro data. Specifically, the closely correlated association of deposition fraction with the parameter ($D^{0.5}Q^{-0.28}$) indicates a stronger dependence of nasal deposition on particle diffusivity (exponent of 0.5) than on flow rate (exponent –0.28) for the submicrometer particles considered.

As described earlier, the DF-VC approximation accounts for both particle inertia and diffusion, whereas the CS model only considers particle diffusion. As a result, the increased deposition of the DF-VC model over the CS approach can be attributed entirely to particle inertial effects. For ultrafine particles where inertial effects are negligible, the Sherwood number Sh is nearly identical for the two models considered (Figure 11.15a). Deviation of Sh between the DF-VC and CS models begins at $Re^{0.55}Sc^{0.60} = 1.5 \times 10^4$ (equivalently, $St_k = 1.0 \times 10^{-5}$) and becomes progressively significant with increasing values of $Re^{0.55}Sc^{0.60}$ (or St_k). Compared with the inertial limit of $St_k = 5.0 \times 10^{-5}$ for TB airways [20], a smaller value in

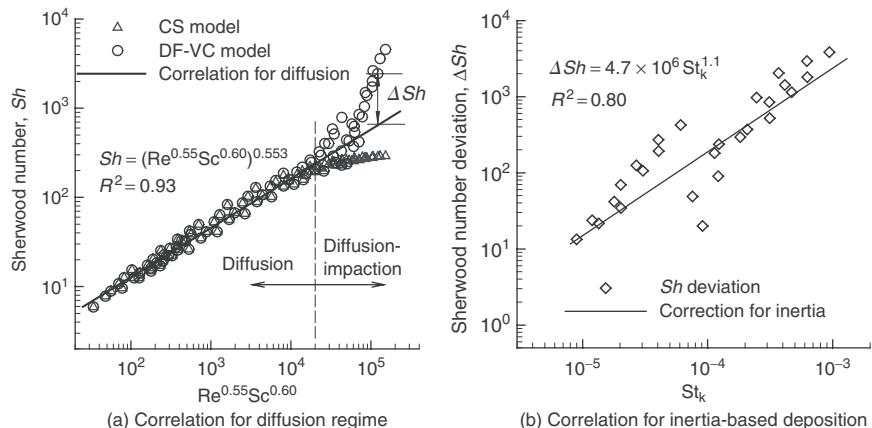


Figure 11.15. Development of a Sherwood number Sh correlation in the nasal airway that accounts for both diffusional and inertial deposition mechanisms for submicrometer aerosols: (a) correlation for ultrafine particles where diffusion dominates deposition and (b) correlation for enhanced mass transport due to particle inertia, ΔSh , as a function of Stokes number, St_k .

the nasal cavity indicates an earlier onset of particle inertial effects, which may be due to the high complexity of this geometry.

While the nasal deposition can be reasonably predicted by existent empirical correlations in the literature, these expressions are typically limited to ultrafine or coarse particles where either diffusion or inertial impaction are the predominate mechanisms for particle loss. Therefore, a correlation that is valid for both diffusional and inertial deposition regimes is of value to inhalation toxicologists and is sought in this study based on numerical experiments. Figure 11.15 shows the development of a mass transfer correlation in the nasal cavity in terms of the non-dimensional Sherwood number. The dependence of mass transfer on convective diffusion is plotted in Figure 11.15a with a best-fit correlation for the diffusion zone given by

$$Sh = (Re^{0.55} Sc^{0.60})^{0.553} = Re^{0.30} Sc^{0.33} \quad (St_k \leq 1.0 \times 10^{-5}) \quad (19)$$

The convective-diffusion coefficient ($Re^{0.55} Sc^{0.60}$) adopted here was suggested by Cheng et al. [68] for in vivo nasal deposition data. For fine respiratory particles that are influenced by diffusion and impaction deposition mechanisms, the deviation from the above equation (ΔSh , see Figure 11.15a) due to particle inertia can be correlated as a function of St_k using (Figure 11.15b)

$$\Delta Sh = 4.7 \times 10^6 St_k^{1.1} \quad (20)$$

Based on the assumption of weak coupling between the inertial and diffusive deposition mechanisms, the overall correlation that is valid for all submicrometer particle sizes can be obtained by adding equations (19) and (20) as:

$$Sh = Re^{0.30} Sc^{0.33} + 4.7 \times 10^6 St_k^{1.1} \quad (d_p = 1 - 1,000 \text{ nm}) \quad (21)$$

Alternatively, the above correlation can be expressed in terms of total nasal deposition fraction,

$$Df = 1 - \exp \left[- \left(\frac{A_s}{\bar{A}_c} \right) \left(Re^{-0.70} Sc^{-0.67} + 4.7 \times 10^6 \frac{St_k^{1.1}}{Re Sc} \right) \right] \quad (22)$$

where A_s is the total surface area and \bar{A}_c the mean cross-sectional area of the nasal passage. The above equation is valid for particle size ranging from 1 to 1,000 nm.

11.7 Discussion

Numerical simulations of fine respiratory aerosols are challenging due to low deposition efficiencies and the action of concurrent inertial and diffusive deposition mechanisms. The CS Eulerian model that is typically applied for ultrafine aerosols is highly efficient and shows reasonable agreement with analytic solutions for ultrafine diffusive deposition in a tubular geometry [17,20,69]. However, the typical CS Eulerian model neglects the effects of finite particle inertia [20,32] and cannot be used to predict fine particles in the range of 100 to 1,000 nm. As presented in this chapter, finite particle inertia may significantly affect the total deposition of submicrometer aerosols as small as 70 nm and the local deposition of particles as small as 40 nm [25,40,42]. Based on the available two-phase flow literature, the standard DF model should adequately model the deposition of dilute fine respiratory aerosols based on diffusional and inertial transport mechanisms [32,33]. However, results presented by Longest and Oldham [25] and summarized above indicated a significant overprediction of respiratory aerosol deposition with the use of the standard DF formulation. This overestimation was due to significant changes in particle slip velocity between the near-wall control volume center location and conditions at particle-to-wall contact. To improve the simulations of submicrometer aerosols in the respiratory tract, a hybrid DF-VC model was developed and tested both experimentally and numerically [25,40,42]. In comparison with the CS model, deposition results of the DF-VC approach persistently agreed better with experimental findings on a total and subbranch basis, which indicated that the DF-VC model effectively captured the influence of finite particle inertia. Results of this study indicate that a DF particle transport model with near-wall velocity corrections can provide an effective approach for simulating the transport and deposition of submicrometer respiratory aerosols in complex human respiratory airways.

While the DF-VC model has been shown to be effective for submicrometer respiratory aerosols, it does have several limitations. The current DF-VC model was

developed mainly for submicrometer aerosols based on the use of the pressure gradient term in equation (11b). This model can also be extended to larger particle sizes if equation (11a) is used to compute particle slip. Another limitation of the general DF approach is the assumption of a dilute aerosol in which the momentum of the discrete phase does not influence the flow field. As a result, low loadings of particle concentration are required, which is generally the case for respiratory aerosols. Finally, the current model was developed for monodisperse aerosol distributions.

The results of the studies summarized in this chapter should not be interpreted to imply that the continuous field DF-VC model is superior to Lagrangian particle tracking for simulating aerosol transport in all situations. A primary advantage of Lagrangian particle tracking is the direct treatment of particles as individual or potentially interacting discrete elements, in contrast with the approximation of a continuous aerosol field. Treating particles as discrete elements allows for the direct evaluation of individual force terms that represent a number of physical phenomena such as lift, near-wall lubrication, and particle-to-particle interactions [29]. Furthermore, Lagrangian particle tracking can be directly applied to non-spherical shapes, like rotating fibers, and polydisperse aerosols. In contrast to the Lagrangian approach, the DF-VC model approximates finite particles as a continuous species. As a result, transport and deposition are evaluated from the solution of a convection-diffusion equation, which may introduce numerical dissipation errors into the solution. Furthermore, it does not appear that the DF-VC model can effectively account for particle-to-particle interactions or the motion and interception of complex shapes, like rotating fibers. A primary advantage of the DF-VC model is high efficiency, where successive additions of discrete particles are not required to establish converged local deposition characteristics. As a result of these observations, it is clear that both the continuous field DF-VC model and Lagrangian particle tracking have inherent strengths and weaknesses. Further testing and evaluation is required to extend the flexibility of the DF-VC model and to overcome some current limitations, such as monodisperse aerosols. It is likely that selection of Eulerian versus Lagrangian models for submicrometer aerosol transport will remain dependent on the particular problem of interest in conjunction with the advantages and disadvantages of each method. However, results of this study indicate that the DF-VC model does provide an effective approach that can simulate the concurrent action of diffusion and inertia in a complex transient and laminar-to-turbulent flow field.

In conclusion, a continuous-phase DF particle transport model with a near-wall velocity correction provided an effective solution for the deposition of submicrometer aerosols under transient and laminar-to-turbulent flow conditions. Comparisons with a standard CS model indicated that the DF-VC approach was incorporating the effects of finite particle inertia in determining total and local deposition characteristics for the complex TB and nasal geometries that were considered. Comparisons of steady and transient conditions indicated that cyclical flow decreased the total deposition of submicrometer particles but significantly increased relative particle localization, as indicated with the DEF parameter. Future studies are necessary to address wall surface roughness and compliance, mucus clearance, and real-life

breathing waveforms before the current DF model can be directly applied to make dose-response and health effects predictions.

References

- [1] Morawska, L., Hofmann, W., Hitchins-Loveday, J., Swanson, C., and Mengersen, K. Experimental study of the deposition of combustion aerosols in the human respiratory tract, *Journal of Aerosol Science*, 36, pp. 939–957, 2005.
- [2] Kittelson, D. B. Engines and nanoparticles: A review, *Journal of Aerosol Science*, 29(5–6), pp. 575–588, 1998.
- [3] Bernstein, G. M. A review of the influence of particle size, puff volume, and inhalation pattern on the deposition of cigarette smoke particles in the respiratory tract, *Inhalation Toxicology*, 16, pp. 675–689, 2004.
- [4] Mandell, G. L., Bennett, J. E., and Bolin, R. D. *Principles and Practices of Infectious Diseases*, 6th ed., Churchill Livingstone, New York, 2004.
- [5] ICRP. *Human Respiratory Tract Model for Radiological Protection*, Elsevier Science Ltd., New York, 1994.
- [6] Chan, T. L., and Lippmann, M. Experimental measurements and empirical modeling of the regional deposition of inhaled particles in humans, *American Industrial Hygiene Association Journal*, 41, pp. 399–409, 1980.
- [7] Heyder, J., Gebhart, J., Rudolf, G., Schiller, C. F., and Stahlhofen, W. Deposition of particles in the human respiratory tract in the size range of 0.005–15 microns, *Journal of Aerosol Science*, 17(5), pp. 811–825, 1986.
- [8] Stahlhofen, W., Rudolf, G., and James, A. C. Intercomparison of experimental regional aerosol deposition data, *Journal of Aerosol Medicine*, 2(3), pp. 285–308, 1989.
- [9] Binkova, B., Giguere, Y., Rossner, P., Dostal, M., and Sram, R. J. The effect of dibenzo[a,l]pyrene and benzo[a]pyrene on human diploid lung fibroblasts: The induction of DNA adducts, expression of p53 and p21 WAF1 proteins and cell cycle distribution, *Mutation Research*, 471, pp. 57–70, 2000.
- [10] Pandya, R. J., Solomon, G., Kinner, A., and Balmes, J. R. Diesel exhaust and asthma: Hypotheses and molecular mechanisms of action, *Environmental Health Perspectives*, 110, pp. 103–112, 2002.
- [11] Balashazy, I., Hofmann, W., and Heistracher, T. Computation of local enhancement factors for the quantification of particle deposition patterns in airway bifurcations, *Journal of Aerosol Science*, 30, pp. 185–203, 1999.
- [12] Bell, K. A., and Friedlander, S. K. Aerosol deposition in models of a human lung bifurcation, *Staub-Reinhaltuder Luft*, 33, pp. 183–187, 1973.
- [13] Longest, P. W., Vinchurkar, S., and Martonen, T. B. Transport and deposition of respiratory aerosols in models of childhood asthma, *Journal of Aerosol Science*, 37, pp. 1234–1257, 2006.
- [14] Martonen, T. B., Rosati, J. A., and Isaacs, K. K. Modeling deposition of inhaled particles, in *Aerosols Handbook*, L. S. Ruzer & N. H. Harley, eds., CRC Press, New York, pp. 113–155, 2005.
- [15] Oldham, M. J., Phalen, R. F., and Heistracher, T. Computational fluid dynamic predictions and experimental results for particle deposition in an airway model, *Aerosol Science and Technology*, 32(1), pp. 61–71, 2000.

- [16] Schlesinger, R. B., and Lippmann, M. Particle deposition in casts of the human upper tracheobronchial tree, *American Industrial Hygiene Association Journal*, 33, pp. 237–251, 1972.
- [17] Zhang, Z., Kleinstreuer, C., Donohue, J. F., and Kim, C. S. Comparison of micro- and nano-size particle depositions in a human upper airway model, *Journal of Aerosol Science*, 36(2), pp. 211–233, 2005.
- [18] Balashazy, I., Hofmann, W., and Heistracher, T. Local particle deposition patterns may play a key role in the development of lung cancer, *Journal of Applied Physiology*, 94, pp. 1719–1725, 2003.
- [19] Kimbell, J. S., Subramaniam, R. P., Gross, E. A., Schlosser, P. M., and Morgan, K. T. Dosimetry modeling of inhaled formaldehyde: Comparisons of local flux predictions in the rat, monkey, and human nasal passages, *Toxicological Sciences*, 64(1), pp. 100–110, 2001.
- [20] Longest, P. W., and Xi, J. Computational investigation of particle inertia effects on submicron aerosol deposition in the respiratory tract, *Journal of Aerosol Science*, 38(1), pp. 111–130, 2007.
- [21] Shi, H., Kleinstreuer, C., Zhang, Z., and Kim, C. S. Nanoparticle transport and deposition in bifurcating tubes with different inlet conditions, *Physics of Fluids*, 16(7), pp. 2199–2213, 2004.
- [22] Phalen, R. F., Oldham, M. J., and Nel, A. E. Tracheobronchial particle dose considerations for in vitro toxicology studies, *Toxicological Sciences*, 92(1), pp. 126–132, 2006.
- [23] Longest, P. W., and Oldham, M. J. Mutual enhancements of CFD modeling and experimental data: A case study of one micrometer particle deposition in a branching airway model, *Inhalation Toxicology*, 18(10), pp. 761–772, 2006.
- [24] Robinson, R. J., Oldham, M. J., Clinkenbeard, R. E., and Rai, P. Experimental and numerical smoke carcinogen deposition in a multi-generation human replica tracheobronchial model, *Annals of Biomedical Engineering*, 34(3), pp. 373–383, 2006.
- [25] Longest, P. W., and Oldham, M. J. Numerical and experimental deposition of fine respiratory aerosols: Development of a two-phase drift flux model with near-wall velocity corrections, *Journal of Aerosol Science*, 39, pp. 48–70, 2008.
- [26] Hofmann, W., Golser, R., and Balashazy, I. Inspiratory deposition efficiency of ultrafine particles in a human airway bifurcation model, *Aerosol Science and Technology*, 37(12), pp. 988–994, 2003.
- [27] Ingham, D. B. Diffusion of aerosols in the entrance region of a smooth cylindrical pipe, *Journal of Aerosol Science*, 22(3), pp. 253–257, 1991.
- [28] Martonen, T. B., Zhang, Z., and Yang, Y. Particle diffusion with entrance effects in a smooth-walled cylinder, *Journal of Aerosol Science*, 27(1), pp. 139–150, 1996.
- [29] Longest, P. W., Kleinstreuer, C., and Buchanan, J. R. Efficient computation of micro-particle dynamics including wall effects, *Computers & Fluids*, 33(4), pp. 577–601, 2004.
- [30] Longest, P. W., and Xi, J. Effectiveness of direct Lagrangian tracking models for simulating nanoparticle deposition in the upper airways, *Aerosol Science and Technology*, 41, pp. 380–397, 2007.
- [31] Fernandez de la Mora, J., and Rosner, D. E. Effects of inertia on the diffusional deposition of small particles to spheres and cylinders at low Reynolds numbers, *Journal of Fluid Mechanics*, 125, pp. 379–395, 1982.
- [32] Friedlander, S. K. *Smoke, Dust and Haze: Fundamentals of Aerosol Dynamics*, 2 ed., Oxford University Press, New York, 2000.

- [33] Manninen, M., Taivassalo, V., and Kallio, S. On the mixture model for multiphase flow, *VTT Publications 288, Technical Research Center of Finland*, 1996.
- [34] Wang, J. B., and Lai, A. C. K. A new drift-flux model for particle transport and deposition in human airways, *Journal of Biomechanical Engineering*, 128, pp. 97–105, 2006.
- [35] Xi, J., Longest, P. W., and Martonen, T. B. Effects of the laryngeal jet on nano- and microparticle transport and deposition in an approximate model of the upper tracheobronchial airways, *Journal of Applied Physiology*, 104, pp. 1761–1777, 2008.
- [36] Ghalichi, F., Deng, X., Champlain, A. D., Douville, Y., King, M., and Guidoin, R. Low Reynolds number turbulence modeling of blood flow in arterial stenoses, *Biorheology*, 35(4–5), pp. 281–294, 1998.
- [37] Wilcox, D. C. *Turbulence Modeling for CFD*, 2nd ed., DCW Industries, Inc., CA, 1998.
- [38] Xi, J., and Longest, P. W. Transport and deposition of micro-aerosols in realistic and simplified models of the oral airway, *Annals of Biomedical Engineering*, 35(4), pp. 560–581, 2007.
- [39] Xi, J., and Longest, P. W. Effects of oral airway geometry characteristics on the diffusional deposition of inhaled nanoparticles, *ASME Journal of Biomechanical Engineering*, 130, p. 011008, 2008.
- [40] Xi, J., and Longest, P. W. Numerical predictions of submicrometer aerosol deposition in the nasal cavity using a novel drift flux approach, *International Journal of Heat and Mass Transfer*, 51, pp. 5562–5577, 2008.
- [41] Longest, P. W., and Vinchurkar, S. Validating CFD predictions of respiratory aerosol deposition: effects of upstream transition and turbulence, *Journal of Biomechanics*, 40, pp. 305–316, 2007.
- [42] Xi, J., and Longest, P. W. Evaluation of a drift flux model for simulating submicrometer aerosol dynamics in human upper tracheobronchial airways, *Annals of Biomedical Engineering*, 36(10), pp. 1714–1734, 2008.
- [43] Crowe, C. T., Troutt, T. R., and Chung, J. N. Numerical models for two-phase turbulent flows, *Annual Review of Fluid Mechanics*, 28, pp. 11–43, 1996.
- [44] Allen, M. D., and Raabe, O. G. Slip correction measurements of spherical solid aerosol particles in an improved Millikan apparatus, *Aerosol Science and Technology*, 4, pp. 269–286, 1985.
- [45] Hinds, W. C. *Aerosol Technology: Properties, Behavior, and Measurement of Airborne Particles*, Wiley, New York, 1999.
- [46] Li, A., and Ahmadi, G. Dispersion and deposition of spherical particles from point sources in a turbulent channel flow, *Aerosol Science and Technology*, 16, pp. 209–226, 1992.
- [47] Matida, E. A., Finlay, W. H., and Grgic, L. B. Improved numerical simulation of aerosol deposition in an idealized mouth-throat, *Journal of Aerosol Science*, 35, pp. 1–19, 2004.
- [48] Lumb, A. B. *Nunn's Applied Respiratory Physiology*, Butterworth Heinemann, Oxford, 2000.
- [49] Xi, J., and Longest, P. W. Effects of improved near-wall modeling on micro-particle deposition in oral airway geometries, *Proceedings of the 2007 ASME Summer Bioengineering Conference*, Paper No. SBC2007-176227, Keystone, CO, 2007.
- [50] Cohen, B. S., and Asgharian, B. Deposition of ultrafine particles in the upper airways: An empirical analysis, *Journal of Aerosol Science*, 21, pp. 789–797, 1990.

- [51] Oldham, M. J., Mannix, R. C., and Phalen, R. F. Deposition of monodisperse particles in hollow models representing adult and child-size tracheobronchial airways, *Health Physics*, 72(6), pp. 827–833, 1997.
- [52] Broday, D. M. Deposition of ultrafine particles at carinal ridges of the upper bronchial airways, *Aerosol Science and Technology*, 38, pp. 991–1000, 2004.
- [53] Zhang, Z., and Kleinstreuer, C. Airflow structures and nano-particle deposition in a human upper airway model, *Journal of Computational Physics*, 198(1), pp. 178–210, 2004.
- [54] Chen, F., and Lai, A. C. K. An Eulerian model for particle deposition under electrostatic and turbulent conditions, *Journal of Aerosol Science*, 35, pp. 47–62, 2004.
- [55] Cohen, B. S., Sussman, R. G., and Lippmann, M. Ultrafine particle deposition in a human tracheobronchial cast, *Aerosol Science and Technology*, 12, pp. 1082–1093, 1990.
- [56] van Ertbruggen, C., Hirsch, C., and Paiva, M. Anatomically based three-dimensional model of airways to simulate flow and particle transport using computational fluid dynamics, *Journal of Applied Physiology*, 98(3), pp. 970–980, 2005.
- [57] Pedley, T. J. Pulmonary fluid dynamics, *Annual Review of Fluid Mechanics*, 9, pp. 229–274, 1977.
- [58] Chan, T. L., Schreck, R. M., and Lippmann, M. Effect of the laryngeal jet on particle deposition in the human trachea and upper bronchial airways, *Journal of Aerosol Science*, 11, pp. 447–459, 1980.
- [59] Gurman, J. L., Lippmann, M., and Schlesinger, R. B. Particle deposition in replicate casts of the human upper tracheobronchial tree under constant and cyclic inspiratory flow. I. Experimental, *Aerosol Science and Technology*, 3, pp. 245–252, 1984.
- [60] Martonen, T. B., Zhang, Z., and Lessmann, R. Fluid dynamics of the human larynx and upper tracheobronchial airways, *Aerosol Science and Technology*, 19, pp. 133–144, 1993.
- [61] Corcoran, T. E., and Chigier, N. Inertial deposition effects: A study of aerosol mechanics in the trachea using laser Doppler velocimetry and fluorescent dye, *Journal of Biomechanical Engineering*, 124, pp. 629–637, 2002.
- [62] Kabilan, S., Lin, C. L., and Hoffman, E. A. Characteristics of airflow in a CT-based ovine lung: A numerical study, *Journal of Applied Physiology*, 102, pp. 1469–1482, 2007.
- [63] Li, Z., Kleinstreuer, C., and Zhang, Z. Particle deposition in the human tracheobronchial airways due to transient inspiratory flow patterns, *Journal of Aerosol Science*, 38, pp. 625–644, 2007.
- [64] Lin, C. L., Tawhai, M. H., McLennan, G., and Hoffman, E. A. Characteristics of the turbulent laryngeal jet and its effect on airflow in the human intra-thoracic airways, *Respiratory Physiology and Neurobiology*, 157, pp. 295–309, 2007.
- [65] Nowak, N., Kakade, P. P., and Annapragada, A. V. Computational fluid dynamics simulation of airflow and aerosol deposition in human lungs, *Annals of Biomedical Engineering*, 31, pp. 374–390, 2003.
- [66] Zhang, Z., Kleinstreuer, C., and Kim, C. S. Cyclic micron-size particle inhalation and deposition in a triple bifurcation lung airway model, *Journal of Aerosol Science*, 33(2), pp. 257–281, 2002.
- [67] Cheng, K. H., Cheng, Y. S., Yeh, H. C., and Swift, D. L. Deposition of ultrafine aerosols in the head airways during natural breathing and during simulated breath-holding using replicate human upper airway casts, *Aerosol Science and Technology*, 23(3), pp. 465–474, 1995.

- [68] Cheng, Y. S., Yeh, H. C., Guilmette, R. A., Simpson, S. Q., Cheng, K. H., and Swift, D. L. Nasal deposition of ultrafine particles in human volunteers and its relationship to airway geometry, *Aerosol Science and Technology*, 25(3), pp. 274–291, 1996.
- [69] Shi, H., Kleinstreuer, C., and Zhang, Z. Laminar airflow and nanoparticle or vapor deposition in a human nasal cavity model, *Journal of Biomechanical Engineering*, 128, pp. 697–706, 2006.

This page intentionally left blank

12 Algorithm stabilization and acceleration in computational fluid dynamics: exploiting recursive properties of fixed point algorithms

Aleksandar Jemcov and Joseph P. Maruszewski
ANSYS, Inc., Canonsburg, PA, USA

Abstract

Convergence acceleration of nonlinear flow solvers through use of techniques that exploit recursive properties fixed-point methods of computational fluid dynamics (CFD) algorithms is presented. It is shown that nonlinear iterative algorithms create Krylov subspace that can be considered linear within the second order of accuracy in the vicinity of the local solutions. Moreover, repeated application of the fixed-point iterations give rise to various vector extrapolation methods that are used to accelerate underlying iterative method. In particular, suitability of the reduced rank extrapolation (RRE) algorithm for the use of convergence acceleration of nonlinear flow solvers is examined. In the RRE algorithm, the solution is obtained through a linear combination of Krylov vectors with weighting coefficients obtained by minimizing L_2 norm of error in this space with properly chosen constraint conditions. This process effectively defines vector sequence extrapolation process in Krylov subspace that corresponds to the GMRES method applied to nonlinear problems. Moreover, when the RRE algorithm is used to solve nonlinear problems, the flow solver plays the role of the preconditioner for the nonlinear GMRES method. Benefits of the application of the RRE algorithm include better convergence rates, removal of residual stalling, and improved coupling between equations in numerical models. Proposed algorithm is independent of the type of flow solver and it is equally applicable to explicit-, implicit-, pressure-, and density-, based algorithms. RRE algorithm can also be considered as an generalization of matrix-free algorithms that are used extensively in CFD.

Keywords: Fixed-point methods, Krylov subspace, Recursive properties, Reduced rank extrapolation, Vector extrapolation

12.1 Introduction

Computational fluid dynamics (CFD) in modern engineering applications is characterized by complex physical models, complex shapes of computational domain, and large mesh sizes. Governing equations of fluid flow and associated equations for the transport of conserved variables are seemingly simple and yet they contain very prominent nonlinearities of which the most famous one is manifested through the phenomena of turbulence. The need to model turbulent flows coupled with radiation, heat and mass transfer, and multiphase interactions lead to very complex physical models that contain many nonlinear terms and wide range of spatial and temporal scales. This complexity directly translates into difficulties in numerical applications and they are loosely referred to here as computational stiffness (for the current purposes of the qualitative discussion, we will use the term stiffness to signify computational difficulties). Another source of stiffness in CFD is associated with the use of large mesh sizes in order to capture relevant spatial scales associated with physical models. It is well-known fact that the condition number of linear system increases with the increase of the mesh size as it can be verified by the uniform refinement of the computational mesh. Furthermore, topological complexities of the computational domains lead CFD practitioners to the use of unstructured meshes with nearly arbitrary shapes of control volumes. The main issues of unstructured meshes such as presence of highly skewed volumes, high aspect ratios close to the boundaries, loss of orthogonality, and lack of convexity directly contribute to the stiffness of the computational problem. This is usually manifested in the loss of the diagonal dominance of the matrix in the linear system of discrete equations. Iterative solvers for linear systems of equations are not particularly effective when the matrix of the linear system departs strongly from the diagonal dominance expressed through M -matrix condition.

Complexity is not just restricted to physical models and related issues as described above; flow simulation is also characterized by a great number of differing algorithms ranging from explicit to fractional step to implicit methods [1]. Furthermore, depending on the choice of the solution variables and chosen discrete formulation, pressure [2–4] and density-based solvers [5,6] can be distinguished. Pressure- and density-based solvers can be formulated as implicit, explicit, or fractional step methods. Complexity of the landscape is further increased by the choice of linear solver used in implicit algorithms. Additionally, various flux formulations are in the use today including Riemann solver, flux vector splitting, central difference with dissipation flux formulation [6], to name just a few. Each of these algorithms have control parameters that have optimal values for a given application area thus producing an optimal convergence rate during the iterative process. Examples of control parameters are CFL numbers in density-based and relaxation factors in pressure-based algorithms.

Optimal range of control parameters are defined by the stability constraints of the numerical algorithm, underlying physics, computational mesh, geometrical complexity of the computational domain, and boundary conditions. Due to the non-linear nature of equations describing the flow phenomena, selection of the values

of control parameters that are outside of the optimal range results in either slow convergence rates or divergence of the residual norm. Linear stability analysis [5] provides some guidelines for the selection of the optimal values of control parameters but in the industrial applications of flow solvers, problem complexity due to physics, domain geometry, boundary conditions, etc. dramatically changes the optimal range of control parameters. Furthermore, the nonlinear nature of governing equations changes the optimal values of control parameters during the iterative process. During initial iterations, depending on the initial guess, optimal values are very different from the optimal values of the control parameters during later stages in iterative process. Usually, initially conservative values of control parameters are used to maintain the stability of the iterative process and as iterations approach the solution, more aggressive values can be used.

The need to dynamically change the values of control parameters during iterations is addressed in algorithms that compute optimal or nearly optimal values by utilizing the knowledge of the state of computed variables and some criterion that measures stability of the computed state. Continuation methods [7] provide a possible solution to this problem. Another approach to the problem of optimal control parameters is in the use of backtracking or line search algorithm in conjunction with Newton methods [8]. Both continuation and Newton with backtracking algorithms are effective tools for the solution of the problem of dynamically changing control parameters but they require algorithmic changes to the flow solver. Many times it is not feasible to implement new algorithms in large codes, and even though the new algorithms provide added stability and increase convergence rates, they cannot be implemented in timely manner. Less intrusive ways of improving the convergence rates and overall stability of the flow solver are required.

Another source of instability and low convergence rates in flow solvers is related to artificial segregation of discrete equations in the numerical model. Example of the algorithms that use segregated approach are the SIMPLE family of algorithms [2,4]. Another example of segregated algorithms is found in the treatment of turbulence models [9] in density-based algorithms where they are lagged behind flow quantities. In multi-equation turbulence models, each turbulent transport equation may be solved separately, thus artificially decoupling equations in the numerical model. Decoupling of equations in segregated flow solvers has a detrimental impact on convergence rates. The solution to this problem is usually found in creation of coupled solvers at the expense of increased memory and code complexity.

Vector sequence extrapolation methods [10] offer an alternative solution to convergence acceleration and stabilization of flow solvers. The idea of vector sequence extrapolation is based on the generalization of scalar sequence convergence acceleration methods, such as Richardson extrapolation [10], to vector sequences. Vector sequences are produced by iterative procedures of solver algorithms while seeking the solution of the given flow problem. In addition to accelerating properties of vector sequence extrapolation methods, they also exhibit coupling properties since only one set of extrapolation coefficients is calculated for all fields. This has a positive effect on convergence rates, very similar to the action of Krylov subspace solvers acting on systems with block coefficients. Moreover, formal equivalence

between vector sequence extrapolation methods to Krylov subspace solvers such as GMRES and FOM [11] exists.

Early examples of the application [13] of vector sequence methods in CFD was the application of reduced rank extrapolation (RRE), minimal polynomial extrapolation (MPE), and modified minimal polynomial extrapolation (MMPE) to explicit flow solvers based on JST scheme [12] and implicit flow solver with flux vector splitting formulation. Acceleration of convergence rates of inviscid flow solvers was examined [13] and marginal improvements were obtained with explicit solvers, whereas implicit solvers showed better results. Another application [14] of an extrapolation-related technique to several types of flow demonstrated the suitability of extrapolation to accelerate inviscid compressible and viscous laminar incompressible flows. The method [14] used in the studies relied on residual minimization and required many evaluations of residuals making this method expensive. Other applications of vector sequence extrapolation to nonlinear problems include Chandrasekhar's H-equation [15]. Vector sequence extrapolation is also used for acceleration of stationary iterative methods [16] and the algebraic multigrid (AMG) method [17].

Traditional view of extrapolation methods applied to vector sequences is that they perform the transformation of one sequence into another sequence resulting with faster convergence rates. Since a flow solver will be generating vector sequences that will be used to compute extrapolation coefficients, an alternative view of this process is that the flow solver plays the role of a nonlinear preconditioner for an extrapolation solver. Due to equivalence of RRE and GMRES algorithms, the proposed acceleration technique corresponds to a nonlinearly preconditioned GMRES solver.

The outline of the paper is as follows. CFD algorithms were examined from the fixed-point algorithm point of view and the recursive property of these algorithms is shown. Recursive property of the fixed-point algorithms represent, to second-order approximation, a power iteration with the matrix of Jacobian of the fixed-point function. This property plays an important role in the derivation of vector sequence extrapolation algorithms. This is followed by a description of the vector sequence extrapolation method where the recursive property of the fixed-point function was used to obtain the extrapolation coefficients. This is done through the formulation of an optimization problem in the Krylov subspace and a proper choice of constraints lead to the RRE algorithm. In the section on numerical experiments, the application of nonlinear acceleration is demonstrated on various CFD problems utilizing several flow solvers. Finally, a summary and conclusion is presented, following results of numerical experiments.

12.2 Iterative Methods for Flow Equations

In this section we will consider the discrete form of the governing equations and iterative methods of finding the solution for nonlinear problems in CFD. Iterative methods are considered from the point of view of the fixed-point algorithms that

have a recursive property related to powers of the Jacobian of the fixed-point function. Recursive property plays important role in establishing the existence of Krylov subspace in which the solution of the problem is approximated. In fact, we demonstrate how the recursive property generates Krylov subspace for any fixed-point function thus enabling acceleration and stabilization of arbitrary iterative problems under the certain conditions.

12.2.1 Discrete form of governing equations

Consider Navier–Stokes system of equations in integral form

$$\frac{\partial}{\partial t} \int_{\Omega} \mathbf{Q} d\Omega + \oint_{\partial\Omega} (\mathbf{F}_c - \mathbf{F}_v) dA - \int_{\Omega} \mathbf{S} d\Omega = 0 \quad (1)$$

Here \mathbf{Q} represents vector of conserved quantities, \mathbf{F}_c and \mathbf{F}_v are convective and viscous flux vectors, and \mathbf{S} is the source term vector

$$\mathbf{Q} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix}, \quad \mathbf{F}_c = \begin{pmatrix} \rho V \\ \rho uV + n_x p \\ \rho vV + n_y p \\ \rho wV + n_z p \\ \rho HV \end{pmatrix}, \quad \mathbf{F}_v = \begin{pmatrix} 0 \\ n_x \tau_{xx} + n_y \tau_{xy} + n_z \tau_{xz} \\ n_x \tau_{yx} + n_y \tau_{yy} + n_z \tau_{yz} \\ n_x \tau_{zx} + n_y \tau_{zy} + n_z \tau_{zz} \\ n_x \Theta_x + n_y \Theta_y + n_z \Theta_z \end{pmatrix}$$

$$\mathbf{S} = \begin{pmatrix} 0 \\ \rho f_{e,x} \\ \rho f_{e,y} \\ \rho f_{e,z} \\ \rho(f_{e,x}u + f_{e,y}v + f_{e,z}w) + \dot{q}_h \end{pmatrix}$$

and V and Θ_x , Θ_y , and Θ_z are defined as:

$$V = n_x u + n_y v + n_z w$$

$$\Theta_x = u\tau_{xx} + v\tau_{xy} + w\tau_{xz} + k \frac{\partial T}{\partial x}$$

$$\Theta_y = u\tau_{yx} + v\tau_{yy} + w\tau_{yz} + k \frac{\partial T}{\partial y}$$

$$\Theta_z = u\tau_{zx} + v\tau_{zy} + w\tau_{zz} + k \frac{\partial T}{\partial z}$$

The system of equations in equation (1) may include additional scalar or vector transport equations

$$\frac{\partial}{\partial t} \int_{\Omega} \Phi d\Omega + \oint_{\partial\Omega} (\mathbf{F}_{\Phi,c} - \mathbf{F}_{\Phi,v}) dA - \int_{\Omega} \mathbf{S}_{\Phi} d\Omega = 0 \quad (2)$$

Depending on the choice of the transported quantity, equation (2) can represent species, passive scalar, turbulent transport, etc. Appropriate boundary conditions [1] must be also supplied to define a well-posed problem that can be solved numerically.

The first step in obtaining the discretized equations is to define a residual $\mathbf{R}(\mathbf{Q})$

$$\mathbf{R}(\mathbf{Q}) = \oint_{\partial\Omega} (\mathbf{F}_c - \mathbf{F}_v) dA - \int_{\Omega} \mathbf{S} d\Omega \quad (3)$$

Equation (3) defines a continuous residual that can be discretized on a computational mesh using a cell-centered finite volume method. For i th finite volume cell, discrete residual is given by:

$$\mathbf{R}_i(\mathbf{Q}) = \sum_f (\mathbf{F}_c^f - \mathbf{F}_v^f)_i - \mathbf{S}_i \quad (4)$$

Here symbol $\sum_f (\cdot)$ represents summation over all faces of a given finite volume cell i . If the discrete value of the conserved variables vector \mathbf{Q}_i is defined as:

$$\mathbf{Q}_i = \frac{\int_{\Omega_i} \mathbf{Q} d\Omega}{\Omega_i} \quad (5)$$

The discrete representation of the Navier–Stokes system in equation (1) becomes

$$\frac{d(\Omega\mathbf{Q})_i}{dt} = -\mathbf{R}_i(\mathbf{Q}) \quad (6)$$

The discrete system of equations given by equation (6) represents the nonlinear discrete system due to nonlinear nature of the residual \mathbf{R}_i . An explicit algorithm can use the system in equation (6) directly by applying the time marching procedure such as the multistage Runge–Kutta [1] procedure and full approximation scheme (FAS) [18] to obtain the solution of the problem. The explicit algorithm is given by the following expression:

$$\begin{aligned} \Delta\mathbf{Q}_i^{(v+1)} &= -\frac{\Delta t}{(\Omega)_i} \mathbf{R}_i(\mathbf{Q}^{(v)}) \\ \mathbf{Q}^{(v+1)} &= \mathbf{Q}^{(v)} + \Delta\mathbf{Q}^{(v)} \end{aligned} \quad (7)$$

In the case of an implicit algorithm, a Newton or Picard linearization of the residual is applied to form an iterative scheme. If Newton linearization is used, the discretized system of equations is given by the following expression:

$$\begin{aligned} \left[\frac{(\Omega)_i}{\Delta t} + (\partial_Q \mathbf{R})_i \right] \Delta\mathbf{Q}_i^{(v)} &= -\mathbf{R}_i(\mathbf{Q}^{(v)}) \\ \mathbf{Q}^{(v+1)} &= \mathbf{Q}^{(v)} + \Delta\mathbf{Q}^{(v)} \end{aligned} \quad (8)$$

Since the solution of the system in equation (8) requires inversion of matrix on the left-hand side, iterative methods for solving large sparse systems of linear equations are used. Popular choices for linear solvers are Krylov subspace methods [11] and AMG method [18]. There are also number of choices for the convective and viscous flux evaluations at cell faces, popular choices in density-based solvers are flux difference splitting and flux vector splitting [1], whereas in the case of pressure-based solvers, collocated schemes with Rhie–Chow dissipation [4] are often used.

12.2.2 Recursive property of iterative methods

Both explicit and implicit algorithms are particular instances of the fixed-point method that can be generally expressed as:

$$\mathbf{Q}^{(v+1)} = \mathbf{Q}^{(v)} - \mathcal{M}^{(v)}(\mathbf{R}(\mathbf{Q}^{(v)})) \quad (9)$$

The general form of any fixed-point method is

$$\mathbf{Q}^{(v+1)} = \mathbf{F}(\mathbf{Q}^{(v)}) \quad (10)$$

with fixed-point function $\mathbf{F}(\mathbf{Q}^{(v)})$

$$\mathbf{F}(\mathbf{Q}^{(v)}) = \mathbf{Q}^{(v)} - \mathcal{M}^{(v)}(\mathbf{R}(\mathbf{Q}^{(v)})) \quad (11)$$

Operator $\mathcal{M}^{(v)}(\mathbf{R}(\mathbf{Q}^{(v)}))$ is, in principle, a nonlinear operator that may not even be known explicitly. In the case of stationary linear iteration schemes such as Gauss–Seidel, Jacobi, and ILU(n), this operator may be constructed explicitly. However, in the case of the AMG method with multiple coarse levels, it is usually known implicitly through implementation in the flow solver. In the case of nonlinear flow solvers, this operator is almost always defined algorithmically through code implementation except in trivial cases. However, as it will be shown later, this knowledge is not required for the vector sequence extrapolation method to work.

An important property of any fixed-point algorithm is the recursive relation that can be established between the initial solution correction $\Delta\mathbf{Q}^{(0)}$ and later corrections $\Delta\mathbf{Q}^{(v)}$ through powers of the fixed-point function Jacobian $\partial_Q \mathbf{F}$. This relation can be established by forming the following difference:

$$\Delta\mathbf{Q}^{(v)} = \mathbf{Q}^{(v+1)} - \mathbf{Q}^{(v)} \quad (12)$$

From the definition of the fixed-point algorithm, equation (10), equation (12) becomes

$$\Delta\mathbf{Q}^{(v)} = \mathbf{F}(\mathbf{Q}^{(v)}) - \mathbf{F}(\mathbf{Q}^{(v-1)}) \quad (13)$$

By the definition, $\mathbf{Q}^{(v)}$ is given by

$$\mathbf{Q}^{(v)} = \mathbf{Q}^{(v-1)} + \Delta\mathbf{Q}^{(v-1)} \quad (14)$$

and if this expression is substituted in equation (13), we arrive to expression

$$\Delta \mathbf{Q}^{(v)} = \mathbf{F}(\mathbf{Q}^{(v-1)} + \Delta \mathbf{Q}^{(v-1)}) - \mathbf{F}(\mathbf{Q}^{(v-1)}) \quad (15)$$

Using the linear expansion of the term $\mathbf{F}(\mathbf{Q}^{(v-1)} + \Delta \mathbf{Q}^{(v-1)})$ in the previous equation, we obtain

$$\mathbf{F}(\mathbf{Q}^{(v-1)} + \Delta \mathbf{Q}^{(v-1)}) = \mathbf{F}(\mathbf{Q}^{(v-1)}) + \partial_Q \mathbf{F}(\mathbf{Q}^{(v-1)}) + \mathcal{O}(\varepsilon^2) \quad (16)$$

Neglecting the terms of higher order, expression in equation (16) becomes

$$\mathbf{F}(\mathbf{Q}^{(v-1)} + \Delta \mathbf{Q}^{(v-1)}) \approx \mathbf{F}(\mathbf{Q}^{(v-1)}) + \partial_Q \mathbf{F}(\mathbf{Q}^{(v-1)}) \quad (17)$$

Substituting equation (17) into equation (15), expression linking $\Delta \mathbf{Q}^{(v)}$ and $\Delta \mathbf{Q}^{(v-1)}$ is formulated

$$\Delta \mathbf{Q}^{(v)} \approx \partial_Q \mathbf{F}(\mathbf{Q}^{(v-1)}) \Delta \mathbf{Q}^{(v-1)} \quad (18)$$

By letting the index v to range from 0 to $v + 1$, equation (18) is used recursively to obtain the link between $\Delta \mathbf{Q}^{(0)}$ and $\Delta \mathbf{Q}^{(v+1)}$

$$\Delta \mathbf{Q}^{(v+1)} \approx (\partial_Q \mathbf{F}(\mathbf{Q}^{(v)}))^v \Delta \mathbf{Q}^{(0)} \quad (19)$$

The recursive property of the fixed-point algorithm is given by equation (19) and upon closer inspection it can be seen that it represents, a power iteration of $\Delta \mathbf{Q}^{(v+1)}$ with matrix $\partial_Q \mathbf{F}(\mathbf{Q}^{(v)})$, within second order of accuracy. Moreover, it can also be observed that the sequence of $\Delta \mathbf{Q}^{(v+1)}$ vectors will tend to lie in the dominant eigenspace of matrix $\partial_Q \mathbf{F}(\mathbf{Q}^{(v)})$ and also define the basis of the Krylov subspace associated with the iterative procedure described by the fixed-point method, equation (9). This property is used to approximate the solution of the fixed-point problem by using the minimal polynomial of the Jacobian matrix $\partial_Q \mathbf{F}(\mathbf{Q}^{(v)})$. A Krylov subspace generated by the iterative process defined by equation (10) can be defined formally by using the local linearization of the fixed-point function

$$\mathcal{K}^n(\mathcal{M}^{-1}(\partial_Q \mathbf{R}), \mathbf{R}^{(0)}) = \text{span}(\mathbf{R}^{(0)}, (\mathcal{M}^{-1}(\partial_Q \mathbf{R}))\mathbf{R}^{(0)}, \dots, (\mathcal{M}^{-1}(\partial_Q \mathbf{R}))^{n-1}\mathbf{R}^{(0)}) \quad (20)$$

Since in equation (20) operator \mathcal{M} is kept constant in the local neighborhood of solution vectors, Krylov subspace \mathcal{K}^n can be shown to be equivalent [14,15] to the space spanned by correction vectors $\Delta \mathbf{Q}^{(v)}$

$$\begin{aligned} \mathcal{K}^n(\mathcal{M}^{-1}(\partial_Q \mathbf{R}), \mathbf{R}^{(0)}) &= \mathcal{M}^{-1} \mathcal{K}^n(\Delta \mathbf{Q}^{(n)}) \\ &= \mathcal{M}^{-1} \text{span}(\Delta \mathbf{Q}^{(0)}, \Delta \mathbf{Q}^{(1)}, \dots, \Delta \mathbf{Q}^{(n)}) \end{aligned} \quad (21)$$

For strongly nonlinear fixed-point function, the local approximation used in equation (21) is not valid and the space generated by corrections $\Delta \mathbf{Q}^{(v)}$ is no longer

equivalent to the classical Krylov subspace. However, the solution to the fixed-point problem can be still sought in this space although convergence cannot be guaranteed.

12.3 Reduced Rank Extrapolation

After n iterations, vector sequence extrapolation searches for the solution of the fixed-point problem, equation (10), in the space defined in expression equation (21) by approximating the solution by the following linear combination:

$$\mathbf{Q}^{(n)} = \mathbf{Q}^{(n-1)} + \sum_{v=0}^n \alpha_v \mathbf{e}^{(v)} \quad (22)$$

Here α_v are extrapolation coefficients that need to be determined, \mathbf{e} is error after n iterations defined as

$$\mathbf{e}^{(v)} = \mathbf{Q}^{(v)} - \mathbf{Q}^* \quad (23)$$

and \mathbf{Q}^* is the solution (fixed point) of the problem. Error $\mathbf{e}^{(v)}$ shares the recursive property defined in equation (19)

$$\Delta \mathbf{e}^{(v+1)} \approx (\partial_Q \mathbf{F}(\mathbf{Q}^{(v)}))^v \Delta \mathbf{e}^{(0)} \quad (24)$$

Equation (24) represents the approximate error propagation equation of the given fixed-point function and it follows from the linearity of the recursive relation, equation (19). The recursive property of the error, equation (24), is used to define extrapolation coefficients. To achieve that, first we must find what conditions error must satisfy.

At convergence, the following identity holds

$$\mathbf{Q}^* = \mathbf{Q}^* + \sum_{v=0}^n \alpha_v \mathbf{e}^{(v)} \quad (25)$$

due to the fact that the error \mathbf{e} is zero. Therefore, at convergence after n iterations, the following identity must be true:

$$\sum_{v=0}^n \alpha_v \mathbf{e}^{(v)} = 0 \quad (26)$$

Equation (26) represents the formal condition that can be used to determine extrapolation coefficients α_v . Before coefficients can be computed, a formal connection between equation (26) and Krylov subspace $\mathcal{K}^n(\Delta \mathbf{Q}^{(n)})$ must be established.

It is a known fact that Krylov space methods for linear problems have the property to determine implicitly a minimal polynomial of the system matrix with

respect to initial residual. If the minimal polynomial of the Jacobian of the fixed-point function is denoted by $P_m(\partial_Q \mathbf{F})$, then the following identity will hold locally

$$P_m(\partial_Q \mathbf{F}) \Delta \mathbf{Q}^{(0)} = \sum_{v=0}^n c_v (\partial_Q \mathbf{F}(\mathbf{Q}^{(v)}))^v \Delta \mathbf{Q}^{(0)} = 0 \quad (27)$$

Furthermore, using equations (24) and (27)

$$\sum_{v=0}^n c_v (\partial_Q \mathbf{F}(\mathbf{Q}^{(v)}))^v \Delta \mathbf{Q}^{(0)} = (\mathbf{I} - \partial_Q \mathbf{F}(\mathbf{Q}^{(v)})) \sum_{v=0}^n c_v e^{(v)} = 0 \quad (28)$$

and since $(\mathbf{I} - \partial_Q \mathbf{F}(\mathbf{Q}^{(v)}))$ has a full rank we obtain the following expression:

$$\sum_{v=0}^n c_v e^{(v)} = 0 \quad (29)$$

From this analysis it is possible to conclude that if the coefficients c_v are chosen so that they are the coefficients of the minimal polynomial P_m with respect to $\Delta \mathbf{Q}^{(0)}$, then the condition in equation (29) holds. If we chose coefficients α_v to be the coefficients of the minimal polynomial

$$\alpha_v = c_v \quad (30)$$

then they can be determined by requiring that the L_2 norm of the approximation of the solution in Krylov space is minimized

$$\alpha = \operatorname{argmin}_{\alpha_v} \left\| \mathbf{Q}^{(v)} + \sum_{v=0}^n \alpha_v \Delta \mathbf{Q}^{(v)} \right\|_2 \quad (31)$$

The constraints for this optimization problem is obtained from the property of the minimal polynomial

$$\sum_{v=0}^n \alpha_v = 1 \quad (32)$$

Using the recursive property equation (19) and substituting it into equation (27), the following over-determined linear system results

$$\sum_{v=0}^n \alpha_v \Delta \mathbf{Q}^{(v)} = 0 \quad (33)$$

Equations (32) and (33) represent a constrained optimization problem for finding the extrapolation coefficients for the problem defined by equation (31). The resulting system of normal equations is solved using a QR-decomposition algorithm.

Once the coefficients are determined, the solution is expressed by the following extrapolation formula

$$\mathbf{Q}^{(v+1)} = \mathbf{Q}^{(v)} + \sum_{v=0}^n \alpha_v \Delta \mathbf{Q}^{(v)} \quad (34)$$

Given the fact that equation (33) represent an over-determined system of linear equations, it is obvious that not all linear equations in that system will be satisfied by the computed vector α_v . This is the consequence of the inconsistent nature of the over-determined linear system in equation (33) and this fact plays an important role in the selection of the size of the dimension of the restart space.

In practical terms, the RRE algorithm is usually implemented in a restarted version by collecting n vectors $\Delta \mathbf{Q}^{(v)}$ and storing them in the rectangular matrix $[\Delta \mathbf{Q}]$

$$[\Delta \mathbf{Q}] = [\Delta \mathbf{Q}^{(0)}, \Delta \mathbf{Q}^{(1)}, \dots, \Delta \mathbf{Q}^{(n)}] \quad (35)$$

leading to the following matrix equation

$$[\Delta \mathbf{Q}] \alpha = 0 \quad (36)$$

With previous definitions, the RRE algorithm is shown in Figure 12.1.

It is evident from the RRE pseudo-code that the iterative algorithm acts as a preconditioner for the RRE extrapolation algorithm and only the sequence of arrays $\Delta \mathbf{Q}^{(v)}$ is required in order to form the matrix $[\Delta \mathbf{Q}]$ that is used to obtain extrapolation coefficients. Therefore, the RRE algorithm is implemented as a wrapper around

Algorithm 1

for $k = 0, N$

 for $v = 0, n$

$$\Delta \mathbf{Q}^{(v)} = -\mathcal{M}^{(v)}(\mathbf{R}(\mathbf{Q}^{(v)}))$$

$$\mathbf{Q}^{(v+1)} = \mathbf{Q}^{(v)} + \Delta \mathbf{Q}^{(v)}$$

 form matrix $[\Delta \mathbf{Q}]$

end

Find $\alpha = \operatorname{argmin}_{\alpha_v} \|\mathbf{Q}^{(v)} + \sum_{v=0}^n \alpha_v \Delta \mathbf{Q}^{(v)}\|_2$ such that $\sum_{v=0}^n \alpha_v = 1$

$$\mathbf{Q}^{(k+1)} = \mathbf{Q}^{(k)} + \sum_{v=0}^n \alpha_v \Delta \mathbf{Q}^{(v)}$$

if converged

 stop

else

 restart

end if

end

Figure 12.1. RRE pseudo-code.

existing code that implements the fixed-point algorithm. A further advantage of the proposed algorithm is that no explicit knowledge of the operator $\mathcal{M}^{(v)}(\mathbf{R}(\mathbf{Q}^{(v)}))$ in equation (11) is ever required; it is only required that the fixed-point algorithm in equation (10) produce a sequence of solutions. This particular property makes this approach widely applicable and nonintrusive from the implementation point of view. Furthermore, due to equivalence between Krylov spaces, equation (21), the RRE algorithm is equivalent to the GMRES method. In the particular case of nonlinear flow solvers, the RRE algorithm corresponds to GMRES algorithm with nonlinear preconditioning and it is sometimes referred to as nonlinear GMRES.

Since the theory of RRE and its application relies on the linearized problem, most of the properties of the algorithm are valid within some small neighborhood of the operator $\mathcal{M}(\mathbf{R}(\mathbf{Q}))$. Furthermore, since we are using a restarted version of RRE, the choice of the restart space is very important. The size of the restart space is very important and somewhat problem dependent. Restarted versions of GMRES suffer from the same problems [11] and there is no theory on how to choose the size of the restart space. Therefore, some numerical experimentation is needed in order to establish the proper size of the restart space for a given class of problems.

12.4 Numerical Experiments

As was shown in the algorithm in Figure 12.1, the RRE algorithm requires only a sequence of arrays produced by the preconditioning solver $\mathcal{M}(\mathbf{R}(\mathbf{Q}^{(v)}))$. Due to this property, RRE is implemented as a wrapper around several nonlinear flow solvers including coupled pressure-based, segregated pressure-based, explicit coupled density-based and implicit coupled density-based solvers [19]. Here we describe results obtained through numerical experiments in the application of the RRE algorithm to mentioned solvers.

12.4.1 RRE acceleration of implicit density-based solver

Here we consider inviscid and viscous turbulent flow around NACA 0012 airfoil at 0° angle of attack with the free-stream Mach number $Ma = 0.7$. The computational grid is shown in Figure 12.2 and it consists of 4,800 finite volume cells. The governing equations are described by two dimensional inviscid Euler equations that can be obtained from equation (1) by removing viscous fluxes \mathbf{F}_v and dropping the momentum, continuity, and energy term in the z -direction

$$\frac{\partial}{\partial t} \int_{\Omega} \mathbf{Q} d\Omega + \oint_{\partial\Omega} \mathbf{F}_c dA = 0 \quad (37)$$

Here \mathbf{Q} and \mathbf{F}_c are given by

$$\mathbf{Q} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix}, \quad \mathbf{F}_c = \begin{pmatrix} \rho V \\ \rho uV + n_x p \\ \rho vV + n_y p \\ \rho HV \end{pmatrix}$$

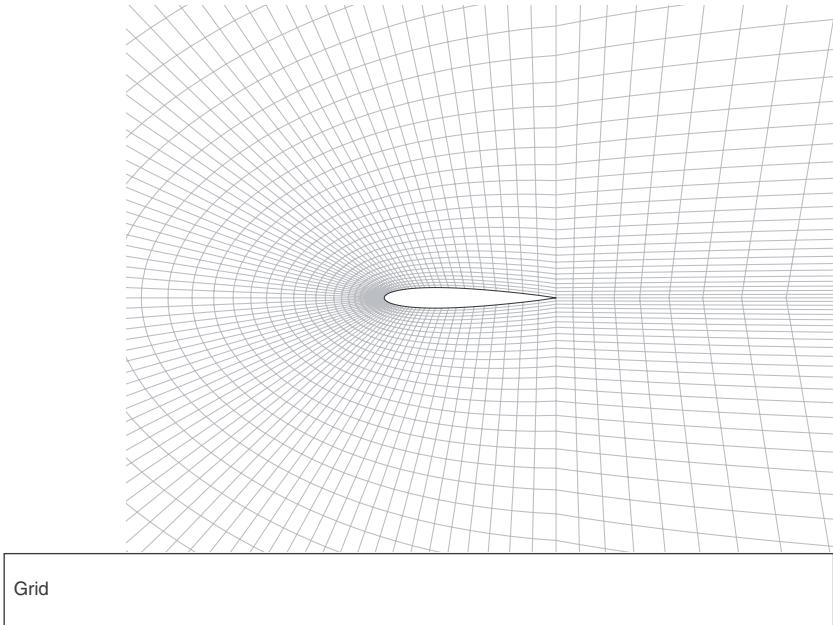


Figure 12.2. Computational grid for NACA 0012 airfoil.

The numerical flux scheme used to approximate fluxes at the centroids of faces of finite volumes is the Roe flux difference splitting scheme [1,6] with second-order spatial accuracy. Steady state was obtained after 375 implicit iterations according to the scheme in equation (8) and with CFL number equal to 25. A symmetric flow field pattern that was expected at 0° angle of attack is shown in Figure 12.3. Figure 12.4 shows the convergence history of all conserved variables. Convergence behavior is overall linear, but a small marginally unstable mode in the momentum equation is observed that corresponds to almost a periodic half-sinusoidal oscillation of the residuals. Results of the RRE algorithm that uses coupled density-based implicit solver are shown in Figure 12.5. These results are obtained by using a size of the restart space of 25 and it is evident from Figure 12.5 that the accelerated case converges after 265 implicit iterations of density-based solver. Moreover, it can be observed that an almost periodic marginally unstable mode evident in Figure 12.4 is no longer present and it is substituted with higher more irregular modes. The overall savings in terms of number of iterations between baseline and accelerated runs is about 100 iteration or 25%.

A more interesting case to consider is turbulent viscous flow over the same NACA 0012 at an angle of attack of 1.25° . For this angle of attack, a weak shock wave appears on the upper surface of the airfoil, close to the leading edge. The basic set of equations, equation (1) was extended with additional turbulence transport equations. The generic form of transport equations is given by equation (2) and in

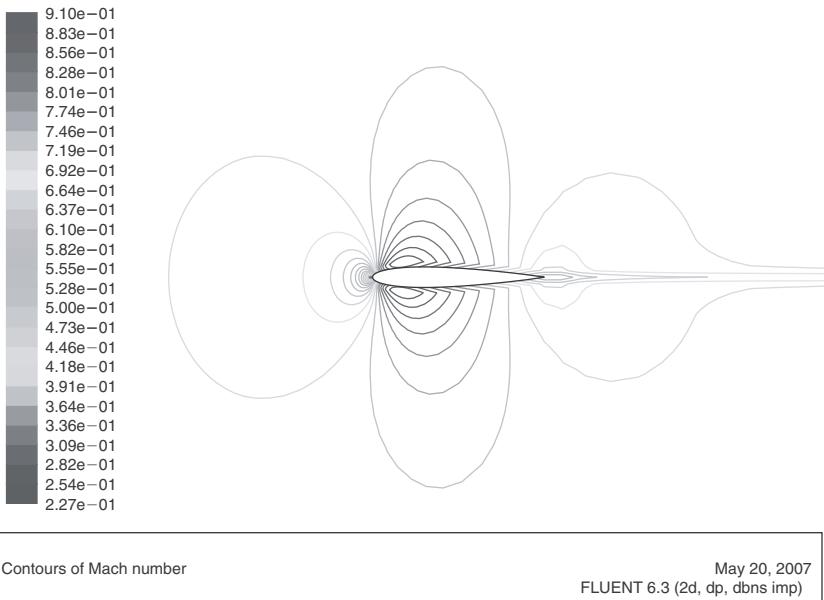


Figure 12.3. Contours of Mach number for NACA 0012 airfoil at the zero angle of attack using implicit density-based solver.

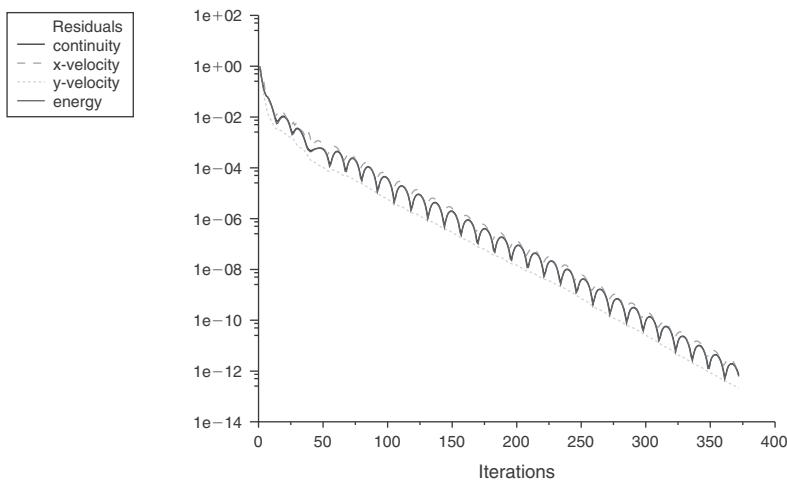


Figure 12.4. Residual history of the baseline run (no acceleration) of NACA 0012 airfoil in inviscid flow at the zero angle of attack using implicit density-based solver.

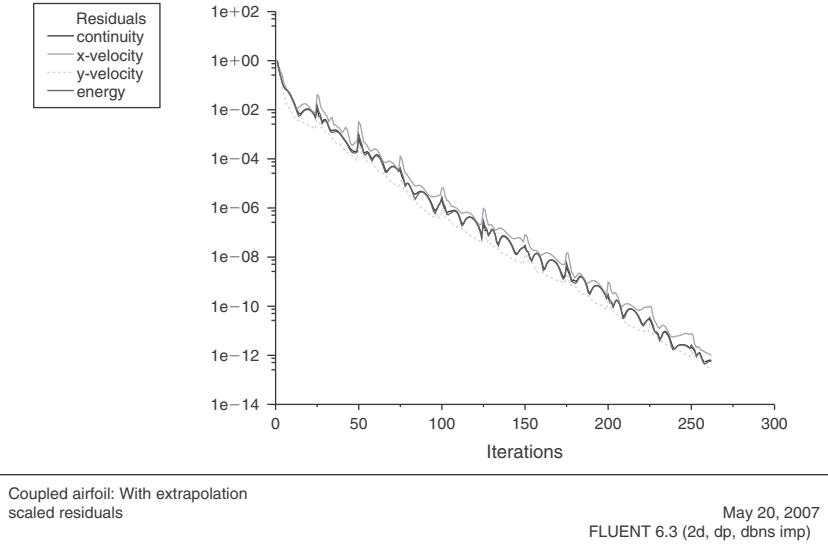


Figure 12.5. Residual history of the the accelerated run of NACA 0012 airfoil in inviscid flow at the zero angle of attack using implicit density-based solver.

the case of two equation turbulence model it takes the following form:

$$\frac{\partial}{\partial t} \int_{\Omega} \Phi d\Omega + \oint_{\partial\Omega} (\mathbf{F}_{\Phi,c} - \mathbf{F}_{\Phi,v}) dA - \int_{\Omega} \mathbf{S}_{\Phi} d\Omega = 0$$

Here Φ , $\mathbf{F}_{\Phi,c}$, and $\mathbf{F}_{\Phi,v}$ are given by

$$\Phi = \begin{pmatrix} \rho K \\ \rho \epsilon \end{pmatrix}, \quad \mathbf{F}_{\Phi,c} = \begin{pmatrix} \rho KV \\ \rho \epsilon^* V \end{pmatrix}, \quad \mathbf{F}_{\Phi,v} = \begin{pmatrix} n_x \tau_{xx}^K + n_y \tau_{yy}^K \\ n_x \tau_{xx}^\epsilon + n_y \tau_{yy}^\epsilon \end{pmatrix}$$

whereas \mathbf{S}_{Φ} is given by

$$\mathbf{S}_{\Phi} = \begin{pmatrix} P - \rho \epsilon \\ (C_{\epsilon 1} f_{\epsilon 1} - C_{\epsilon 2} f_{\epsilon 2} \rho \epsilon^*) \frac{\epsilon^*}{K} + \phi_{\epsilon} \end{pmatrix}$$

where P is production term [20] and τ_{xx} and τ_{yy} are normal turbulent viscous stresses [20]. The turbulence model that was used in the computation is realizable $k - \epsilon$ model with the free stream turbulence conditions that correspond to 0.5% turbulence intensity with a viscous ratio of turbulent to laminar viscosity of 10. These conditions correspond to free stream conditions in the calm atmosphere. Results of the computations using baseline coupled implicit density-based solver are shown in Figure 12.6. Convergence behavior for this case is shown in Figure 12.7 where it can be seen that the convergence was achieved after 390 nonlinear iterations. It is also

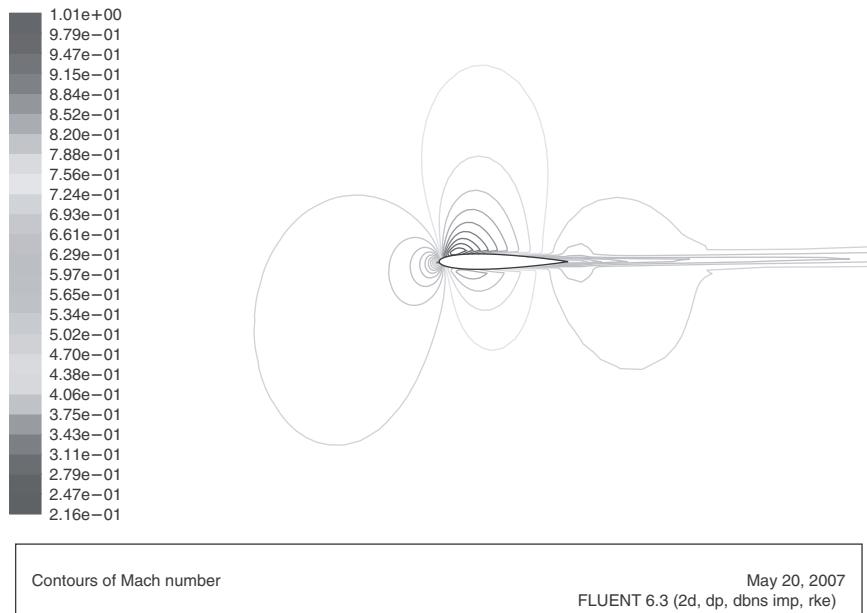


Figure 12.6. Mach contours of NACA 0012 airfoil in viscous turbulent flow at 1.25° angle of attack using implicit density-based solver.

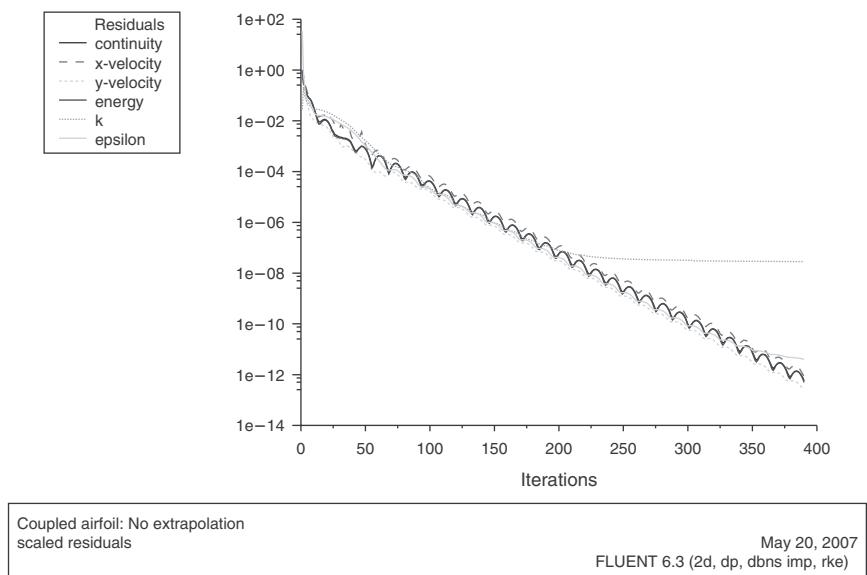


Figure 12.7. Residual history of the baseline (no acceleration) run of NACA 0012 in viscous turbulent flow at 1.25° angle of attack using implicit density-based solver.

observed that turbulent dissipation rate experiences a plateau effect after 200 nonlinear iterations, keeping the level of residuals at $1E-07$ without any further decrease. For practical computations, this level of residuals may be acceptable but it clearly indicates a difficulty with the convergence of k -equation. Application of the RRE accelerating algorithm produces better convergence behavior since the problem converges in 275 nonlinear iterations without the plateau effect in k -equation.

12.4.2 RRE acceleration of explicit density-based solver

Here again we consider inviscid and viscous turbulent flow around NACA 0012 airfoil at 0° and 1.25° angles of attack with the free-stream Mach number $Ma = 0.7$. However, this time we use the explicit multistage Runge–Kutta scheme with the full approximation storage scheme [1, 18, 19]. The same set of equations, equation (37) and the nonlinear update is performed by the explicit scheme in equation (7) with modifications to accommodate FAS and Runge–Kutta multistage algorithms. Due to stability restrictions, CFL number of 2 is used with second-order scheme in space (Figure 12.8).

The first case that was considered is the inviscid flow over NACA 0012 at the zero angle of attack with free-stream Mach number $Ma = 0.7$. Convergence behavior of residuals of the conserved variables are shown in Figure 12.9 where it can be seen that it takes approximately 3,470 nonlinear iterations to reduce residuals below tolerance level set to be $1E - 12$.

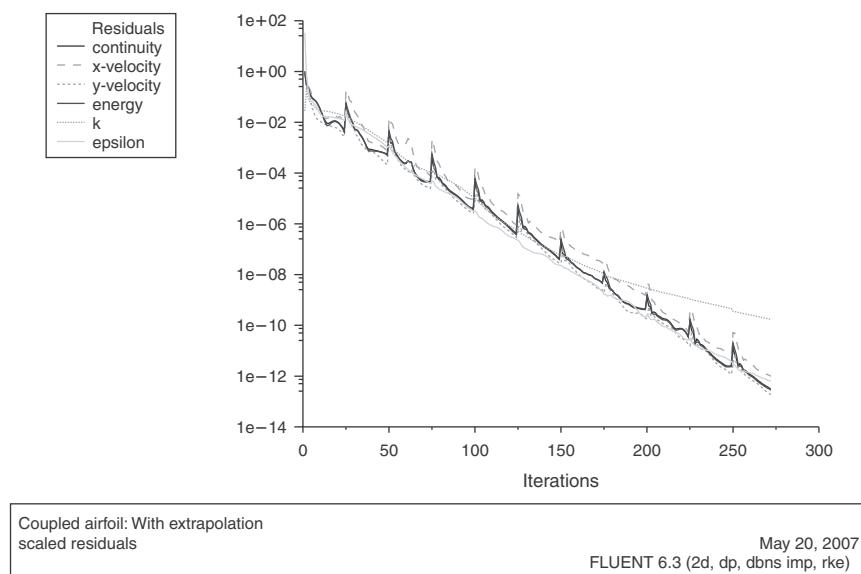


Figure 12.8. Residual history of the accelerated run of NACA 0012 in viscous turbulent flow at 1.25° angle of attack using implicit density-based solver.

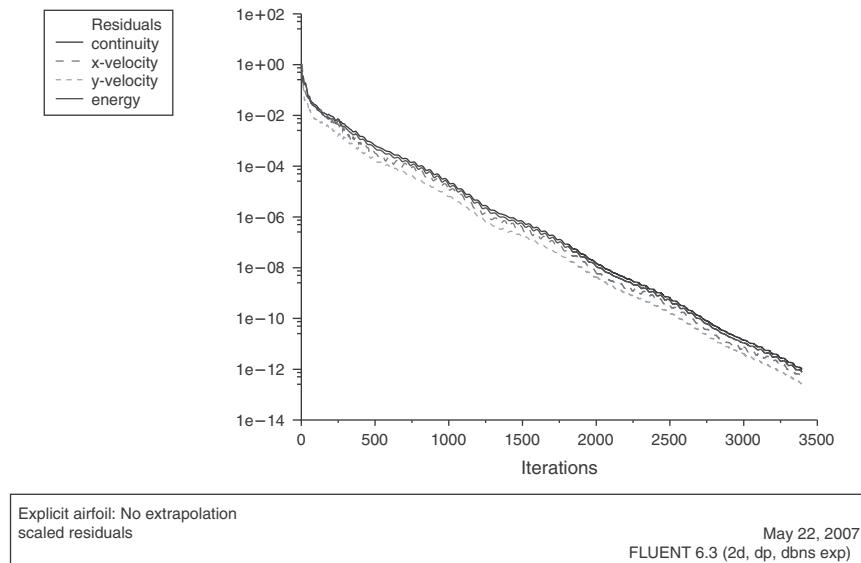


Figure 12.9. Residual history of the baseline (no acceleration) run of NACA 0012 in inviscid flow at the zero angle of attack using explicit density-based solver.

In the case of the RRE accelerated explicit density-based algorithm, the size of a restart space is 25 as in the case of implicit solvers. However, due to the fact that the explicit solvers are not very good preconditioners, the resulting Krylov subspace does not have a good basis to guarantee convergence of the RRE method. It was determined experimentally that if every 10th correction vector is stored, then RRE converges. This behavior is induced by a very nonlinear behavior of the system on a scale of 10 to 15 iterations as it can be seen in Figure 12.9. Skipping iterations results in a better basis of the Krylov subspace and the RRE method of accelerating explicit density-based solver converges as shown in Figure 12.10 where it can be seen that the convergence is achieved after approximately 2,250 nonlinear iterations. This behavior requires further research and some possible approaches include evaluation of residuals to determine the good candidate for the extrapolation [14]. Some other measures of the quality of the Krylov subspace that involve singular value decomposition of matrix $[\Delta \mathbf{Q}]$ and thresholding based on the desired accuracy can be used.

A viscous turbulent case of NACA 0012 at an angle of attack of 1.25° with the same free-stream conditions is examined next. Stalled convergence as shown in Figure 12.11 is experienced in this case with the residual levels in the vicinity of $1E - 06$. Employing the same strategy of storing every 10th correction vector, convergence is accelerated and stalled residuals are removed. Convergence improvement in this case is attributed to improved coupling between flow and turbulence equations by the RRE algorithm. Current algorithm in density-based solver

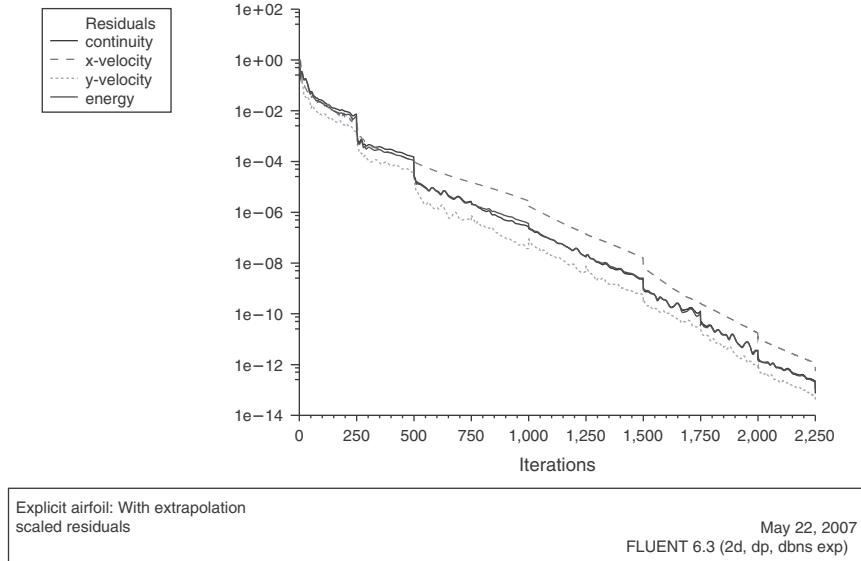


Figure 12.10. Residual history of the accelerated run of NACA 0012 in inviscid flow at an 1.25° angle of attack using explicit density-based solver.

solves flow equations in a coupled manner and then it solves turbulence equations separately by using computed velocity field. This procedure sometimes leads into convergence problems that are observed in Figure 12.11 and the RRE algorithm helps the convergence by providing the missing coupling between equations by computing one set of extrapolation coefficients used in extrapolation of all fields. Furthermore, in a view of previous connection between the RRE and GMRES method, this approach corresponds to applying the GMRES method on a coupled system of equations thus improving the convergence (Figure 12.12).

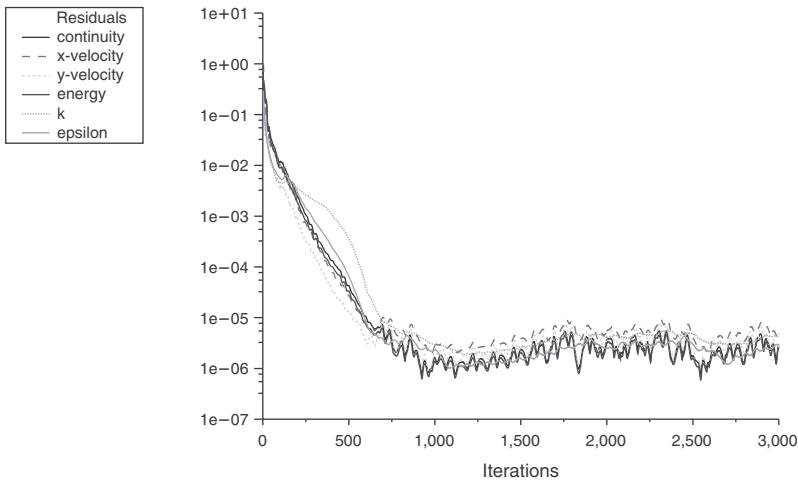
12.4.3 RRE acceleration of segregated pressure-based solver

In this section we consider acceleration of the computation of laminar incompressible flows by the RRE method. The basic set of equations is given in equation (1) and their specialization for the laminar incompressible flow is given by the following equation:

$$\oint_{\partial\Omega} (\mathbf{F}_c - \mathbf{F}_v) dA = 0 \quad (38)$$

with the incompressibility constraint enforced through the divergence-free condition

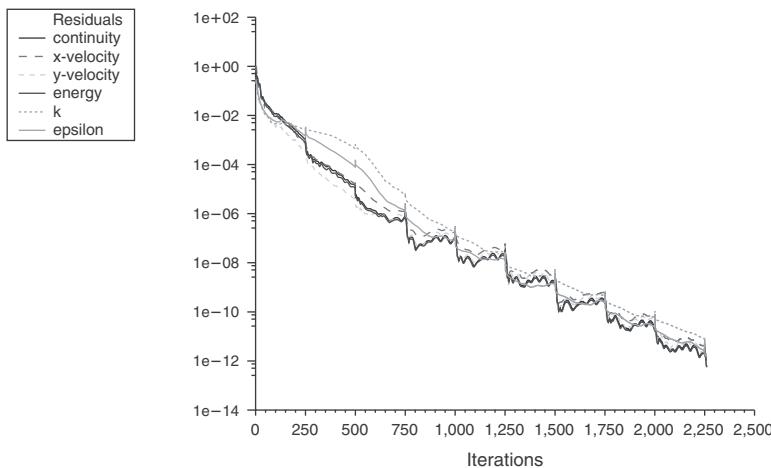
$$\int_{\Omega} \nabla \cdot (\rho u) d\Omega = 0 \quad (39)$$



Explicit airfoil: No extrapolation
scaled residuals

May 23, 2007
FLUENT 6.3 (2d, dp, dbns exp, ske)

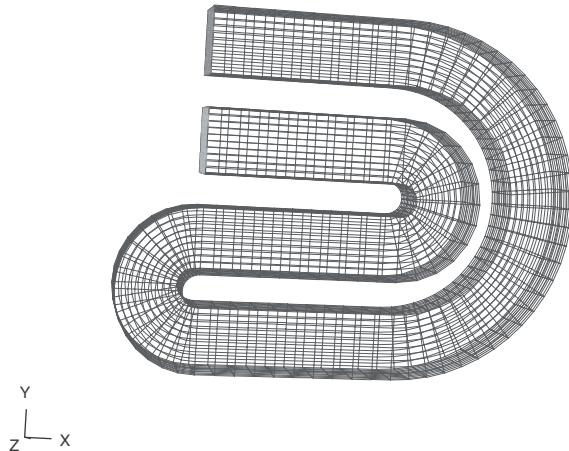
Figure 12.11. Residual history of the baseline (no acceleration) run of NACA 0012 in a viscous turbulent flow at an 1.25° angle of attack using explicit density-based solver.



Explicit airfoil: With extrapolation
scaled residuals

May 23, 2007
FLUENT 6.3 (2d, dp, dbns exp, ske)

Figure 12.12. Residual history of the accelerated run of NACA 0012 in a viscous turbulent flow at an 1.25° angle of attack using explicit density-based solver.



Grid

Feb 04, 2007
FLUENT 6.3 (3d, dp, pbn, lam)

Figure 12.13. Computational grid for laminar viscous flow in serpentine geometry.

Variables in equation (1) for the case of incompressible laminar flows are given by following expressions:

$$\mathbf{F}_c = \begin{pmatrix} \rho u V + n_x p \\ \rho v V + n_y p \\ \rho w V + n_z p \end{pmatrix}, \quad \mathbf{F}_v = \begin{pmatrix} n_x \tau_{xx} + n_y \tau_{xy} + n_z \tau_{xz} \\ n_x \tau_{yx} + n_y \tau_{yy} + n_z \tau_{yz} \\ n_x \tau_{zx} + n_y \tau_{zy} + n_z \tau_{zz} \end{pmatrix}$$

The incompressible system of equations (38) and (39) is usually solved through the projection methods [1,3,4] where the primary unknown variables are the primitive variables such as pressure and velocity. Furthermore, due to the kinematic constraint, equation (39) imposed on the velocity field, there is no direct link between pressure and velocity fields, a special pressure equation is usually formed [1,4]. Forming the pressure equation, together with the momentum equation and the boundary conditions, creates a well-posed problem that can be solved in a coupled or segregated manner. If the SIMPLE algorithm [4] is adopted, segregated algorithm results.

As a study case, we consider the flow through the geometry shown in Figure 12.13. Due to the nonlinearities in equation (38), and due to the initial guesses that lay out of the convergence domain of the SIMPLE methods, relaxation factors are used in the discrete system of equations to maintain the stability of the computations. Standard relaxation factors (0.3 pressure and 0.7 for momentum equation) together with the second-order formulations for pressure and velocity variables is used in computation. The problem in Figure 12.13 is described by one

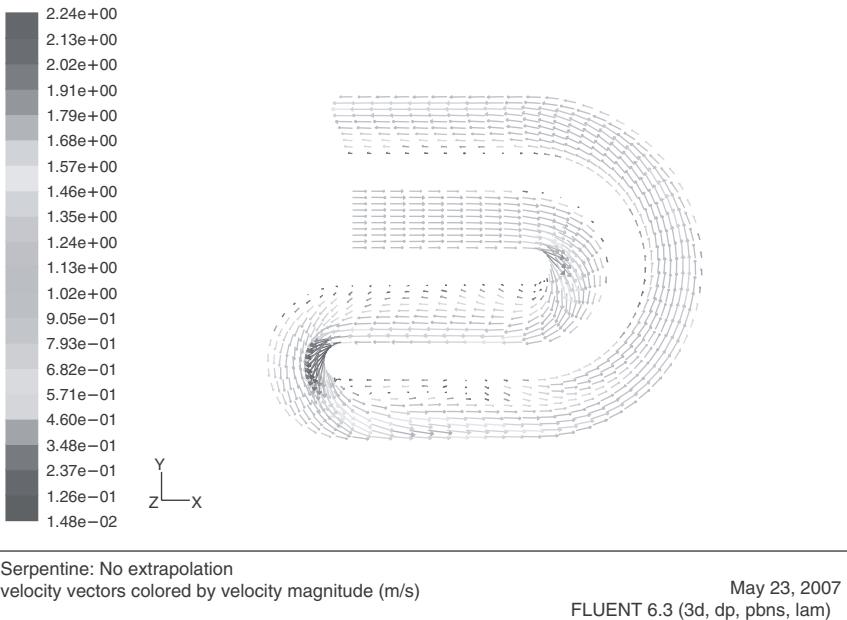
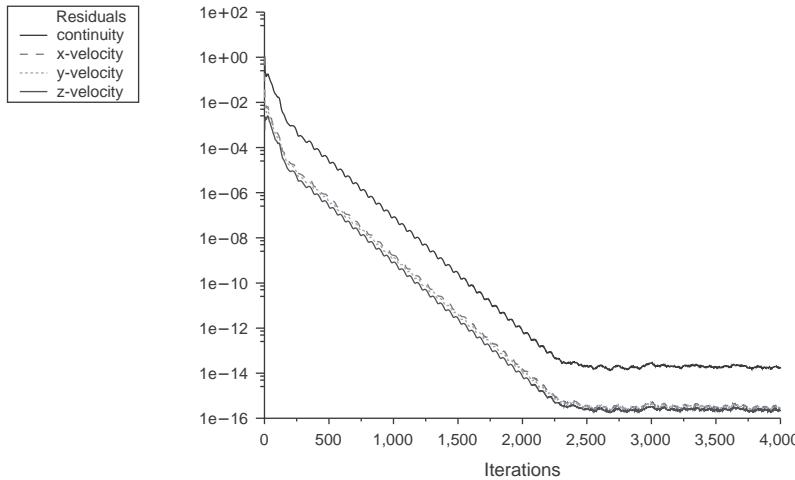


Figure 12.14. Vector field in incompressible flow plotted in the middle plane of the serpentine geometry.

inflow boundary condition prescribed as a velocity inlet, one outlet boundary condition prescribed as an uniform pressure outlet, and no-slip walls. Problem was discretized using hexagonal 8,900 finite volumes. System of discretized equations for the pressure and velocity variables was solved using implicit formulation that forms four systems of linear equations that are solved using the AMG method [18,19] in a segregated manner and velocity vectors on the middle plane are shown in Figure 12.14. Convergence criterion was set for all equations to be $1E - 14$ for the norm of residuals and the results of the computations are shown in Figure 12.15 where it can be seen that it takes close to 2,500 iterations to reach the prescribed level of convergence. The accelerated case is shown in Figure 12.16 where the RRE algorithm results in approximately 1,000 iterations required to reach the prescribed levels of residuals. As in the case of explicit density-based solver, the size of a restart space is 25 and RRE stored every 10th vector in order to obtain useful basis of Krylov subspace.

12.4.4 RRE acceleration of coupled pressure-based solver

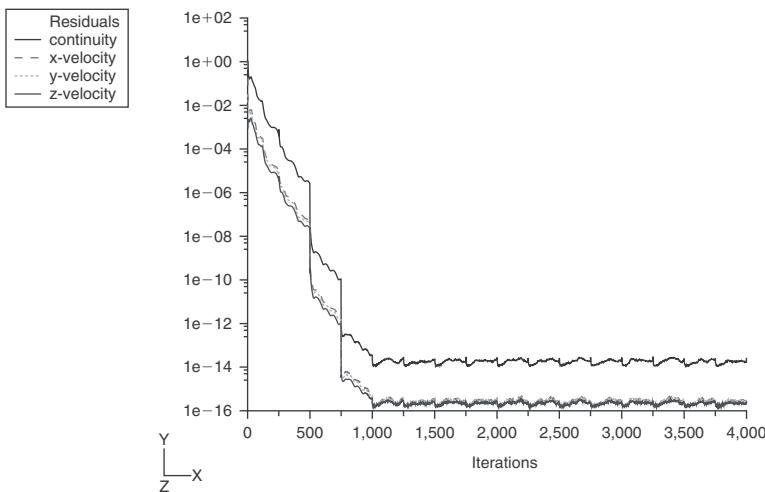
The set of governing equations (38) and (39) for the incompressible flow can be discretized and solved simultaneously resulting in a so-called coupled pressure-based solver. In contrast to the segregated solver, matrix coefficients obtained through coupled discretization result in a point coupled system where three velocities and



Serpentine: No extrapolation
scaled residuals

May 23, 2007
FLUENT 6.3 (3d, dp, pbns, lam)

Figure 12.15. Residual history of the baseline run (no acceleration) of the incompressible laminar flow in the serpentine geometry using segregated pressure-based solver.



Serpentine: With extrapolation
scaled residuals

May 23, 2007
FLUENT 6.3 (3d, dp, pbns, lam)

Figure 12.16. Residual history of the accelerated run of the incompressible laminar flow in the serpentine geometry using segregated pressure-based solver.

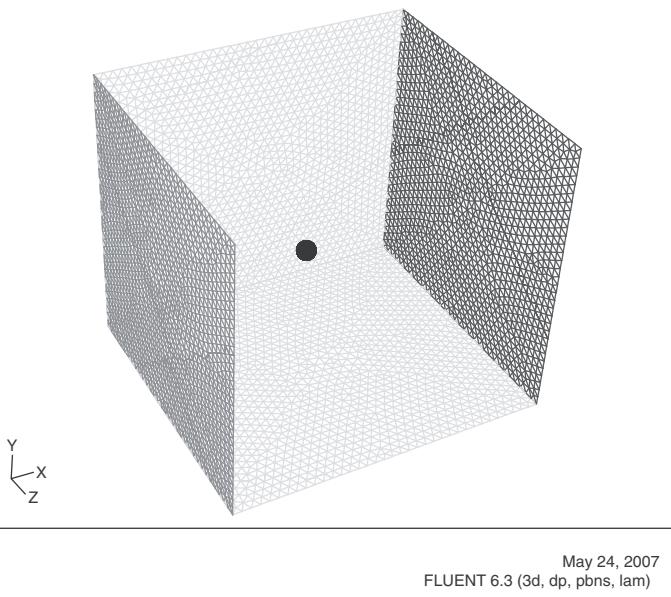


Figure 12.17. Computational grid in the middle plane of a sphere in a free stream flow at $\text{Re} = 250$.

one pressure is stored in a block coefficient for each finite volume cell [19]. This system of equations is solved using again the AMG method with the modifications for block coefficients.

The RRE method of acceleration was applied to the case of sphere in a free stream flow conditions with Reynolds number equal to 250. The grid consists of $1.2E + 06$ finite volume cells of a tetrahedral shape as shown in Figure 12.17. Stability of the computational run was controlled by the CFL number which was chosen to be 200 in this case. The flow is laminar and the contours of the pressure and vectors of the velocity field are shown in Figures 12.18 and 12.19, respectively. The convergence history of baseline run without acceleration is shown in Figure 12.20 where the slow convergence is observed. Convergence behavior for the accelerated case is shown in Figure 12.21 where faster convergence is observed. As in the case of density-based implicit solver, restart base is selected to be 25 and correction vectors are collected after each nonlinear iteration.

12.5 Conclusion

We have demonstrated the application of the RRE method as an acceleration method for the nonlinear solvers in CFD. The RRE method relies on the recursive property of the fixed-point algorithms used in CFD to form the basis of the Krylov subspace that contains the solution to the problem. Due to the nonlinearity of the fixed-point

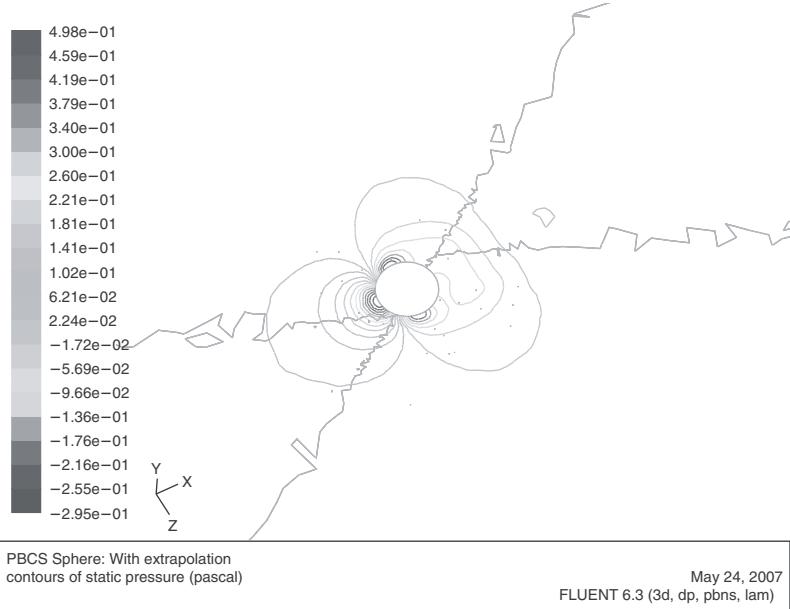


Figure 12.18. Pressure field in the middle plane of a sphere in a free stream flow at $Re = 250$ using coupled pressure-based solver.

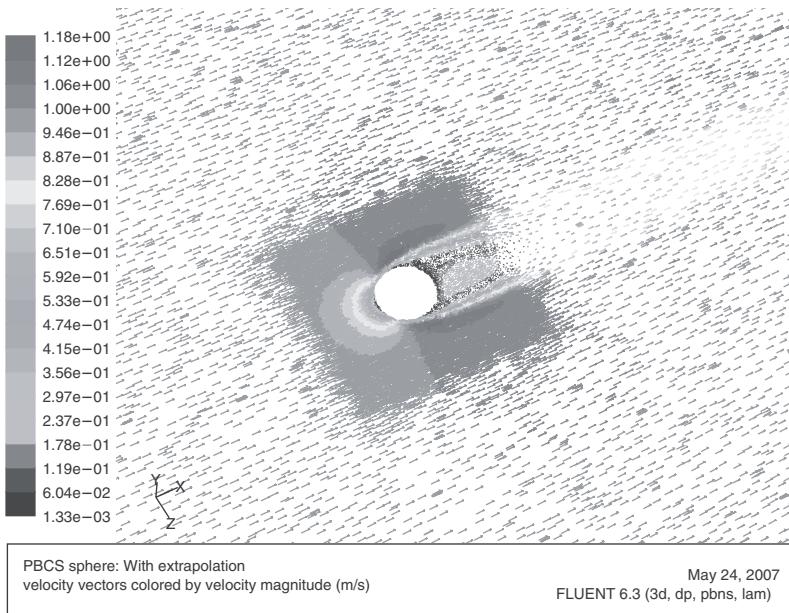


Figure 12.19. Velocity vectors in the middle plane of a sphere in a free stream flow at $Re = 250$ using coupled pressure-based solver.

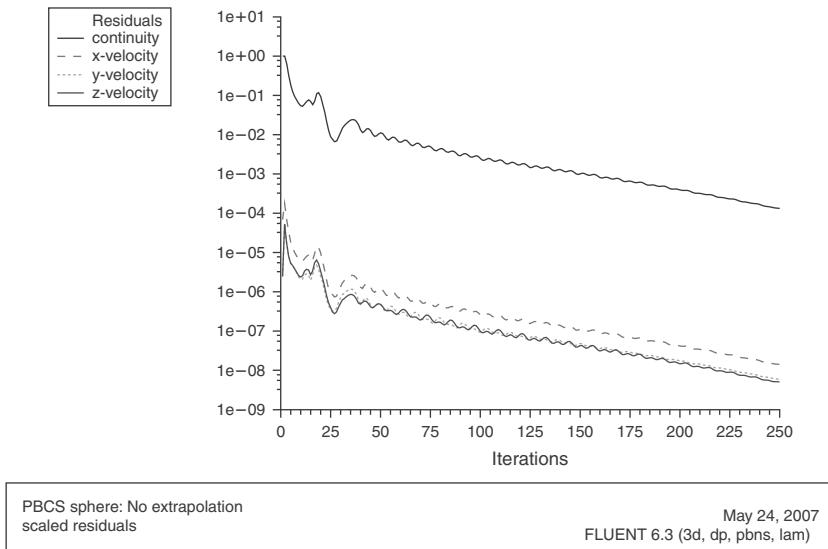


Figure 12.20. Residual history of the baseline (no acceleration) run of a sphere in a free stream flow at $Re = 250$ using coupled pressure-based solver.

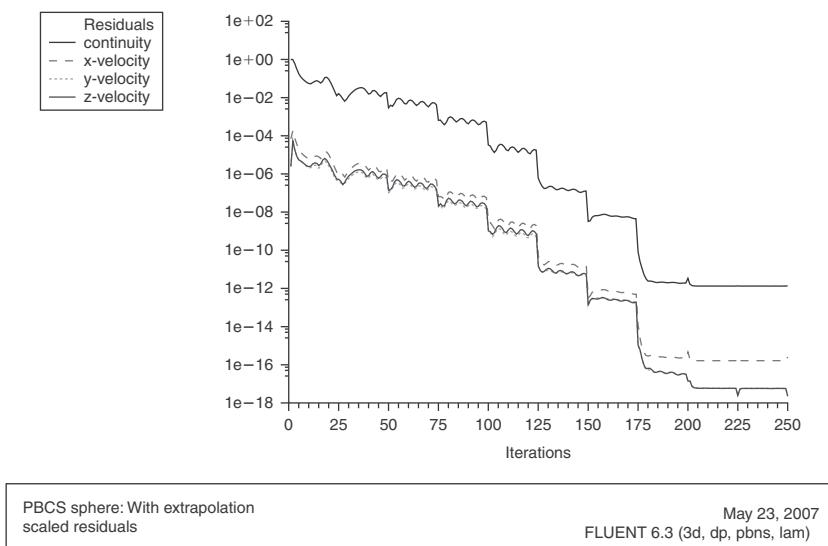


Figure 12.21. Residual history of the accelerated run of a sphere in a free stream flow at $Re = 250$ using coupled pressure-based solver.

function of CFD algorithms, restarted algorithm is used to make the algorithm memory efficient and to rely on local linear properties of the restarted Krylov subspace to represent the solution correctly. The proposed algorithm is shown to be nonintrusive and applicable to a wide range of solvers including explicit and implicit formulations of pressure- and density-based algorithms. The RRE algorithm requires only the storage of correction vectors and does not require evaluation of the residuals which makes this algorithm easy to implement for the solvers that do not form residual vector. Improved coupling of the systems of equations when used with the RRE algorithm is demonstrated and shown to improve convergence properties of segregated solvers.

References

- [1] Wesseling, P. *Principles of Computational Fluid Dynamics*, Springer-Verlag, New York, 2000.
- [2] Patankar, S. V. *Numerical Heat Transfer and Fluid Flow*, McGraw-Hill, New York, 1980.
- [3] Chorin, A. J. A Numerical Method for Solving Incompressible Viscous Flow Problems, *Journal of Computational Physics*, 2(1), pp. 12–26, 1967.
- [4] Ferziger, J. H., and Perić, M. *Computational Methods for Fluid Dynamics*, Springer, Berlin, 1996.
- [5] Hirsch, C. *Numerical Computation of Internal and External Flows*, 2, John Wiley, Chichester, 1990.
- [6] Laney, C. B. *Computational Gasdynamics*, Cambridge University Press, Cambridge, 1998.
- [7] Allgower, E. L., and Georg, K. *Introduction to Numerical Continuation Methods*, SIAM, Philadelphia, 1987.
- [8] Deuflhard, P. *Newton Methods for Nonlinear Problems*, Springer-Verlag, Berlin, 2004.
- [9] Wilcox, D. C. *Turbulence Modeling for CFD*, DCW Industries, 2006.
- [10] Brezinski, C., and Zaglia, M. R. *Extrapolation Methods Theory and Practice*, Elsevier Science Publishing, Amsterdam, 1991.
- [11] Saad, Y. *Iterative Methods for Sparse Linear Systems* SIAM, Philadelphia, 2004.
- [12] Jameson, A., Schmidt, W., and Turkel, E. Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping, *AIAA Paper 81-1259*, 1981.
- [13] Sidi, A., and Celestina, M. L. Convergence Acceleration for Vector Sequences and Application to Computational Fluid Dynamics, *NASA Technical Memolen*, 101327 (ICOMP-88-17).
- [14] Washio, T., and Oosterlee, C. W. Krylov Subspace Acceleration for Nonlinear Multigrid Schemes, *Electronic Transactions on Numerical Analysis*, 6, pp. 271–290, December 1997, ISSN 1068-9613. Institute for Computational Mechanics in Propulsion, Lewis Research Center, Cleveland, OH.
- [15] Jaschke, L. Preconditioned Arnoldi Methods for Systems of Nonlinear Equations, PhD Thesis, *Swiss Federal Institute of Technology*, Zurich, Switzerland, 2003.
- [16] Andersson, C. Solving Linear Equations on Parallel Distributed Memory Architectures by Extrapolation, Technical Report, TRITA-NA-E9746, Department of Numerical Analysis and Computer Science, Royal Institute of Technology, Sweden, 1997.

- [17] Jemcov, A., Maruszewski, J. P., and Jasak, H. Performance Improvement of Algebraic Multigrid Solver by Vector Sequence Extrpolation, *15th Annual Conference of CFD Society of Canada*, May 27–31, Toronto, 2007.
- [18] Briggs, W., Henson, V. E., and McCormick, S. A Multigrid Tutorial, *SIAM*, Philadelphia, 2000.
- [19] Fluent Inc., Fluent Users Manual, Fluent Inc., 2006.
- [20] Pope, S. B. *Turbulent Flows*, Cambridge University Press, Cambridge, 2000.

This page intentionally left blank



WITPRESS ...for scientists by scientists

Computational Methods in Multiphase Flow V

Edited by: A. MAMMOLI, The University of New Mexico, USA and C.A. BREBBIA, Wessex Institute of Technology, UK

Together with turbulence, multiphase flow remains one of the most challenging areas of computational mechanics and experimental methods. Numerous problems remain unsolved to date. Multiphase flows are found in all areas of technology, at all length scales and flow regimes. The fluids involved can be compressible or incompressible, linear or nonlinear.

Because of the complexity of the problems, it is often essential to utilize advanced computational and experimental methods to solve the equations that describe them. Challenges with these simulations include modelling and tracking interfaces, dealing with multiple length scales, modelling nonlinear fluids, treating drop breakup and coalescence, characterizing phase structures, and many others. Experimental techniques, although expensive and difficult to perform, are essential to validate models.

This volume includes papers from the fifth international conference on the subject. Featured topics include: Multiphase Flow Simulation; Interaction of Gas, Liquids and Solids; Turbulent Flow; Environmental Multiphase Flow; Bubble and Drop Dynamics; Flow in Porous Media; Heat Transfer; Image Processing; Interfacial Behaviour.

WIT Transactions on Engineering Sciences, Vol 63
ISBN:978-1-84564-188-7 eISBN: 978-1-84564-365-2
2009 544pp £198.00

*All prices correct at time of going to press but
subject to change.*

*WIT Press books are available through your
bookseller or direct from the publisher.*



WITPRESS ...for scientists by scientists

Advances in Fluid Mechanics VIII

Edited by: M. RAHMAN, Dalhousie University, Canada and C.A. BREBBIA, Wessex Institute of Technology, UK

This book discusses the basic formulations of fluid mechanics and their computer modelling, as well as the relationship between experimental and analytical results. Containing papers from the Eighth International Conference on Advances in Fluid Mechanics, the book will be a seminal text to scientists, engineers and other professionals interested in the latest developments in theoretical and computational fluid mechanics.

The conference will cover a wide range of topics, with emphasis on new applications and research currently in progress, which include: Computational Methods in Fluid Mechanics; Environmental Fluid Mechanics; Experimental Versus Simulation Methods; Multiphase Flow; Hydraulics and Hydrodynamics; Heat and Mass Transfer; Industrial Applications; Wave Studies; Biofluids; Fluid Structure Interaction.

WIT Transactions on Engineering Sciences, Vol 69

ISBN: 978-1-84564-476-5 eISBN: 978-1-84564-477-2
2010 608pp £231.00

Transport Phenomena in Fires

Edited by: M. FAGHRI, University of Rhode Island, USA and B. SUNDÉN, Lund University, Sweden

Controlled fires are beneficial for the generation of heat and power while uncontrolled fires, like fire incidents and wildfires, are detrimental and can cause enormous material damage and human suffering. This edited book presents the state of the art of modeling and numerical simulation of the important transport phenomena in fires. It describes how computational procedures can be used in analysis and design of fire protection and fire safety. Computational fluid dynamics, turbulence modeling, combustion, soot formation, and thermal radiation modeling are demonstrated and applied to pool fires, flame spread, wildfires, fires in buildings and other examples.

Series: Developments in Heat Transfer, Vol 20

ISBN: 978-1-84564-160-3 eISBN: 978-1-84564-304-1
2008 496pp £163.00



WITPRESS ...for scientists by scientists

Advanced Computational Methods in Heat Transfer XI

Edited by: B. SUNDÉN, Lund University, Sweden, Ü. MANDER, University of Tartu, Estonia and C.A. BREBBIA, Wessex Institute of Technology, UK

Research and development of computational methods for solving and understanding heat transfer problems continue to be important because heat transfer topics are commonly of a complex nature and different mechanisms such as heat conduction, convection, turbulence, thermal radiation and phase change may occur simultaneously. Typically, applications are found in heat exchangers, gas turbine cooling, turbulent combustions and fires, electronics cooling, melting and solidification. Heat transfer has played a major role in new application fields such as sustainable development and the reduction of greenhouse gases as well as for micro- and nano-scale structures and bio-engineering.

In engineering design and development, reliable and accurate computational methods are required to replace or complement expensive and time-consuming experimental trial and error work. Tremendous advancements have been achieved during recent years due to improved numerical solutions of non-linear partial differential equations and computer developments to achieve efficient and rapid calculations. Nevertheless, to further progress, computational methods will require developments in theoretical and predictive procedures – both basic and innovative – and in applied research. Accurate experimental investigations are needed to validate the numerical calculations.

This book contains papers originally presented at the Eleventh International Conference, arranged into the following topic areas: Natural and Forced Convection and Radiation; Heat Exchanges; Advances in Computational Methods; Heat Recovery; Heat Transfer; Modelling and Experiments; Renewable Energy Systems; Advanced Thermal Materials; Heat Transfer in Porous Media; Multiphase Flow Heat Transfer.

WIT Transactions on Engineering Sciences, Vol 68

ISBN: 978-1-84564-462-8 eISBN: 978-1-84564-463-5

2010 352pp £134.00



WITPRESS ...for scientists by scientists

Transport Phenomena in Fuel Cells

Edited by: B. SUNDÉN, Lund University, Sweden and M. FAGHRI, University of Rhode Island, USA

Fuel cells are expected to play a significant role in the next generation of energy systems and road vehicles for transportation. However, before this can happen, substantial progress is required in reducing manufacturing costs and improving performance. Many of the heat and mass transport processes associated with fuel cells are not well understood and, depending on the fuel being used, modifications in the design of the next generation of cells are needed.

This is the first book to provide a comprehensive analysis of transport phenomena in fuel cells, covering fundamental aspects of their function, operation and practical consequences. It will contribute to the understanding of such processes in Solid Oxide Fuel Cells (SOFC), Proton Exchange Membrane Fuel Cells (PEMFC) and Direct Methanol Fuel Cells (DMFC). Written by eminent scientists and research engineers in the field, individual chapters focus on various mathematical models and simulations of transport phenomena in multiphase flows, including dominant processes such as heat and mass transport and chemical reactions. Relevant experimental data is also featured.

A detailed summary of state-of-the-art knowledge and future needs, the text will be of interest to graduate students and researchers working on the development of fuel cells within academia and industry.

Series: Developments in Heat Transfer, Vol 19

ISBN: 1-85312-840-6

2005 384pp £129.00

WIT eLibrary

Home of the Transactions of the Wessex Institute, the WIT electronic-library provides the international scientific community with immediate and permanent access to individual papers presented at WIT conferences. Visitors to the WIT eLibrary can freely browse and search abstracts of all papers in the collection before progressing to download their full text.

Visit the WIT eLibrary at
<http://library.witpress.com>