A NEW ALGORITHM FOR TURBULENT FLOW VISUALIZATION IN A RECONFIGURABLE ADVANCED VISUALIZATION ENVIRONMENT

Supreeya Boontham Miller
University of Southern Mississippi

Follow this and additional works at: https://aquila.usm.edu/dissertations

Part of the Computer Engineering Commons

Recommended Citation
Miller, Supreeya Boontham, "A NEW ALGORITHM FOR TURBULENT FLOW VISUALIZATION IN A RECONFIGURABLE ADVANCED VISUALIZATION ENVIRONMENT" (2007). Dissertations. 1249.
https://aquila.usm.edu/dissertations/1249

This Dissertation is brought to you for free and open access by The Aquila Digital Community. It has been accepted for inclusion in Dissertations by an authorized administrator of The Aquila Digital Community. For more information, please contact Joshua.Cromwell@usm.edu.
A NEW ALGORITHM FOR TURBULENT FLOW VISUALIZATION IN
A RECONFIGURABLE ADVANCED VISUALIZATION ENVIRONMENT

by

Supreeya Boontham Miller

A Dissertation
Submitted to the Graduate Studies Office
of The University of Southern Mississippi
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy

Approved:

May 2007
A NEW ALGORITHM FOR TURBULENT FLOW VISUALIZATION IN
A RECONFIGURABLE ADVANCED VISUALIZATION ENVIRONMENT

by

Supreeya Boontham Miller

Abstract of a Dissertation
Submitted to the Graduate Studies Office
of The University of Southern Mississippi
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy

May 2007
ABSTRACT

A NEW ALGORITHM FOR TURBULENT FLOW VISUALIZATION IN
A RECONFIGURABLE ADVANCED VISUALIZATION ENVIRONMENT

by
Supreeya Boontham Miller

May 2007

This dissertation begins with a brief background of some of the different visualization
techniques and approaches to modeling sediment transport in turbulent flow vector fields.
A new physical model for sediment transport is derived as well as new 3D immersive
visualization software. Our new physical model and visualization system provides a novel
scientist-interactive environment for studying problems of interest in the area of sediment
transport within turbulent flows. Three sample problems are studied using the new system
with graphical results and explanations. The system is designed to study turbulent flow
problems as a function of time and boundary conditions. Domain tessellations are used
to incorporate a boundary collision detection algorithm in order to effect particle/bottom
surface interactions. The results can be used for model validation, forecasting, and for
supporting new scientific conclusions concerning particular problems. The immersive 3D
software provides scientists with a new tool for studying model results in a Reconfigurable
Advanced Visualization Environment.
I would like to thank my major professor, Dr. Shahrdad Sajjadi, for his patience in directing me through this research project. I greatly appreciate the administrative efforts Dr. Ray Seyfarth made in helping me keep graduate school and departmental requirements satisfied. I would like to thank Dr. Jim Miller for his efforts in editing my dissertation. I thank the rest of my committee, Dr. Louise Perkins and Dr. Adel Ali, for their unswerving support and encouragement. Also, I extend great thanks to my supervisors and colleagues at the Center of Higher Learning, Mr. Joe Swaykos, Mr. Conrad Johnson, and Mr. Gary Morris, for their encouragement and support throughout my research. I extend a special thanks to fellow visualization researcher, Dr. Robert S. Laramee, for allowing me to reference some of his work. Finally, I would like to thank my whole family for their encouragement, support, and sacrifice throughout this process. I especially thank my husband, Dr. MJ Miller, for supporting me as both a scientist and spouse. I especially thank my daughter, Annette Miller, for her sacrifice and understanding. Finally, I extend a special thanks to my parents, Manus and Kularb Boontham, who have supported and provided so many things to me throughout my life, least of all their love.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>SECTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>ii</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>iii</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>vi</td>
</tr>
<tr>
<td>LIST OF ILLUSTRATIONS</td>
<td>vii</td>
</tr>
<tr>
<td>LIST OF ABBREVIATIONS</td>
<td>xii</td>
</tr>
<tr>
<td>1 INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>2 SCIENTIFIC VISUALIZATION</td>
<td>6</td>
</tr>
<tr>
<td>2.1 Flow Visualization</td>
<td>9</td>
</tr>
<tr>
<td>3 COMPUTATIONAL FLUID DYNAMICS</td>
<td>16</td>
</tr>
<tr>
<td>3.1 Introduction</td>
<td>16</td>
</tr>
<tr>
<td>3.2 Finite-Volume Discretization</td>
<td>19</td>
</tr>
<tr>
<td>3.3 Convection Schemes</td>
<td>24</td>
</tr>
<tr>
<td>3.4 Pressure Correction Algorithm</td>
<td>29</td>
</tr>
<tr>
<td>3.5 Second-Moment Closure</td>
<td>38</td>
</tr>
<tr>
<td>3.6 Numerical Aspects of Reynolds-Stress Modeling</td>
<td>45</td>
</tr>
<tr>
<td>3.7 Wall-Reflection Terms</td>
<td>56</td>
</tr>
<tr>
<td>3.8 Boundary Conditions</td>
<td>60</td>
</tr>
<tr>
<td>4 NUMERICAL GRID GENERATION FOR CURVILINEAR GEOMETRIES</td>
<td>75</td>
</tr>
<tr>
<td>4.1 Theoretical background</td>
<td>75</td>
</tr>
<tr>
<td>5 IMMERSIVE VISUALIZATION</td>
<td>90</td>
</tr>
<tr>
<td>5.1 Immersive Visualization Technical Difficulties</td>
<td>98</td>
</tr>
<tr>
<td>6 IMMERSIVE VISUALIZATION RESEARCH ENVIRONMENT</td>
<td>101</td>
</tr>
<tr>
<td>6.1 Our System</td>
<td>102</td>
</tr>
<tr>
<td>7 APPLICATION VISUALIZATION TO TURBULENT FLOWS</td>
<td>105</td>
</tr>
<tr>
<td>7.1 Introduction</td>
<td>105</td>
</tr>
<tr>
<td>7.2 Oscillatory Turbulent Flow</td>
<td>106</td>
</tr>
<tr>
<td>8 A PROTOTYPICAL APPLICATION</td>
<td>114</td>
</tr>
<tr>
<td>8.1 Domain Initialization</td>
<td>114</td>
</tr>
<tr>
<td>8.2 Processing</td>
<td>116</td>
</tr>
</tbody>
</table>
9 CFD SIMULATIONS FOR
SEDIMENT TRANSPORT IN DREDGE LANE ........................................................... 123
9.1 Introduction .................................................................................................................. 123
9.2 Sediment transport model .......................................................................................... 126
9.3 Results ......................................................................................................................... 130
9.4 Dredging Scenarios ...................................................................................................... 137
9.5 Conclusions ................................................................................................................. 142

10 CONCLUSION .............................................................................................................. 172
10.1 Applications and Future Directions ............................................................................ 187

APPENDIX

A AN ANALYTICAL SOLUTION FOR MIXING-LENGTH MODEL .................................... 188

B COMPUTER RESULTS .................................................................................................. 191
B.1 Computer Output ....................................................................................................... 191

BIBLIOGRAPHY ................................................................................................................ 194
LIST OF TABLES

Table

3.1 One-parameter family of convection schemes. ......................................................... 25
3.2 Coefficients involved in the source of Reynolds-stress equation. ....................... 46
9.1 Ellipse parameters for scenarios. ........................................................................... 134
LIST OF ILLUSTRATIONS

Figure

1.1 Perhaps the world’s first use of visualization as a scientific tool to study turbulent flow. Graphic by Leonardo da Vinci circa 1500 saved from eFluids website. .......................................................... 2
1.2 Muller-Lyer Illusion - Which line is longer? ......................................................... 4
2.1 Computer Science Knowledge Classification. ....................................................... 6
2.2 A visualization result involving elements from volume rendering, GPU-based programming, and medical visualization. ................................................................. 7
2.3 Classification of flow visualization techniques: from left to right, direct, texture-based, geometric object-based, and feature-based. ......................................................... 12
3.1 Finite volume and storage arrangement. ................................................................. 22
3.2 Grid stencil for one-dimensional case scheme. ....................................................... 25
3.3 Non-staggered grid arrangement. .......................................................................... 31
3.4 Definition of geometrical parameter. ................................................................. 36
3.5 Notation for a damping function on a curved surface. ...................................... 44
3.6 Staggered Reynolds-stress fields. ...................................................................... 47
3.7 Stresses on a tetrahedral element. ...................................................................... 57
3.8 Boundary condition in the near-wall region. ....................................................... 63
3.9 Two-layer model in the low-Reynolds-number region. ....................................... 69
3.10 Relationship between $B_s$ and $k_f$. ○ and + experimental data; solid line, the fit given by (3.181). ................................................................. 73
4.1 Grid generated about a two dimensional hill with no control functions. ....... 76
4.2 Schematic representation of the physical space near the surface $\zeta = \zeta_1$. .... 78
4.3 Grid generated about a two dimensional hill ($a = 0.5, c = 0.4$). ................. 81
4.4 Grid generated about a two dimensional hill ($a = 0.5, c = 1.0$). ................. 82
4.5 Grid generated about a two dimensional hill ($a = 0.2, c = 0.4$). ................. 83
4.6 Grid generated about a two dimensional hill, without transmission of the grid spacing into the interior of the domain ($a = 0.5, c = 0.4$). ................. 83
4.7 Dependence of the sub-relaxation parameter $\gamma$ on the values of $\beta$. ....... 87
4.8 The flow simulation over the Askervin hill (surface $j = \frac{3}{2} n_{i,j}$). ................. 89
5.1 Example CAVE immersive visualization system displaying a terrain data set. 91
5.2 Example visualization of a knot produced with KnotPlot. ............................... 95
5.3 Example 2D refined Delaunay tesselation. ....................................................... 99
5.4 Example Terrain data set with 3D glyphs to show direction of flow. ............. 100
6.1 RAVE configured as a standard CAVE system. .............................................. 103
6.2 RAVE configured as a 30 foot flat wall. ......................................................... 103

vii

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
7.1 Turbulence introduced into flow field by flow around an obstacle ......................................... 105
7.2 3D gridded domain representing a rippled sand bed .......................................................... 107
7.3 Interpolation Scheme ........................................................................................................... 112

8.1 Grid generation on leading wave .......................................................................................... 115
8.2 Grid generation on trailing wave .......................................................................................... 116
8.3 Flow Vectors on leading wave ............................................................................................. 116
8.4 Flow Vectors on trailing wave ............................................................................................. 117
8.5 Application Menu for control the display on the RAVE ........................................................ 118
8.6 Normal vectors of bottom surface ....................................................................................... 119
8.7 3D sand particles at rest ....................................................................................................... 120
8.8 3D sand particles when lefted ................................................................................................ 120
8.9 3D sand particles lefted and suspended .............................................................................. 121
8.10 3D sand particles when falling ........................................................................................... 121
8.11 3D sand particles when falling and deposited ................................................................... 122
8.12 3D sand particles deposited near the bottom surface .......................................................... 122

9.1 Hypothetical distribution of macro-bedforms around an individual dredging lane. Net bed sediment transport is towards the left. A-A' and B-B' are sections across the dredging lane and show the accumulation of individual dredge tracks. The red line shows the bed profile immediately after dredging ceases, green a short period later (months) and the orange lines a long period later (years) ................................................................................................................................................................. 146
9.2 The 2-D curvilinear (non-orthogonal) mesh (Horizontal scale in metres) .............................................. 147
9.3 Numerical (crossed line) and analytic (solid line) profiles of horizontal shear stress \((m^{-2}s^{-2})\), with bed roughness \(z0=0.01\text{ m}\). Crosses show position of numerical grid points ......................................................................................................................... 148
9.4 Numerical (crossed line) and analytic (solid line), profiles of mean streamwise velocity \((\text{ms}^{-1})\), with bed roughness \(z0=0.01\text{ m}\). Crosses show position of numerical grid points ......................................................................................................................... 148
9.5 Numerical versus analytic solution for suspended sediment profiles for 100 and 200 \(\mu\text{m}\) particles (vertical scale expanded to emphasise nearbed values). Solid lines are the analytic solution and the stars the numerical solution ................................................................................................................................. 149
9.6 Vertical profiles of mean velocity \((\text{ms}^{-1})\) over a tidal cycle, a) from Davies & Jones [37]. b) calculated by present model. Note, profiles from the present and previous study were not plotted at exactly the same phase through the tidal cycle as the phase of outputs profiles was not specified in Davies and Jones. The results indicate that the two sets of calculations are consistent with one another ................................................................................................................................................................. 150
9.7 Vertical profiles of shear stress (Nm$^{-2}$) over a tidal cycle, a) from Davies & Jones [37], b) Calculated by present model. Note, profiles from the present and previous study were not plotted at exactly the same phase through the tidal cycle as the phase of outputs profiles was not specified in Davies and Jones. The results indicate that the two sets of calculations are consistent with one another.

9.8 Vertical profiles of turbulent kinetic energy ($m^2s^{-2}$) over a tidal cycle, a) from Davies & Jones [37], b) Calculated by present model. Note, profiles from the present and previous study were not plotted at exactly the same phase through the tidal cycle as the phase of outputs profiles was not specified in Davies and Jones. The results aim to show the consistency of the two sets of calculations rather than to make a quantitative comparison.

9.9 Vertical profiles of turbulent viscosity ($m^2s^{-1}$) over a tidal cycle, a) from Davies & Jones [37], b) Calculated by present model. Note, profiles from the present and previous study were not plotted at exactly the same phase through the tidal cycle as the phase of outputs profiles was not specified in Davies and Jones. The results aim to show the consistency of the two sets of calculations rather than to make a quantitative comparison.

9.10 Nearbed bed cross-track velocity ($V$) for 10° orientation.

9.11 Nearbed bed cross-track velocity ($V$) for 45° orientation.

9.12 Nearbed bed cross-track velocity ($V$) for 90° orientation.

9.13 Turbulent Kinetic Energy (TKE) for track orientation 10° - U tidal velocity oscillates back and forth into and out of the page with the cross-track velocity ($V$) going across the page.

9.14 Turbulent Kinetic Energy (TKE) for track orientation 45° - U tidal velocity oscillates back and forth into and out of the page with the cross-track velocity ($V$) going across the page.

9.15 Turbulent Kinetic Energy (TKE) for track orientation 90° - U tidal velocity oscillates back and forth into and out of the page with the cross-track velocity ($V$) going across the page.

9.16 Bed shear distribution. Track orientation 0°.


9.18 Bed shear distribution. Track orientation 45°.

9.19 Bed shear distribution. Track orientation 90°.

9.20 $Log_{10}$ sediment concentrations for 10° and 100 μm.

9.21 $Log_{10}$ sediment concentrations for 10° and 250 μm.

9.22 $Log_{10}$ sediment concentrations for 10° and 500 μm.

9.23 $Log_{10}$ sediment concentrations for 10° and 100 μm.

9.24 $Log_{10}$ sediment concentrations for 45° and 250 μm.

9.25 $Log_{10}$ sediment concentrations for 45° and 500 μm.

9.26 $Log_{10}$ sediment concentrations for 90° and 100 μm.
9.27 \( \log_iO \) sediment concentrations for 90° and 250 \( \mu m \).

9.28 \( \log_iO \) sediment concentrations for 90° and 500 \( \mu m \).

9.29 Dredge tracks observed using Side-scan sonar from Area 107 in the Outer Wash.

9.30 Side-scan sonar image from Hastings Single bank showing intensive tide parallel dredging with Sub Area Y of the Licence area in 1999.

9.31 Side-scan sonar image from Owers Bank off Shoreham in 2004 showing new and old dredge tracks. The new dredge track (line stretching centre left curving to east) also shows "lips" either side of the dredge tracks due to the draghead.

9.32 Swath Bathymetry showing dredge tracks and dredge pits on the Owers Bank off Shoreham in 2004.

9.33 Snapshot of bed sediment distribution for scenario 1 for 100 \( \mu m \) sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).

9.34 Snapshot of bed sediment distribution for scenario 1 for 250 \( \mu m \) sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).

9.35 Snapshot of bed sediment distribution for scenario 1 for 500 \( \mu m \) sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).

9.36 Time series of bed retention for scenario 1 for various particle sizes.

9.37 Snapshot of bed sediment distribution for scenario 2 for 100 \( \mu m \) sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).

9.38 Snapshot of bed sediment distribution for scenario 2 for 250 \( \mu m \) sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).

9.39 Snapshot of bed sediment distribution for scenario 2 for 500 \( \mu m \) sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).

9.40 Time series of bed retention for scenario 2 for various particle sizes.

9.41 Snapshot of bed sediment distribution for scenario 3 for 100 \( \mu m \) sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).

9.42 Snapshot of bed sediment distribution for scenario 3 for 250 \( \mu m \) sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).

9.43 Snapshot of bed sediment distribution for scenario 3 for 500 \( \mu m \) sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).

9.44 Time series of bed retention for scenario 3 for various particle sizes.
**LIST OF ABBREVIATIONS**

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAT</td>
<td>Computer Aided Tomography</td>
</tr>
<tr>
<td>MRI</td>
<td>Magnetic Resonance Imaging</td>
</tr>
<tr>
<td>GPU</td>
<td>Graphical Processing Unit</td>
</tr>
<tr>
<td>CFD</td>
<td>Computational Fluid Dynamics</td>
</tr>
<tr>
<td>IBFV</td>
<td>Image Based Flow Visualization</td>
</tr>
<tr>
<td>ISA</td>
<td>Image Space Advection</td>
</tr>
<tr>
<td>LEA</td>
<td>Lagrangian-Eulerian Advection</td>
</tr>
<tr>
<td>FV</td>
<td>Finite Volume</td>
</tr>
<tr>
<td>GENOTS</td>
<td>General Non-Orthogonal Turbulent Solver</td>
</tr>
<tr>
<td>LUDS</td>
<td>Linear Upwind Difference Scheme</td>
</tr>
<tr>
<td>QUICK</td>
<td>Quadratic Upstream Interpolation for Convective Kinematics</td>
</tr>
<tr>
<td>CUI</td>
<td>Cubic Upwind Interpolation</td>
</tr>
<tr>
<td>MUSCL</td>
<td>Monotonic Upstream Scheme for Conservation Laws</td>
</tr>
<tr>
<td>RSTM</td>
<td>Reynolds-Stress Transport Model</td>
</tr>
<tr>
<td>CAVE</td>
<td>Cave Automatic Virtual Environment</td>
</tr>
<tr>
<td>VE</td>
<td>Virtual Environments</td>
</tr>
<tr>
<td>CVE</td>
<td>Collaborative Virtual Environments</td>
</tr>
<tr>
<td>DVR</td>
<td>Distributed Virtual Reality</td>
</tr>
<tr>
<td>DIVE</td>
<td>Distributed Interactive Virtual Reality</td>
</tr>
<tr>
<td>IML</td>
<td>Immersive Media Lab</td>
</tr>
<tr>
<td>NewMIC</td>
<td>New Media Innovation Center</td>
</tr>
<tr>
<td>HPC</td>
<td>High Performance computers</td>
</tr>
<tr>
<td>OpenGL</td>
<td>Open Graphics Library</td>
</tr>
<tr>
<td>DIVERSE</td>
<td>Device Independent Virtual Environments:</td>
</tr>
<tr>
<td></td>
<td>- Reconfigurable, Scalable, and Extensible</td>
</tr>
<tr>
<td>DPF</td>
<td>DIVERSE Graphics Interface for Performer</td>
</tr>
<tr>
<td>DGL</td>
<td>DIVERSE OpenGL</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interfaces</td>
</tr>
<tr>
<td>NCSA</td>
<td>National Center for Supercomputing Applications</td>
</tr>
<tr>
<td>NSF</td>
<td>Nation Science Foundation</td>
</tr>
<tr>
<td>PACI</td>
<td>Partnership in Advanced Computational Infrastructure</td>
</tr>
<tr>
<td>CCC</td>
<td>Collaborative CAVE Console</td>
</tr>
<tr>
<td>RAVE</td>
<td>Reconfigurable Advanced Visualization Environment</td>
</tr>
<tr>
<td>USB</td>
<td>Universal Serial Bus</td>
</tr>
<tr>
<td>HMD</td>
<td>Head Mounted Display</td>
</tr>
<tr>
<td>DOF</td>
<td>Degree Of Freedom</td>
</tr>
<tr>
<td>GNU</td>
<td>GNU's Not Unix</td>
</tr>
<tr>
<td>GPL</td>
<td>General Public License</td>
</tr>
<tr>
<td>LGPL</td>
<td>Lesser General Public License</td>
</tr>
<tr>
<td>SGI</td>
<td>Silicon Graphics, Inc.</td>
</tr>
<tr>
<td>EVL</td>
<td>Electronic Visualization Laboratory</td>
</tr>
</tbody>
</table>
Visualization of Scientific Data describes the application of graphical methods to enhance interpretation and meaning of scientific data. Merriam-Webster Online Dictionary defines visualization as 'the formation of mental visual images, the act or process of interpreting in visual terms or of putting into visible form, the process of making an internal organ visible by the introduction (as by swallowing, by an injection, or by an enema) of a radiopaque substance followed by roentgenography'. This dissertation will focus on sediment transport, a subtopic of computational fluid dynamics. Immersive visualization involves surrounding, or immersing the scientist, in a visual sense, into the data set, and allows the scientist to gain visual perceptions from within the spatial confines of the data set. Current visualization methodologies only allow visual perceptions to be developed from outside the data set. Our research involves the development of a new algorithm that allows scientists to gain a better understanding of the physical system being studied in an immersive visualization environment. The proposed algorithm is applied to a prototypical volumetric data set generated from a complex model of turbulent oscillatory flow with sediment transport. We will show that this new algorithm has several computational and visualization advantages over other algorithms currently available.

Scientific data can be derived from various sources, including measuring instruments, or may be obtained as a result of scientific computations performed on supercomputers (Hill [64]). However, data do not become useful until at least some of the information...
Figure 1.1: Perhaps the world’s first use of visualization as a scientific tool to study turbulent flow. Graphic by Leonardo da Vinci circa 1500 saved from eFluilds website.

is extracted (Hagen et al. [56]). The goal of scientific visualization is to provide concepts, methods and tools to create expressive and effective visual representations from scientific data (Laramee [89]). Such visual representations convey new insights and an improved understanding of the underlying physical processes, mathematical concepts and other quantifiable phenomena expressed in the data (Schroeder et al. [144], Tufte [160], Wagon [166], Ware [167], Zabala [177]). See Figure 1.1 for an example of one of the first uses of visualization. Concepts and tools of scientific visualization are based on other disciplines: psychology, perception and human factors offer a scientific basis to understand human visual performance, its abilities and limitations Herr [61]). Experts in computer graphics provide algorithms and tools to transfer numerical data values into pictures. Artists and graphic designers offer their knowledge of aesthetics and other design...
issues to increase interpretability of visual representations. Scientists define their needs to explore scientific data and thus drive the quest for visual exploration (Koning [82], Laszlo [94]). Scientific visualization provides concepts, methods and tools from existing disciplines to best use human abilities and computer algorithms for the display of scientific data (Press et al. [129]).

On the other hand, there is a fine difference between the goals of visualization and goals of other disciplines or subdisciplines. While psychology and perception are important for understanding abilities and limitations of a scientist viewing a picture, basic principles of perception theories and the awareness of visual illusions, such as the Hermann grid or the Muller-Lyer illusion (See Figure 1.2) do not fully explain the complex visual information present in a three-dimensional vector field visualization (Sekuler & Blake [145]). While image processing exclusively deals with images, it uses a limited number of visual representations (gray values or color pixel displays; shaded surfaces) to visually express the result of numerical algorithms. The chosen visual representations are not of essence to image processing, but rather the development of the underlying techniques, such as filters, geometric corrections, or image compression. Computer vision is concerned with the computerized extraction of information from images (Boyle & Thomas [15]). It is therefore not of concern to computer vision how the human viewer extracts information from a picture, but rather how the computer may accomplish a similar task and initiate a certain action dependent on the result. The field of computer graphics provides tools to design pictures from symbolic or numeric descriptions and to interact
with these pictures (Hill [64]). Computer graphics is concerned with the development of algorithms (and their efficiency) to create pictures on a computer display. While computer graphics works hand in hand with visualization, it is not concerned with pictures on displays once their appearance is satisfactory. The extraction of meaning from the picture in the human mind is not of concern to this field. User interface issues have developed in parallel, but separately, from computer graphics (Olsen [121]). The wide availability of bitmapped graphics gave access to new visual appearances of user interfaces that have cumulated in the widespread use of windows and widgets. However, user interface design is not applying its methods to the understanding of processes and data, but rather to the ease-of-use of programs. In a similar way, human factors take into account the problems humans encounter when working with machines, not the output from these machines. In having reviewed a series of areas that add to the understanding of visualization, we can conclude that scientific visualization cannot be replaced by existing disciplines, but offers more than the sum of knowledge derived from these separate disciplines. Therefore, scientific visualization has become a discipline of its own.

In order to understand the complexities of sediment transport, we begin with a history of sediment transport in Chapter 2. Chapter 3 will cover the history of Compu-
CHAPTER 1. INTRODUCTION

tational Fluid Dynamics and specifically how sediment transport models have been derived. Chapter 4 presents the mathematical details involved with the domain Grid Generation schemes used in this work. We focus on related immersive scientific visualization research in Chapter 5. Chapter 6 includes our research environment describing the computing platform, software and graphical libraries used to conduct our research. Chapter 7 includes the theory behind our contribution to sediment transport and how immersive scientific visualization can be used to understand and improve physical models. A prototypical application and results will be presented in Chapter 8. Chapter 9 will include our development of a new Computational Fluid Dynamics sediment transport model and comparison to earlier work in the field. Finally, Chapter 10 will include our conclusions of the test results and future directions of this research.
Where does scientific visualization begin? Refer to the knowledge hierarchy map in Figure 2.1. The hierarchy starts with Computer Science which itself consists of several disciplines shown in the hierarchy. Many researchers would not agree with the classification given in this hierarchy. We chose this classification based on simplicity and research experience. Note that visualization overlaps many other disciplines such as computer graphics, imaging, modeling, and human perception. Also, levels in the hierarchy can continue down to other subjective levels of specialization. We use Figure 2.1 as a visual construct to locate our core topic of immersive visualization in the world of computer research. In 1986 the National Science Foundation sponsored an advisory panel on Graphics, Image Processing and Workstations made recommendations in response to the needs developed by high data rates and the opportunity of using the new generation of

![Figure 2.1: Computer Science Knowledge Classification.](image-url)
Figure 2.2: A visualization result involving elements from volume rendering, GPU-based programming, and medical visualization.

graphics workstations. The widely published report produced by the panel called for new tools in a new field termed Visualization in Scientific Computing, or in short Scientific Visualization (McCormick et al. [112]). Since 1987 a multitude of new applications have confirmed the necessity and power of this new methodology (Boyl et al. [15], Brittain et al. [17], Cox [31], Daniel & Chen [35], Docecek et al. [38], Doleisch et al. [54], Hagen et al.) [56], Koning [82], McCormick et al. [112]). Data can come from a wide variety of sources such as simulation, modeling, measurements, or from nonscientific disciplines such as finance, marketing, and business. The goal of visualization is to gain a deeper understanding of data. Visualization allows us to see structures and find patterns that we are unable to see from a vast array of raw numbers. Some (but not all) of the classic
CHAPTER 2. SCIENTIFIC VISUALIZATION

sub-topics of visualization are:

1. volume visualization: a methodology for visualizing 3-D data that may use discrete polygonal primitives or volume rendering techniques. Volume rendering is based on the voxel primitive (Yacoob [176]). Data sources often come from the medical domain, e.g., computer aided tomography (CAT) or magnetic resonance imaging (MRI)(See Figure 2.2).

2. information visualization: assigns an abstract geometry or topology to data that does not already have an inherent geometric representation. Data sources are often financial or economic in nature. Example visualization techniques include the use of pie charts, scatter plots, and parallel coordinates (Ware [167]).

3. GPU-based techniques: a rapidly growing area of research is centered around programmable graphical processing units (GPU)s. The goal is to speed up computation, that might otherwise take place on the CPU by taking advantage of the computing features offered by the GPU.

4. medical visualization: this is the area that many people intuitively connect with the field of visualization. Medical visualization techniques illustrate subsets of the human body, such as the skeleton or brain, using data generated by medical tools such as CAT scanners. Clearly advances in medicine are a strong driving factor for innovation in this field.
2.1 Flow Visualization

Flow visualization is one of the classic subfields of visualization, covering a rich variety of applications, from the automotive industry, aerodynamics, turbomachinery design, to weather simulation, meteorology, climate modeling, ground water flow, and medical visualization. Consequently, the spectrum of flow visualization solutions is very rich, spanning multiple technical challenges: 2D vs. 3D solutions and techniques for steady or time-dependent data. Several options of subdividing this broad field of literature are possible. Hesselink et al., for example, addressed the problem of how to categorize techniques in their 1994 overview of research issues and considered dimensionality as a means to classify the literature (Hesselink et al.[62]). In the following, several aspects are discussed on an abstract level before literature is addressed directly. The majority of the remainder of this chapter has been published elsewhere (Laramee et al. [92], Post et al. [126, 127]).

Flow visualization involves the visualization of vector data. Each vector has a spatial component and a direction component. Vector data often results from the study of Computational Fluid Dynamics (CFD) where fluid flows or derived quantities are involved. The first and simplest way to visualize vector data is to use glyphs. Glyph-based visualization is a technique to visualize multi-variate data. A glyph is an icon or graphical object that has certain properties such as $x, y, z$ location in 3D space, color, opacity, orientation, shape and size. A single dimension of data is mapped to each of these attributes of the glyph. An array of glyphs is created which is then volume rendered to visualize the data.
The simplest glyph is just a line whose length is varied to represent the vector magnitude and orientation represents the vector direction. A line does not indicate which end is the head or tail. Other glyphs can be used such as an arrow, tetrahedron, solid cone, etc to make it easier to discern direction. Color can be used with the glyph to represent an additional vector component. In this case, a color scale must be included with the graphic to aid observer interpretation. However, more complicated glyphs require more processing and may affect display performance. There are several problems to note in using glyphs including occlusion, visual complexity, problems with placement, either too sparse or too dense, problems with interpretation (Tufte [160]), lack of spatial coherency, and more (Laramee [89]). A more detailed discussion will be presented later in this chapter and in Chapter 4.

To understand flow visualization, consider the basic equation of motion for velocity of a massless particle:

\[ \mathbf{v} = \frac{d\mathbf{x}}{dt}. \]  

(2.1)

through a vector field, we need to know the particle’s position as a function of time and initial position which can be solved by expressing equation (2.1) in integral form:

\[ x(t, x_0) = x_0 + \int_{\tau=0}^{t} \mathbf{v}(\tau)d\tau. \]  

(2.2)

Equation (2.2) is fundamental to the study of flow visualization. In general, all geometric flow visualization techniques (Post et al. [126]), texture-based techniques and feature-based approaches (Post et al. [127]) relate to equation 2.2 in some way (Laramee [89]).
CHAPTER 2. SCIENTIFIC VISUALIZATION

For computer simulation purposes, equation (2.2) cannot be solved analytically, and must resort to numerical integration techniques which yields an approximate solution. One such approximation is given by Euler's first order numerical scheme:

\[ x_{i+1} = x_i + v_i \Delta t \]  

(2.3)

where the position of the particle at time \( i+1 \) is given by the sum of the previous position and the product of the velocity and an incremental time \( \Delta t \). A major concern with any numerical integration is the truncation error. For the equation (2.3) the truncation error is \( O(\Delta t^2) \) and is not accurate enough in many cases (Schroeder et al. [144]). Hence, more accurate alternatives are needed such as the second-order Runge-Kutta integrator (Ralston & Rabinowitz [130]):

\[ x_{i+1} = x_i + \frac{\Delta t}{2} (v_i + v_{i+1}) \]  

(2.4)

where velocity \( v_{i+1} \) is computed using equation (2.3). The truncation error is in equation (2.4) \( O(\Delta t^3) \) (Ralston & Rabinowitz [130]). A practical advantage to equation (2.4) is a larger integration step is possible for the cost of one additional function evaluation. Higher order integrators are also available such as the fourth-order Runge Kutta method. How these computational math concepts are implemented will be discussed in detail in chapters 3 and 4.

2.1.1 Classification of Flow Visualization

Four different approaches are widely used in flow visualization (Post et al. [126]): direct, geometric, texture-based, and feature-based flow visualization. See Figure 2.3.
Direct flow visualization: This category of techniques uses a translation that is as straightforward as possible for representing flow data in the resulting visualization. The result is an overall picture of the flow. Common approaches are drawing arrows or color coding velocity. Intuitive pictures can be provided, especially in the case of two dimensions. Direct flow visualization approaches are amongst the oldest available techniques and therefore well known. The use of color mapping in glyphs is standard in graphics software. Hence, this is not an area of focus in this dissertation. However, we do make use of direct flow visualization in the context of a resampling approach as detailed in Chapter 7, as well as in other cases. As such, direct flow visualization is a closely related topic.

Geometric flow visualization: These approaches often first integrate the flow data and use geometric objects in the resulting visualization. The objects have a geometry that reflects the properties of the flow. Examples include streamlines, streaklines, and timelines.
CHAPTER 2. SCIENTIFIC VISUALIZATION

Not all geometric objects are based on integration. Another useful geometric approach is to generate isosurfaces, e.g., with respect to an isovalue of pressure or magnitude of velocity. Geometric flow visualization techniques are one topic of research within this dissertation.

Dense, texture-based flow visualization: A texture is computed that is used to generate a dense representation of the flow. A notion of where the flow moves is incorporated through co-related texture values along the vector field. In most cases this effect is achieved through filtering of texels according to the local flow vector. Texture-based methods offer a dense representation of the flow with complete coverage of the vector field. In this dissertation we use advection approaches according to Image Based Flow Visualization (IBFV) (Wijk [170]) and Image Space Advection (ISA) (Laramee et al. [90]), which can generate both Spot Noise (Wijk [169]) and LIC-like (Cabral & Leedom [21]) imagery. Both approaches are related to Lagrangian-Eulerian Advection (LEA) (Jobard et al. [75]). A full comparison of texture-based flow visualization techniques is given in the next chapter. We focus on interactive visualization techniques because an interactive exploration of parameter space is essential for improving the design of modeled or simulated components that undergo CFD analysis. The long computation time associated with texture-based approaches has been a problem since the introduction of these techniques themselves starting in the early 1990s (Cabral & Leedom [21], Wijk [169]). The computation time barrier was hurdled with the introduction of LEA (Jobard et al. [74]) and again with IBFV (Wijk [170]). IBFV solved the computation time problem for the case
CHAPTER 2. SCIENTIFIC VISUALIZATION

of 2D, unsteady flow. Additional research has been carried out to extend texture-based techniques to surfaces in 3D for both steady and unsteady flow (Laramee [89]). With the introduction of ISA (Laramee et al. [90]) and IBFVS (IBFV for Curved Surfaces) (Wijk [171]) we saw texture-based flow visualization on surfaces at fast frame rates for the first time. And since these methods were introduced to the visualization community at the same time, a comparison of the two approaches was a natural choice for further research (Laramee et al. [91]).

Feature-based flow visualization: For the sake of completeness, we note that feature-based flow visualization, another class of techniques including feature extraction and tracking, is beyond the scope of this dissertation. Other researchers, such as Post et al., have covered feature-based flow visualization in detail (Post et al.[126]). See Doleisch et al. ([53, 54]) for even more recent research results in this area.

Immersive flow visualization: This is the latest area of visualization research that allows the user to view and interact with the data while being immersed within the spatial confines of the data (Ebert et al. [41]). One of the major problems that hindered immersive visualization research until now is the availability of a visualization system. Computer engineering and manufacturing capabilities had to progress to the point where individual components such as computer processors, memory, networking, and specialized high-performance graphics hardware could meet the processing and display requirements of an immersive system. Prototype systems were very expensive, but recent advances have made these systems more affordable. In order to create a true 3D display, the data
must be rendered in multiple planes simultaneously. The type of display created is stereographic and requires a special set of glasses to view the true 3D structures correctly. This area of research is growing in popularity and holds the most promise for achieving greater advances in many industries such as automotive, aerospace, nuclear, pharmaceutical, chemistry, medicine and more.

In general, direct flow visualization techniques require less computation than the other three categories, whereas immersive visualization techniques require the most computation. This dissertation focuses on the body of research related to immersive visualization using direct numerical simulation modeling techniques.
3.1 Introduction

The aid of any computational procedure for fluid dynamics and turbulence may be argued to hinge on the following principle issues:

- the adaptability of the procedure to complex geometric features;
- its numerical accuracy;
- the realism of mathematical models embedded in the procedure, for example, that representing effects of turbulence;
- the procedure's computational economy;
- its applicability over a wide range of flow conditions;
- the ease of its application.

Hitherto, there does not exist any numerical technique that satisfies even remotely all the above desiderata. Procedures developed in the commercial, industrial, and environmental are often highly flexible in terms of geometry, possess sophisticated pre- and post-processing capabilities and are often robust too. However, they also tend to embody relatively simple numerical discretization techniques and even simpler turbulence models, both of which significantly enhance robustness and economy. Conversely, inflexible algorithms tend to incorporate front-line mathematical models and discretization schemes.
Over the past fifteen years, much efforts have been devoted to advance the state of turbulence modeling for industrial and environmental applications (Launder [97], Leschziner [103], Sajjadi & Waywell [142]). These efforts have focused, almost exclusively, on developing and validating variants of second-moment closure. In essence, this type of model consists of non-linear, highly coupled differential equations, each expressing a balance between transport, generation, redistribution and destruction of time-averaged turbulence correlations representing turbulent fluxes of momentum, heat and other conserved flow properties. The principal advantage of this approach, in marked contrast to those widely used at present in an industrial context, is that it accounts for anisotropy of mixing processes arising from the interaction between turbulence, swirl, streamline curvature and buoyancy forces. However, drawbacks include greater computational effort, reduced iterative stability and the need for more refined numerical approximations ensuring an adequate, error-free resolution of the often very steep flow-property gradients.

Much of the above work has progressed initially within a simple geometrical environment involving incompressible thin shear flow and recirculating flow contained within rectilinear boundaries (Pope & Witelaw [125], Gibson & Rodi [49]). Major objectives pursued at that stage included a careful assessment of basic model characteristics by reference to data emerging from well controlled experimental studies, a stable implementation of complex closure variants in conjunction with numerically non-diffusive, higher-order schemes approximating convection, and wide-ranging validations. More recently, there have been efforts to branch into geometrically more challenging configurations by means
of algorithms using curved orthogonal meshes (e.g. Kadja [77], Manners [108], Lien & Leschziner [104], Sajjadi [138, 139]). While these did permit more complex geometries to be investigated, the orthogonal constraint was found to pose, even with multi-block arrangements, serious difficulties in terms of grid generation and control, resulting in undesirable local depletion and clustering of mesh nodes as geometric complexity rose.

The realization that much greater meshing flexibility was essential for extending the range of validation of advanced turbulence closures, particularly into 3D flows, provided one strong incentive for pursuing the work reported herein. Within that extended framework, fresh challenges arise from the sheer complexities of models formulated in a general non-orthogonal coordinate system (Sajjadi et al. [141]), the lack of a numerical stabilization mechanism, and from the tendency for resource requirements to rise rapidly due to slow iterative convergence and the use of dense grids. The above issues are among many addressed in the course of evolving the present numerical framework, which combines the following main elements and characters:

1. applicable to complex geometries by use of a general non-orthogonal, cell-collocated formulation;

2. incorporating accurate (numerical non-diffusive) higher-order approximation for convection;

3. incorporating a range of turbulence closures, among them a Reynolds-stress transit port model and a low-Reynolds-number eddy-viscosity variant applicable to the
3. formulated on the basis of a pressure-correction methodology.

The associated Finite Volume (FV) software, to be described here, has been given the acronym GENOTS, standing for General Non-Orthogonal Turbulent Solver. The present paper deals with all of the above items and associated issues in related sections. The application of the algorithm to various diverse test problems provides a firm basis for conclusions to be drawn, and will be presented in the second part of this paper.

3.2 Finite-Volume Discretization

The partial differential equations governing mass conservation and the transport of momentum and any conserved scalar quantity can be expressed in terms of general tensor notation relating to the general coordinates $\xi_j$ as follows:

Continuity

$$\frac{\partial}{\partial t} (J\rho) + \frac{\partial}{\partial \xi^j} [U^j \rho \phi] = 0, \quad (3.1)$$

Momentum $u^i$

$$\frac{\partial}{\partial t} (Jp u^i) + \frac{\partial}{\partial \xi^j} \left[ U^j p u^i - \mu J \left( q_{mn}^{in} \frac{\partial u^i}{\partial \xi^j} \right) \right] = \frac{\partial}{\partial \xi^j} \left[ -J p \beta_i^j + (\mu + \lambda) J \left( q_{mn}^{in} \frac{\partial u^m}{\partial \xi^n} \right) \right], \quad (3.2)$$

Scalar $\phi$

$$\frac{\partial}{\partial t} (J \phi) + \frac{\partial}{\partial \xi^j} \left[ U^j p \phi - \Gamma \phi J \left( q_{mn}^{in} \frac{\partial \phi}{\partial \xi^n} \right) \right] = JS_\phi, \quad (3.3)$$
where the contravariant velocity vector \( U^j \) and the fourth-order tensor \( q_{mi}^{jn} \) are given by

\[
U^j = \beta^j_m u^m, \quad q_{mi}^{jn} = \beta^j_m \beta^n_i,
\]

and \( \beta^j_m \) are the elements of the inverse Jacobian matrix \( \partial(\xi, \eta, \zeta)/\partial(x, y, z) \), whose elements in two-dimensional field is given by

\[
\beta^1_1 = \xi_x, \quad \beta^1_2 = \xi_y, \quad \beta^2_1 = \eta_x, \quad \beta^2_2 = \eta_y.
\]

In the present exposition \( u^j \) are held to be the Cartesian components of the velocity vector. In the actual numerical procedure, an alternative decomposition is also permitted, whereby an arbitrary datum line may be chosen (say, the centreline of a curved geometry) with one component of the orthogonal velocity components made to remain tangential to this line.

For future considerations, it is instructive to express equation (3.3) in a more explicit manner. With \( \xi, \eta, \zeta \) representing curvilinear coordinates in a 3D framework, this equation may be expressed as:

\[
\frac{\partial \rho \phi}{\partial t} + \frac{\partial}{\partial \xi} (\rho U \phi) + \frac{\partial}{\partial \eta} (\rho V \phi) + \frac{\partial}{\partial \zeta} (\rho W \phi) = \underbrace{J S_{\phi}}_{\text{source}}
\]

\[
+ \frac{\partial}{\partial \xi} \left[ (\Gamma_\phi J q_{11}) \frac{\partial \phi}{\partial \xi} \right] + \frac{\partial}{\partial \eta} \left[ (\Gamma_\phi J q_{22}) \frac{\partial \phi}{\partial \eta} \right] + \frac{\partial}{\partial \zeta} \left[ (\Gamma_\phi J q_{33}) \frac{\partial \phi}{\partial \zeta} \right]
\]

where the contravariant velocities \( (U, V, W) \) are given by:

\[
U = J(u \xi_x + v \xi_y + w \xi_z), \quad V = J(u \eta_x + v \eta_y + w \eta_z), \quad W = J(u \zeta_x + v \zeta_y + w \zeta_z).
\]
the coefficients \( q_{11}, q_{22}, q_{33} \) are:

\[
q_{11} = \xi_x^2 + \xi_y^2 + \xi_z^2, \quad q_{22} = \eta_x^2 + \eta_y^2 + \eta_z^2, \quad q_{33} = \zeta_x^2 + \zeta_y^2 + \zeta_z^2.
\]  

(3.8)

and the Jacobian, \( J \), is defined by

\[
J = x_x y_\eta z_\zeta + x_\xi y_\eta z_\zeta - x_y y_\eta z_\zeta - x_\eta y_\xi z_\zeta + x_\eta y_\zeta z_\xi - x_\eta y_\xi z_\xi.
\]  

(3.9)

On the assumption \( \Delta \zeta = \Delta \eta = \Delta \zeta = 1 \), \( J \) is, in fact, the volume of a cell over which the flow-governed equation is integrated. This interpretation opens a convenient route to an evaluation of \( J \) (see Kordulla & Vinokur for details [83]).

For the Reynolds stress model, adopted here, the transport equation for \( \phi \) consists of the following variables (see Docecek et al. [38]): three normal stresses \( \overline{u'^2}, \overline{v'^2}, \overline{w'^2} \), three shear stresses \( \overline{u'v'}, \overline{u'w'}, \overline{v'w'} \), the turbulent dissipation \( \varepsilon \) (see section 3.5), and the sediment concentration \( C \) (see section 9.2). Note that, the concentration fluxes, i.e., \( \overline{u'c'}, \overline{v'c'}, \overline{w'c'} \), where \( c' \) is the fluctuating concentration, are not calculated from their respective transport equations and are instead prescribed by algebraic relationships.
Integration of equation (3.6) over the finite volume shown in Figure 3.1, and application of Gauß Divergence Theorem in conjunction with central differencing for diffusion yields a balance of the rate of change in $\phi$, face fluxes and volume-integrated net source. The transient term is approximated by the (first-order) one-sided difference simply to enable a time-marching solution towards the steady state, the face-values of $\phi$ are initially approximated by the first-order upwind scheme, any cell-corner values involved in cross-diffusion terms are evaluated by trilinear interpolation, and the sources are discretized via a single-point quadrature and linearized as follows:

$$JS_\phi = S_\rho \phi_p + S_C$$

(3.10)

with $S_\rho$ being so chosen as to be unconditionally negative. Insertion of the above approx-
imations into the volume-integrated equation gives:

\[
A_p \phi_p = \sum_{m=E,W,N,S,T,B} A_m \phi_m + S_C + \left( \frac{\rho U}{\Delta t} \right)_p \phi_p^o
\]  

(3.11)

where

\[
A_E = (\Gamma q_{11})_e - \langle (\rho U)_e, 0 \rangle,
\]

\[
A_W = (\Gamma q_{11})_w - \langle (\rho U)_w, 0 \rangle,
\]

\[
A_N = (\Gamma q_{22})_n - \langle (\rho V)_n, 0 \rangle,
\]

\[
A_S = (\Gamma q_{22})_s - \langle (\rho V)_s, 0 \rangle,
\]

\[
A_T = (\Gamma q_{33})_t - \langle (\rho W)_t, 0 \rangle,
\]

\[
A_B = (\Gamma q_{33})_b - \langle (\rho W)_b, 0 \rangle,
\]

and

\[
A_p = A_E + A_W + A_N + A_S + A_T + A_B - S_p + \left( \frac{\rho J}{\Delta t} \right)_p
\]  

(3.12)

here the superscript \( o \) is used to denote the previous time level. In the above, face diffusivity’s are evaluated by linear interpolation using neighboring nodes on either side of any face being considered. Since the grid arrangement employed here is non-staggered, chequerboard oscillations will generally arise if face velocities are linearly interpolated between the related pairs of adjacent node values. To avoid this problem, a special interpolation practice which is an extension of that proposed by Rhie & Chow [131] has been used and will be described in section 3.4. First, however, a class of high-order bounded
3.3 Convection Schemes

The approximation of convection poses challenges which might not be expected at first sight. The problem is, essentially, one of reconciling stability, boundedness and accuracy. Since convection involves first-order derivatives, its most natural approximation is a first-order difference or some related form. However, while such an approximation may be arranged to be fully bounded and thus highly stable, it is also highly diffusive in general multi-dimensional flow and hence often unacceptably erroneous. Reverting to higher-order schemes is a possible route, but these schemes tend to generate parasitic solutions manifesting themselves by unbounded oscillations. A compromise may be attained by adopting an upwind-weighted higher-order scheme, which involves even-order smoothing terms of order 4 or higher. Thus, an alternative route involves a combination of a higher-order symmetric (hence oscillatory) scheme and an interactive oscillation-detecting and damping mechanism which introduced a carefully measured amount of diffusion locally.

A total of three higher-order convection scheme have been embedded in the procedure outlined in Section 3.2. All may conveniently be written in a canonical form involving only one scheme-related parameter. Consider, the following difference operators which
Figure 3.2: Grid stencil for one-dimensional case scheme.

are defined by reference to Figure 3.2 by

<table>
<thead>
<tr>
<th>Scheme</th>
<th>( \kappa )</th>
<th>expression for ( \phi_e ) when ( U_e &gt; 0 )</th>
<th>expression for ( \phi_e ) when ( U_e &lt; 0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUDS</td>
<td>-1</td>
<td>( \frac{3}{2} \phi_p - \frac{1}{2} \phi_W )</td>
<td>( \frac{3}{2} \phi_E - \frac{1}{2} \phi_{EE} )</td>
</tr>
<tr>
<td>QUICK</td>
<td>( \frac{1}{2} )</td>
<td>( \frac{3}{2} \phi_E + \frac{1}{2} \phi_P - \frac{1}{2} \phi_W )</td>
<td>( \frac{3}{2} \phi_P + \frac{3}{2} \phi_E - \frac{3}{2} \phi_{EE} )</td>
</tr>
<tr>
<td>CUI</td>
<td>( \frac{3}{2} )</td>
<td>( \frac{3}{2} \phi_E + \frac{3}{2} \phi_P - \frac{3}{2} \phi_W )</td>
<td>( \frac{3}{2} \phi_P + \frac{3}{2} \phi_E - \frac{3}{2} \phi_{EE} )</td>
</tr>
</tbody>
</table>

Table 3.1: One-parameter family of convection schemes.
\begin{align*}
\phi_e &= \phi_P + \frac{1}{3}[(1 - \kappa)\Delta_e^- + (1 + \kappa)\Delta_e] \quad \text{if } U_e > 0, \quad (3.13) \\
\phi_e &= \phi_E - \frac{1}{3}[(1 - \kappa)\Delta_e^+ + (1 + \kappa)\Delta_e] \quad \text{if } U_e < 0, \quad (3.14)
\end{align*}

where

\[ \Delta_e^- = \phi_P - \phi_W, \quad \Delta_e = \phi_E - \phi_P, \quad \Delta_e^+ = \phi_{EE} - \phi_E, \quad (3.15) \]

Unbounded schemes arise by setting \( \kappa \) to \(-1, 1/2 \) and \( 1/3 \) which respectively yield the LUDS (Linear Upwind Difference Scheme), QUICK (Quadratic Upstream Interpolation for Convective Kinematics) and CUI (Cubic Upwind Interpolation) schemes. Resulting expressions for the face value \( \phi_e \) are summarized in Table 3.1.

A number of comparative studies have been conducted in the 1980's (e.g. Leschziner [102], Han et al. [57] to investigate the relative merits of the schemes QUICK, SUDS, LUDS and first-order variants (e.g. Power-law scheme of Patankar [122]. While different studies gave rise to a range of conclusions, it is fair to assert that QUICK has emerged as offering, overall, the best comprise in terms of accuracy and stability, although its tendency to provoke oscillations in regions of steep property variations has prevented its use for solving transport equations for turbulence quantities. Subsequent bounded or nearly-bounded forms of QUICK have been formulated by Leonard (SHARP) [101], Gaskell (SMART) [47], and Zhu (LODA) [179], and these have permitted the utility of QUICK to be extended to turbulence transport equations. In GENOTS, boundedness is achieved by way of van Leer’s MUSCL (Monotonic Upstream Scheme for Conservation Laws, 1979).
approach. The MUSCL scheme may be written in terms of a slightly modified form of
the canonical representations (3.13) and (3.14) as follows:

\[
\phi_e = \phi_p + \frac{1}{4}[(1 - \kappa)\bar{\Delta}_e^- + (1 + \kappa)\bar{\Delta}_e] \quad \text{if } U_e > 0, \tag{3.16}
\]

\[
\phi_e = \phi_E - \frac{1}{4}[(1 - \kappa)\bar{\Delta}_e^+ + (1 + \kappa)\bar{\Delta}_e] \quad \text{if } U_e < 0, \tag{3.17}
\]

where

\[
\bar{\Delta}_e^- = \minmod(\Delta_e^-, \omega \Delta_e), \quad \bar{\Delta}_e = \minmod(\Delta_e, \omega \Delta_e^-),
\]

\[
\bar{\Delta}_e = \minmod(\Delta_e, \omega \Delta_e^+), \quad \bar{\Delta}_e^+ = \minmod(\Delta_e^+, \omega \Delta_e),
\]

and the \textit{minmod} function is defined by

\[
\minmod(A, \omega B) = \sgn(A) \max\{0, \min\{|A|, \omega B \sgn(A)|\}, \tag{3.18}
\]

with \(1 \leq \omega \leq \frac{3 - \kappa}{1 - \kappa} (\kappa \neq 1)\). In order to simplify equations (3.16) and (3.17) without loss
of generality, \(\omega\) may be set to unity. Since \(\bar{\Delta}_e^- = \bar{\Delta}_e\) and \(\bar{\Delta}_e = \bar{\Delta}_e^+\), equations (3.16) and
(3.17) reduce to:

\[
\phi_e = \phi_p + \frac{1}{2}\minmod(\Delta_e^-, \Delta_e) \quad \text{if } U_e > 0, \tag{3.19}
\]

\[
\phi_e = \phi_E - \frac{1}{2}\minmod(\Delta_e^+, \Delta_e) \quad \text{if } U_e < 0, \tag{3.20}
\]

The above permits the interpretation that MUSCL introduces as an artificial diffusion
controlled by a scale involving the minimum absolute slopes at the east-face and that at the
upstream position. Note that, the expressions (3.13)–(3.14) and (3.19)–(3.20) include the
fragments of the first-order upwind scheme. Therefore, a \textit{Deferred Correction} approach
is implemented by introducing an additional source \(S_{DC}^{\phi}\) to equation (3.10).
Thus, for any unbounded scheme in Table 2.1:

$$S^D_C = \frac{1}{2} \{ \rho_c U_c U_e^+ [(1 - \kappa)\Delta_e^- + (1 + \kappa)\Delta_e] - \rho_c U_c U_e^- [(1 - \kappa)\Delta_e^+ + (1 + \kappa)\Delta_e]$$

$$- \rho_n U_n U_w^+ [(1 - \kappa)\Delta_n^- + (1 + \kappa)\Delta_n] + \rho_n U_n U_w^- [(1 - \kappa)\Delta_n^+ + (1 + \kappa)\Delta_n]$$

$$+ \rho_n V_n V_n^+ [(1 - \kappa)\Delta_n^- + (1 + \kappa)\Delta_n] - \rho_n V_n V_n^- [(1 - \kappa)\Delta_n^+ + (1 + \kappa)\Delta_n]$$

$$- \rho_s V_s V_s^+ [(1 - \kappa)\Delta_s^- + (1 + \kappa)\Delta_s] + \rho_s V_s V_s^- [(1 - \kappa)\Delta_s^+ + (1 + \kappa)\Delta_s]$$

$$+ \rho_l W_l W_l^+ [(1 - \kappa)\Delta_l^- + (1 + \kappa)\Delta_l] - \rho_l W_l W_l^- [(1 - \kappa)\Delta_l^+ + (1 + \kappa)\Delta_l]$$

$$- \rho_b W_b W_b^+ [(1 - \kappa)\Delta_b^- + (1 + \kappa)\Delta_b]$$

$$+ \rho_b W_b W_b^- [(1 - \kappa)\Delta_b^+ + (1 + \kappa)\Delta_b] \} \quad (3.21)$$

While for the MUSCL scheme:

$$S^D_C = \frac{1}{2} \{ \rho_c U_c U_e^+ \text{minmod}(\Delta_e, \Delta_e^-) - \rho_c U_c U_e^- \text{minmod}(\Delta_e, \Delta_e^+)$$

$$- \rho_n U_n U_w^+ \text{minmod}(\Delta_n, \Delta_n^-) + \rho_n U_n U_w^- \text{minmod}(\Delta_n, \Delta_n^+)$$

$$+ \rho_n V_n V_n^+ \text{minmod}(\Delta_n, \Delta_n^-) - \rho_n V_n V_n^- \text{minmod}(\Delta_n, \Delta_n^+)$$

$$- \rho_s V_s V_s^+ \text{minmod}(\Delta_s, \Delta_s^-) + \rho_s V_s V_s^- \text{minmod}(\Delta_s, \Delta_s^+)$$

$$+ \rho_l W_l W_l^+ \text{minmod}(\Delta_l, \Delta_l^-) - \rho_l W_l W_l^- \text{minmod}(\Delta_l, \Delta_l^+)$$

$$- \rho_b W_b W_b^+ \text{minmod}(\Delta_b, \Delta_b^-) + \rho_b W_b W_b^- \text{minmod}(\Delta_b, \Delta_b^+) \} \quad (3.22)$$

where $\Delta_m^\pm, m = w, n, s, t, b$ can be represented similar to equation (3.15). The definition of $U_e^\pm$ is given by

$$U_e^\pm = \frac{1 \pm \text{sgn}(U_e)}{2}, \quad (3.23)$$

and similar expressions apply to $U_w^\pm, V_n^\pm$, etc.
3.4 Pressure Correction Algorithm

The previous section provided a numerical framework for solving any transport equation for a related property $\phi$. This includes the momentum components $\phi = u, v, w$. There remains the question of how to determine the pressure $p$ appearing in the momentum equation. Since the pressure is governed, indirectly, by the continuity equation, attention must focus on this equation and its linkage to the equations of motion. Although there are several routes to take, the most successful approaches are those so-called pressure or pressure-correction, algorithms (e.g. SIMPLE [122], SIMPLR [164], PISO [72]). All are based on combining the discretized forms of the momentum and continuity equations to give an equation interlinking the pressure or pressure correction at a node to its neighbors.

Within an iterative solution sequence, $u, v, w$ are initially obtained with an estimated pressure field. This is continuously updated by reference to local mass residuals (continuity defects), which are used to steer the pressure field towards the correct level. The stability of this iterative sequence relies, to a large extent, on the retention of a strong coupling between perturbations in pressure gradient and velocity. This requirement has given rise to the widely used staggered arrangement of pressure and velocity. The key feature of this approach is that the mass flux across face of the FV over which continuity is to be satisfied is driven by a pressure difference evaluated with pressures at nodes straddling the mass-flux location. In the present effort, a collocated, non-staggered formulation has been adopted, however, and the above key feature is absent. This tends to
CHAPTER 3. COMPUTATIONAL FLUID DYNAMICS

provokes instability due to the onset of odd-even (chequerboard) oscillation arising from
an inappropriate decoupling between the pressure and velocity fields. In 1983, Rhie &
Chow proposed a non-linear interpolation scheme which allowed stability to be regained
within a collocated arrangement (Rhie & Chow [131]). The rationale of the method is
best conveyed by a one-dimensional analysis of the discretized equation, in steady state,
with reference to Figure 3.3. The equation for $u_p$ is given by:

$$A_p u_p = \sum_{m=E,W} A_m u_m + S_C + D_{U_p} (p_w - p_e) p,$$

which may be rewritten as:

$$u_p = \frac{H_p}{A_p} + \frac{D_{U_p}}{A_p} (p_w - p_e) p.$$  \hspace{1cm} (3.25)

Then the Rhie & Chow interpolation for $u_e$ is:

$$u_e = \frac{1}{2} \left[ u_p - \frac{D_{U_p}}{A_p} (p_w - p_e) p + u_e - \frac{D_{U_E}}{A_E} (p_w - p_e) E \right] + \frac{1}{2} \left( \frac{D_{U_p}}{A_p} + \frac{D_{U_E}}{A_E} \right) (p_p - p_E),$$  \hspace{1cm} (3.26)

which may be written in the form:

$$u_e = \frac{1}{2} (u_p + u_e)$$ linear interpolation

$$+ \frac{1}{2} \left( \frac{D_{U_p}}{A_p} + \frac{D_{U_E}}{A_E} \right) (p_p - p_E) - \frac{D_{U_p}}{A_p} (p_w - p_e) p - \frac{D_{U_E}}{A_E} (p_w - p_e) E,$$

pressure smoothing
The key feature in equation (3.26) is that the velocity at the east face is modified such that it is directly linked to the two adjacent pressure nodes \( p_P \) and \( p_E \), as indicated in equation (3.26) above. However, a different point of view is conveyed by equation (3.27), which states that the interpolation practice consists of a centered approximation for \( u_e \) augmented by a stabilizing pressure-smoothing terms. If one assumes \( DU_P = DU_E \) and \( A_P = A_E \), and the values of \( p_e \) and \( p_w \) are linearly interpolated, then the pressure smoothing becomes:

\[
\frac{DU_P}{4A_P} (p_{FE} - 3p_E + 3p_P - p_W),
\]

(3.28)
which is, in essence, a third-order artificial dissipation term. Since the face velocity \( u_e \)
enters the convective flux, \( u_e(pu)Δy \), of the \( u \)-momentum equation, interpolation equa-
tion (3.27) introduces a fourth-order smoothing term into the momentum equation, even if \( \frac{dp}{dx} \) is approximated by a central difference. Within an iterative sequence of the type considered here, it is standard practice to under-relax the changes derived from the mo-
mentum equations such as equation (3.25) in order to enhance stability. Under-relaxation
may be implemented in equation (3.25) via the modification:

\[
\text{Ap} = \frac{\text{Ap}}{\alpha}, \quad \text{Hp} = \frac{\text{Hp} + \frac{1-\alpha}{\alpha} \text{Ap} u_p^*}{\text{Ap}/\alpha},
\]

where \( \alpha \) is the under-relaxation factor and \( u_p^* \) is the value of \( u_p \) at the previous iteration.

It is readily shown that a direct application of equation (3.26) gives rise to an non-unique expression of \( u_e \), depending on the value of \( \alpha \). Solutions have been proposed to remedy the problem (Miller & Schmidt [115], Majumdar [107]) proposed similar remedies to cure this problem. To re-interpret the underlying concept of their approaches, equation
(3.25) with incorporation of equation (3.29) is rearranged in the following form:

\[
\frac{u_p}{\alpha} - \left(1 - \frac{\alpha}{\alpha}\right)u_p^* = \frac{DU_p}{A_p}(p_w - p_e)p = \frac{H_p}{A_p},
\]

with corresponding expressions for \( u_E \) and \( u_E' \):

\[
\frac{u_E}{\alpha} - \left(1 - \frac{\alpha}{\alpha}\right)u_E^* = \frac{DU_E}{A_E}(p_w - p_e)E = \frac{H_E}{A_E},
\]

\[
\frac{u_E'}{\alpha} - \left(1 - \frac{\alpha}{\alpha}\right)u_E^* = \frac{DU_E}{A_E}(p_w - p_e)E = \frac{H_E}{A_E}.
\]
A route to circumventing the dependence of \( u_e \) on \( \alpha \) is based on the assumption that:

\[
\frac{H_e}{A_e} = \frac{1}{2} \left( \frac{H_P}{A_P} + \frac{H_E}{A_E} \right), \tag{3.33}
\]

which does not involve \( \alpha \). In the case of constant \( \alpha \), equation (3.33) can be further expanded to yield:

\[
u_e = \frac{u_P + u_E}{2} + (1 - \alpha) \left[ u_e^* - \frac{1}{2} (u_P^* + u_E^*) \right]
+ \frac{1}{2} \left\{ \frac{D U_e}{A_e} (p_P - p_E) - \frac{D U_P}{A_P} (p_w - p_e)_P + \frac{D U_E}{A_E} (p_w - p_e)_E \right\}, \tag{3.34}
\]

which is identical to that given by Majumdar [107]. However, expression (3.33) permits equation (3.34) to be generalized to variable \( \alpha \) without any difficulties.

A defect of equation (3.33) is that its extension to unsteady flow conditions is not obvious. The reason being that now \( A_P \) involves the time-dependent term \( (\rho \Delta x / \Delta t)_P \). Thus, the value of \( u_e \) may be influenced by the time step \( \Delta t \) (or CFL number). With this difficulty in mind, the extension to unsteady flow will take the following modification of

\[
H_e = \frac{1}{2} (H_P + H_E), \tag{3.35}
\]

with

\[
H_P = \frac{A_P u_P}{\alpha} - \left( \frac{1 - \alpha}{\alpha} \right) A_P u_P^* - \left( \frac{\rho (\rho \Delta x)}{\Delta t} \right)_P u_P^* - D U_P (p_w - p_e)_P, \tag{3.36}
\]

\[
H_E = \frac{A_E u_E}{\alpha} - \left( \frac{1 - \alpha}{\alpha} \right) A_E u_E^* - \left( \frac{\rho (\rho \Delta x)}{\Delta t} \right)_E u_E^* - D U_E (p_w - p_e)_E, \tag{3.37}
\]

\[
H_e = \frac{A_e u_e}{\alpha} - \left( \frac{1 - \alpha}{\alpha} \right) A_e u_e^* - \left( \frac{\rho (\rho \Delta x)}{\Delta t} \right)_e u_e^* - D U_e (p_w - p_e)_E, \tag{3.38}
\]
or, alternatively:

\[
\begin{align*}
\frac{u_e}{A_e} &= \frac{1}{2} \left( \frac{A_p}{A_e} u_p + \frac{A_E}{A_e} u_E \right) + \left( 1 - \alpha \right) \left[ \frac{u_e^*}{A_e} - \frac{1}{2} \left( \frac{A_p}{A_e} u_p^* + \frac{A_E}{A_e} u_E^* \right) \right] \\
&+ \frac{\alpha}{A_e} \left\{ \left( \frac{\rho^a \Delta x}{\Delta t} \right) u_e^0 - \frac{1}{2} \left( \left( \frac{\rho^a \Delta x}{\Delta t} \right) u_p^0 + \left( \frac{\rho^a \Delta x}{\Delta t} \right) u_E^0 \right) \right\} \\
&+ \frac{\alpha}{A_e} \left\{ \frac{DU_e}{A_e} (p_p - p_E) - \frac{1}{2} \left[ \frac{DU_p}{A_e} (p_w - p_e)_p + \frac{DU_E}{A_e} (p_w - p_e)_E \right] \right\} \\
&= \left( 1 - f_i^+ \right) \left\{ \frac{A_p}{A_e} \left[ u_p - (1 - \alpha) u_p^* \right] - \frac{\alpha}{A_e} \left( \frac{\rho^a J}{\Delta t} \right)_p u_p^0 - \frac{DU_p^E}{A_e} (p_w - p_e)_p \right\} \\
&+ f_i^+ \left\{ \frac{A_E}{A_e} \left[ u_E - (1 - \alpha) u_E^* \right] - \frac{\alpha}{A_e} \left( \frac{\rho^a J}{\Delta t} \right)_E u_E^0 - \frac{DU_E^E}{A_e} (p_w - p_e)_E \right\} \\
&+ \left( 1 - \alpha \right) u_e^* + \frac{\alpha}{A_e} \left( \frac{\rho^a J}{\Delta t} \right)_e u_e^0 + \frac{\alpha DU_e^E}{A_e} (p_p - p_E) \\
\end{align*}
\]

\[ (3.39) \]

Although the above derivation has been pursued for 1D uniform grid spacing, its extension to a 3D, general curvilinear coordinate system, allowing the evaluation of the contravariant velocities, say $U_e, V_n$ and $W_s$, is quite straightforward. The interpolation formulae for $u_e, u_n$ and $u_t$, which are ingredients of $U_e, V_n$ and $W_s$, are given in the following form (with related geometrical parameter defined in Figure 3.4):

\[
\begin{align*}
\frac{u_e}{A_e} &= \left( 1 - f_i^+ \right) \left\{ \frac{A_p}{A_e} \left[ u_p - (1 - \alpha) u_p^* \right] - \frac{\alpha}{A_e} \left( \frac{\rho^a J}{\Delta t} \right)_p u_p^0 - \frac{DU_p^E}{A_e} (p_w - p_e)_p \right\} \\
&+ f_i^+ \left\{ \frac{A_E}{A_e} \left[ u_E - (1 - \alpha) u_E^* \right] - \frac{\alpha}{A_e} \left( \frac{\rho^a J}{\Delta t} \right)_E u_E^0 - \frac{DU_E^E}{A_e} (p_w - p_e)_E \right\} \\
&+ \left( 1 - \alpha \right) u_e^* + \frac{\alpha}{A_e} \left( \frac{\rho^a J}{\Delta t} \right)_e u_e^0 + \frac{\alpha DU_e^E}{A_e} (p_p - p_E) \\
\end{align*}
\]

\[ (3.40) \]

\[
\begin{align*}
\frac{u_n}{A_n} &= \left( 1 - f_i^+ \right) \left\{ \frac{A_p}{A_n} \left[ u_p - (1 - \alpha) u_p^* \right] - \frac{\alpha}{A_n} \left( \frac{\rho^a J}{\Delta t} \right)_p u_p^0 - \frac{DU_p^n}{A_n} (p_s - p_n)_p \right\} \\
&+ f_i^+ \left\{ \frac{A_E}{A_n} \left[ u_E - (1 - \alpha) u_E^* \right] - \frac{\alpha}{A_n} \left( \frac{\rho^a J}{\Delta t} \right)_E u_E^0 - \frac{DU_E^n}{A_n} (p_s - p_E)_E \right\} \\
&+ \left( 1 - \alpha \right) u_n^* + \frac{\alpha}{A_n} \left( \frac{\rho^a J}{\Delta t} \right)_n u_n^0 + \frac{\alpha DU_n^n}{A_n} (p_p - p_N) \\
\end{align*}
\]

\[ (3.41) \]
\[ u_i = (1 - f_\zeta^+) \left\{ \frac{A_p}{A_t} [u_p - (1 - \alpha)u^*_p] - \alpha \frac{(\rho^0 J/\Delta t)_p}{A_t} u^*_p - \frac{D U^\zeta_p}{A_t} (p_b - p_i)_p \right\} \\
+ f_\zeta^+ \left\{ \frac{A_T}{A_t} [u_T - (1 - \alpha)u^*_T] - \alpha \frac{(\rho^0 J/\Delta t)_T}{A_t} u^*_T - \frac{D U^\zeta_T}{A_t} (p_b - p_i)_T \right\} \\
+ (1 - \alpha) u^*_i + \alpha \frac{(\rho^0 J/\Delta t)_i}{A_t} u^*_0 + \frac{\alpha D U^\zeta_i}{A_t} (p_p - p_i), \quad (3.42) \]

with

\[ f_\zeta^+ = \frac{P_e}{P_{\xi}}, \quad f_\eta^+ = \frac{P_n}{P_{\eta}}, \quad f_\zeta^+ = \frac{P_l}{P_{\zeta}}. \quad (3.43) \]

Similar expressions for \((v_\epsilon, v_\eta, v_\zeta)\) or \((w_\epsilon, w_\eta, w_\zeta)\) can be obtained by interchanging \((u \leftrightarrow v, DU \leftrightarrow DV)\) for the former, and \((u \leftrightarrow w, DU \leftrightarrow DW)\) for the latter. The coefficients of \(DU, DV\) and \(DW\) are defined below: All above considerations applied to \(U_\epsilon\) readily translate to other face velocities by appropriate and obvious replacements of nodal pressures, coefficients and geometric quantities.

A disadvantage of the above approach is that the fields \(u^*, v^*, w^*, U^*, V^*, W^*, U^0, V^0, W^0\) need to be stored. In practice, it is found that for \(\alpha \simeq 0.7\) the dependence of the basic Rhie & Chow scheme on \(\alpha\) is insignificant. Hence, there is an argument, on the grounds of storage economy, to use the basic scheme unless exceptional circumstances dictates otherwise.

The above considerations permit a stable pressure-correction algorithm to be constructed within the collocated framework. In the present work, the SIMPLE algorithm

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
has been preferred for its simplicity (although other variants, such as SIMPLEC, have been tested). The algorithm starts with a solution of the momentum equations, with an estimated pressure $p^*$, to yield approximate contravariant velocities $U^*, V^*$ and $W^*$. These obviously satisfy the momentum principle but not mass continuity. To achieve conservation of mass, velocity perturbations are added to the approximate values

\[ U = U^* + U', \quad V = V^* + V', \quad W = W^* + W', \]  

(3.44)
Using equations (3.40)–(3.42), the velocity corrections may be linked to neighboring pressure nodes via:

$$U'_e = \frac{\alpha J_e}{A_e} [(\xi_x)_e DU^5_e + (\xi_y)_e DV^5_e + (\xi_z)_e DW^5_e](p'_p - p'_E), \quad (3.45)$$

$$U'_w = \frac{\alpha J_w}{A_w} [(\xi_x)_w DU^5_w + (\xi_y)_w DV^5_w + (\xi_z)_w DW^5_w](p'_w - p'_P), \quad (3.46)$$

$$V'_n = \frac{\alpha J_n}{A_n} [(\eta_x)_n DU^5_n + (\eta_y)_n DV^5_n + (\eta_z)_n DW^5_n](p'_p - p'_N), \quad (3.47)$$

$$V'_s = \frac{\alpha J_s}{A_s} [(\eta_x)_s DU^5_s + (\eta_y)_s DV^5_s + (\eta_z)_s DW^5_s](p'_s - p'_P), \quad (3.48)$$

$$W'_t = \frac{\alpha J_t}{A_t} [(\xi_x)_t DU^5_t + (\xi_y)_t DV^5_t + (\xi_z)_t DW^5_t](p'_p - p'_T), \quad (3.49)$$

$$W'_b = \frac{\alpha J_b}{A_b} [(\xi_x)_b DU^5_b + (\xi_y)_b DV^5_b + (\xi_z)_b DW^5_b](p'_b - p'_P). \quad (3.50)$$

Substituting equation (3.44) in conjunction with equations (3.45)–(3.50) into the discretized continuity equation

$$\left[ \frac{(\rho - \rho^o)J}{\Delta t} \right]_p + (U_e - U_w) + (V_n - V_s) + (W_t - W_b) = 0 \quad (3.51)$$

yields:

$$A_P p'_p = \sum_{m=E,W,N,S,T,B} A_mp'_m + R_m, \quad (3.52)$$

where

$$A_E = \rho_e D_e, \quad A_W = \rho_w D_w, \quad A_N = \rho_n D_n, \quad A_S = \rho_s D_s, \quad A_T = \rho_t D_t, \quad A_B = \rho_b D_b,$$
CHAPTER 3. COMPUTATIONAL FLUID DYNAMICS

\[ A_P = \sum_{E,W,N,S,T,B} A_m, \]  

(3.53)

and the mass imbalance \( R_m \) is defined by

\[ R_m = \left[ \frac{(\rho - \rho^e)\Delta t}{\Delta t} \right]_p + (U^*_w - U^*_e) + (V^*_s - V^*_n) + (W^*_b - W^*_l) = 0. \]  

(3.54)

Then, new pressure fields are obtained via

\[ p = p^* + \alpha_p p', \]  

(3.55)

where \( \alpha_p \) is the under-relaxation factor for pressure correction \( p' \).

With the pressure-correction equation thus formulated, the usual sequence of operations may be performed in precisely the same manner as described by Patankar [122] for the staggered arrangement. Within a single-grid arrangement, the sequence consists of five major steps:

1. Calculate \( u^*, v^*, w^* \) by use of \( p^* \).
2. Evaluate \( R_m \) for all cells.
3. Solve equation (3.52) for \( p' \).
4. Improve \( u^*, v^*, w^* \) and \( p^* \) fields via equations (3.44) and (3.55).
5. Return to 1.

3.5 Second-Moment Closure

The numerical framework detailed in sections 3.2–3.4 would readily operate with eddy-viscosity models. However, such models have fundamental drawback in that they do not...
distinguish between the very different productions and transport rates of the individual Reynolds stresses. Instead, the stresses are determined in terms of a scalar parameter, leading to an isotropic exchange coefficient. This concept is satisfactory if the turbulence field is close to being isotropic or, alternatively, the flow behavior is dictated by a single stress, as is in the case in thin shear-layer flow, whose relationship to the eddy viscosity can be tuned. In complex flows, all stress gradients contribute significantly to the transport of mean-flow quantities, and failure to capture anisotropy and its effects on the shear stresses can lead to serious solution errors. Thus, it is more satisfactory to adopt a second-moment or Reynolds-stress closure model.

A Reynolds-stress transport model (RSTM) consists of a set of differential transport equations governing the distribution of related stresses. Each equation represents, in essence, a balance between stress transport, generation, destruction and redistribution. It is primarily the exact representation of stress generation, which varies greatly from stress to stress, that gives this type of closure the ability to return a realistic representation of stress anisotropy. Over the last two decades, a wide range of two and 3D flows, have been investigated (Launder [96], Leschziner [103]) in efforts to arrive at a reliable statement on the predictive performance, of second-moment closure. The overall conclusion of these studies has been that this form of model, while not being a panacea, is virtually always superior to eddy-viscosity variants.

The closure adopted herein Reynolds-stress components is based on that of Gibson & Launder [48]. Each is described by the following equation written for compactness in
CHAPTER 3. COMPUTATIONAL FLUID DYNAMICS

Cartesian tensor notation:

\[
\frac{D}{Dt}u_i u'_j = P_{ij} + G_{ij} + d_{ij} + \Phi_{ij} - \epsilon_{ij}
\]

(3.56)

where

\[
P_{ij} = - \left\{ \frac{u'_i u'_j}{\partial x_k} + \frac{u'_j u'_k}{\partial x_i} \right\}
\]

\[
G_{ij} = -2\Omega_k \left\{ u'_j u'_m \epsilon_{ikm} + \frac{u'_i u'_m}{\partial x_k} \epsilon_{jkm} \right\}
\]

(3.57)

\[
d_{ij} = \nu \frac{\partial^2 u'_j}{\partial x^2_k}
\]

(3.58)

\[
\Phi_{ij} = \frac{p'}{\rho} \left\{ \frac{\partial u'_i}{\partial x_j} + \frac{\partial u'_j}{\partial x_i} \right\}
\]

(3.59)

\[
\epsilon_{ij} = 2\nu \frac{\partial u'_i}{\partial x_k} \frac{\partial u'_j}{\partial x_k}
\]

(3.60)

\[
d_{ij} = -\frac{\partial}{\partial x_k} \left\{ \frac{u'_i u'_j}{\partial x_k} + \frac{p'}{\rho} \frac{u'_i}{\partial x_j} \delta_{jk} + \frac{p'}{\rho} \frac{u'_j}{\partial x_k} \delta_{ik} \right\}
\]

(3.61)

where, in (3.74), \(\epsilon_{ijk}\) is the alternating tensor.

The quantities \(P_{ij}\) and \(G_{ij}\) the stress generation associated with shear and rotation (like the very minor contribution from viscous diffusion, \(d_{ij}\) need no approximation comprising only Reynolds-stress components and mean field quantities. Since \(G_{kk}\) vanishes, there is no direct turbulence energy created by the rotation. If \(\Omega_k\) is taken as \((0,0,\Omega)\), then the positive \(y(x_2)\) direction points in the direction that the fluid is moving. Thus, for a mean flow that is radially outward \(\partial u/\partial y\) is positive - and hence \(u'v'\) negative - near the
pressure face. The rotational term in the $\overline{u'^2}$ equation, $-4\Omega\overline{u'v'}$, is thus positive, producing enhanced turbulent transport. The same rotational term with opposite sign appears in the $\overline{u'^2}$ equation, while in the shear-stress equation the rotational term is $-2\Omega(\overline{u'^2}-\overline{v'^2})$. In the usual thin-shear-flow situation $\overline{u'^2}$ exceeds $\overline{v'^2}$, so on the “pressure side” where $\partial u/\partial y$ is positive, the rotational and shear generation terms are of the same sign while on the “suction” side they oppose one another.

For the remaining processes in equation (3.73) surrogate models have to be provided before it can be used to determine the turbulent stress field. The turbulent stress diffusion process $(d_{ij})$ is approximated by the generalized-gradient-diffusion hypothesis of Daly & Harlow [34] while local isotropy is assumed for the dissipative correlations $(\varepsilon_{ij})$. Thus:

$$d_{ij} = \frac{\partial}{\partial x^i} \left\{ C_s \frac{k}{\varepsilon} \frac{\partial u'_i u'_j}{\partial x_k} \right\}$$

(3.62)

$$\varepsilon_{ij} = \frac{2}{3} \delta_{ij} \varepsilon$$

(3.63)

where $k$ and $\varepsilon$ are the turbulent kinetic energy and its dissipation rate, and $C_s = 0.22$. The model for the pressure-strain process is broadly taken over from Gibson & Launder [48]. The term is made up of contributions as follows:

$$\Phi_{ij} = \Phi_{ij1} + \Phi_{ij2} + \Phi_{ij3} + \Phi_{ijw}$$

(3.64)

the separate elements being associated respectively with purely turbulence interactions, mean strain, rotation and, finally, pressure reflections from the wall. Rotta’s [135] linear return-to-isotropy model is retained for the first of these:

$$\Phi_{ij1} = -C_1 \frac{\varepsilon}{k} \overline{u'_i u'_j} - \frac{1}{3} \delta_{ij} \overline{u'_k u'_k}$$

(3.65)
CHAPTER 3. COMPUTATIONAL FLUID DYNAMICS

while the analogous isotropization-of-production (IP) model is used for both the mean-strain and rotational parts of $\Phi_{ij}$, with $C_1 = 1.8$. Thus, for the mean-strain part:

$$\Phi_{ij2} = -C_2(P_{ij} - \frac{1}{3}\delta_{ij}P_{kk}) \quad (3.66)$$

A little care is needed in extrapolating this idea to the case of rotating flow essentially because the substantial derivative $Du_{ij}/Dt$ is not as it stands materially invariant. A frame-indifferent convective derivative may, however, be obtained in several ways. Two that suffice for present purposes are either to add $P_{ij}$ to each side of equation (3.5) (Eringen [44]):

$$C'_{ij} \equiv \left(\frac{D}{Dt}u_{ij}' \right) + P_{ij} = 2P_{ij} + G_{ij} + ... \quad (3.67)$$

or to assign half the rotation “generation” to the convection term (Takhar & Thomas [157]):

$$C''_{ij} \equiv \left\{ \frac{D}{Dt}u_{ij}' - \frac{1}{2}G_{ij} \right\} = P_{ij} + \frac{1}{2}G_{ij} + ... \quad (3.68)$$

Either approach suggests that in applying the isotropization-of-production idea to rotating systems the effective generation associated with rotation is only half as great relative shear generation as indicated by equation (3.5). One must of course adopt the same coefficient in the mean strain and rotational parts of $\Phi_{ij}$ so we must infer that

$$\Phi_{ij2} + \Phi_{ij3} = -C_2(P_{ij} + \frac{1}{2}G_{ij} - \frac{1}{3}\delta_{ij}P_{kk}) \quad (3.69)$$

since $G_{kk} = 0$. The value 0.6 is retained for $C_2$ since this satisfies Crow’s [33] exact result:

$$\Phi_{ij2} = 0.4k \left\{ \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right\} \quad (3.70)$$
in the limit of isotropic turbulence.

Here, following Shir [146] and Gibson & Launder [48], pressure reflections from the walls are modelled as:

\[
\Phi_{i,w} = \left\{ C_{1w} \left[ \frac{u_i' u_j'}{\nu} n_k n_m \delta_{ij} - \frac{3}{2} u_i' u_j' n_k n_j - \frac{3}{2} u_j' u_j' n_k n_i \right] + C_{2w} \left[ (\Phi_{i,km2} + \Phi_{i,km3}) n_k n_m \delta_{ij} - \frac{3}{2} (\Phi_{i,jk2} + \Phi_{i,jk3}) n_k n_j \right] \right\} f \left( \frac{k^{3/2}}{\varepsilon n_f n_f} \right) \tag{3.71}
\]

where \( n \) denotes the vector normal to the wall, and the model constants \( C_{1w} \) and \( C_{2w} \) taking values of 0.5 and 0.3, respectively.

Finally, the energy dissipation rate, \( \varepsilon \), appearing in various terms above is obtained from its own transport equation:

\[
\frac{D\varepsilon}{Dt} = C_{e1} \frac{P_{kk}}{k} - C_{e2} \frac{\varepsilon^2}{k} + C_{e3} \frac{\partial}{\partial x_k} \left( \frac{k}{\varepsilon} \frac{u_i' u_i'}{\partial x_k} \frac{\partial \varepsilon}{\partial x_k} \right). \tag{3.72}
\]

The standard values of the empirical coefficients \( C_{e1}, C_{e2} \) and \( C_{e} \) are 1.45, 1.90 and 0.18, respectively.

For simplicity, the above closure equations may be expressed in a 2D general coordinate as follows:

\[
\frac{1}{J} \left\{ \frac{\partial}{\partial \xi} \left[ \rho \frac{U}{J} (q_{11} \phi + q_{12} \phi^2) + \frac{\partial}{\partial \eta} \left[ \rho \frac{V}{J} (q_{12} \phi + q_{22} \phi^2) \right] \right] \right\} = \alpha_1 P_{11} + \alpha_2 P_{22} + \alpha_3 P_{12} + \alpha_4 P_k + \frac{\rho \varepsilon}{k} \left( \alpha_5 \overline{u'^2} + \alpha_6 \overline{v'^2} + \alpha_7 \overline{u'v'} \right) + \alpha_8 \rho \varepsilon, \tag{3.73}
\]
CHAPTER 3. COMPUTATIONAL FLUID DYNAMICS

\[ \epsilon^2_1 = (t_1, t_2) \]

\[ \epsilon^2_2 = (n_1, n_2) \]

Figure 3.5: Notation for a damping function on a curved surface.

where

\[ q_{11} = \frac{C_s \rho k}{\epsilon} \left( \bar{u}^2 y_\eta^2 - 2\bar{u} \bar{v} \bar{x}_\eta y_\eta + \bar{v}^2 x_\eta^2 \right), \]

(3.74)

\[ q_{22} = \frac{C_s \rho k}{\epsilon} \left( \bar{u}^2 y_\xi^2 - 2\bar{u} \bar{v} \bar{x}_\xi y_\xi + \bar{v}^2 x_\xi^2 \right), \]

(3.75)

\[ q_{12} = -\frac{C_s \rho k}{\epsilon} \left( \bar{u} \bar{v} \bar{x}_\eta \bar{y}_\eta - \bar{u} \bar{v} \bar{x}_\xi \bar{y}_\xi \right), \]

(3.76)

the contravariant velocities \( U \) and \( V \) are given by

\[ U = u y_\eta - v x_\eta, \quad V = v x_\xi - u y_\xi, \]

(3.77)

and the turbulence production \( P_{ij} \) arise as

\[ P_{11} = -\frac{2\rho}{f} \left[ \bar{u}^2 (u_\eta y_\eta - u_\eta y_\xi) + \bar{u} \bar{v} (u_\eta x_\eta - u_\xi x_\eta) \right], \]

(3.78)

\[ P_{22} = -\frac{2\rho}{f} \left[ \bar{v}^2 (v_\eta x_\eta - v_\xi x_\eta) + \bar{u} \bar{v} (v_\eta y_\eta - v_\xi y_\xi) \right], \]

(3.79)

\[ P_{12} = -\frac{\rho}{f} \left[ \bar{u} \bar{v} (v_\eta y_\eta - v_\eta y_\xi) + \bar{v}^2 (u_\xi x_\xi - u_\xi x_\eta) + \bar{u} \bar{v} (u_\eta y_\eta + v_\eta x_\eta - u_\xi x_\xi - v_\xi x_\eta) \right], \]

(3.80)
with $P_k = 0.5(P_{11} + P_{22})$. The coefficients $\alpha_i$ in the above expressions for individual stress components are summarized in Table 2.2. In this table, the wall-damping functions on any single curved wall, as shown in Figure 3.5, may be expressed by

$$f_x = n_1^2 f, \quad f_y = n_2^2 f, \quad f_{xy} = n_1 n_2 f,$$

(3.81)

where

$$f = \frac{k^{3/2}/\varepsilon}{C_{ll_n}},$$

with $l_n$ being the wall-normal distance.

3.6 Numerical Aspects of Reynolds-Stress Modeling

Past experience shows that the incorporation of second-moment closure into elliptic-flow solvers is a non-trivial task, mainly because the manner in which the stresses interact with the strain field is not readily implemented in a numerically stable form. In complex geometries, apart from the problems associated with non-orthogonality - particularly at boundaries, the main difficulty in combining a stress model with a collocated finite-volume scheme arises from the fact that storage of all variables at the same spatial location tends to lead to chequerboard oscillations, caused by an inappropriate decoupling of velocity and Reynolds stresses when linear interpolation is used to approximate cell-face stresses in terms of nodal values. The algorithm outlined below re-establishes coupling by use of an interpolation practice similar to that adopted by Rhie & Chow [131] and Obi et al. [124].
To highlight the underlying rationale, it is instructive to focus first on the Boussinesq relationship:

\[- \rho \overline{u'v'} = \mu_f \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \delta_{ij} \rho k. \quad (3.82)\]

As is evident, \( \overline{u'^2} \) is "driven" by \( \partial u/\partial x \), while \( \overline{u'v'} \) is "driven" by \( \partial u/\partial y \) and \( \partial v/\partial x \).

To retain this physical coupling in the numerical representation, it is necessary to store \( \overline{u'^2} \) between \( u \)-velocity locations approximating \( \partial u/\partial x \). Similarly, \( \overline{u'v'} \) needs to be stored between \( u \) and \( v \)-nodes approximating \( \partial u/\partial y \) and \( \partial v/\partial x \), respectively. This rationale is reflected by the arrangement in Figure 3.6. In a 3D environment, a total of seven separate control volumes are thus required if the stress model is used. Examples of flows computed with this arrangement may be found in Iacovides & Launder [70], Lin & Leschziner [104] and Sajjadi et al. [139].

The above methodology is, of course, untenable in a collocated arrangement. To retain this arrangement, a nonlinear interpolation practice has been devised which prevents stress-related odd-even oscillations. In order to facilitate transparency, the method is ex-
plained first by reference to a 2D Cartesian arrangement having a uniform mesh $\Delta x = \Delta y$, with a generalization pursued later. With the Reynolds-stress model of Gibson & Launder \cite{48} chosen to represent turbulence transport, it may be shown that the discretized form of the transport equation for the normal stress $\overline{u'^2}$ at the location $P$ may be written:

$$
\overline{u'^2}_P = \sum_{m=E,W,N,S} A_m \overline{u'^2}_m + S'_C / A_P + \mu_{11}^P \frac{(u_w - u_e)_P}{\Delta x}, \quad (3.83)
$$

where

$$
\mu_{11}^P = \frac{|(2 - \frac{4}{3}C_2 + \frac{2}{3}C_2 C_2 f_x + \frac{2}{3}C_2 C_2 f_y)\rho u'^2| \rho \Delta x \Delta y}{A_P}, \quad (3.84)
$$

and $S'_C$ includes cross-diffusion term arising from Daly & Harlow's stress diffusion model \cite{34} and some fragments of the production, pressure-strain and dissipation processes.

Analogous expressions for $\overline{u'^2}_E$ and $\overline{u'^2}_e$ are:

$$
\overline{u'^2}_E = \frac{H_E}{A_E} + \mu_{11}^E \frac{(u_w - u_e)_E}{\Delta x}, \quad \overline{u'^2}_e = \frac{H_e}{A_e} + \mu_{11}^E \frac{(u_P - u_E)}{\Delta x}, \quad (3.85)
$$
where $\mu_{11}^t = (\mu_{11}^p + \mu_{11}^E)/2$. If condition equation (3.33) is applied, the resulting form of $\overline{u^2}^e$ is:

$$
\overline{u^2}^e = \frac{1}{2}(\overline{u^2}^p + \overline{u^2}^E) \quad \text{linear interpolation}
$$

$$
+ \frac{1}{2} \left[ (\mu_{11}^p + \mu_{11}^E)(u_p - u_E) - \mu_{11}^p (u_w - u_e)p - \mu_{11}^E (u_w - u_e)E \right],
$$

(3.86)

which is identical to that proposed by Obi et al. [124]. The above practice of extracting an apparent viscosity (e.g. $\mu_{11}$ is associated with the mean strain $\partial u/\partial x$) has already been suggested earlier by Huang & Leschziner [68] who employed the staggered Reynolds-stress arrangement. Their objective was to enhance the iterative stability by increasing the magnitude of the diagonal coefficient $A_p$. If a time-marching scheme is employed instead, stability is strongly promoted by using a small time-step, and the inclusion of apparent viscosities is often of little benefit in an unsteady state. Thus, in order not to restrict the time-step the apparent vorticities is simply replaced with the turbulent viscosity $\mu_t$. In contrast, it may be seen from equation (3.86) that the same apparent-viscosity approach applied to the collocated arrangement for all Reynolds stresses introduces fourth-order smoothing, here depending on the $u$-velocity component rather than, as was the case earlier, on pressure. Note that equation (3.86) has been derived for steady-state conditions without under-relaxing the solution sequence. To generalize this formula to unsteady conditions and to include under-relaxation, equation (3.86) may be further modified, in
accordance with equations (3.35)–(3.39), as follows:

\[
\overline{u^2_e} = \frac{1}{2} \left( \frac{A_p}{A_e} \overline{u^2_p} + \frac{A_E}{A_e} \overline{u^2_E} \right) + (1 - \alpha) \left[ \overline{u^2_e} - \frac{1}{2} \left( \frac{A_p}{A_e} \overline{u^2_p} + \frac{A_E}{A_e} \overline{u^2_E} \right) \right] \\
+ \alpha \left\{ \left( \frac{p_s \Delta x \Delta y}{u^2_e} \right) \frac{\overline{u^2_p}}{u^2_e} - \frac{1}{2} \left( \left( \frac{p_s \Delta x \Delta y}{u^2_p} \right) \frac{\overline{u^2_p}}{u^2_p} + \left( \frac{p_s \Delta x \Delta y}{u^2_E} \right) \frac{\overline{u^2_E}}{u^2_E} \right) \right\},
\]

(3.87)

where

\[
\mu_{11}^p = \frac{\left( 2 - \frac{4}{3} C_2 + \frac{8}{3} C_2 C_2w f_x + \frac{2}{3} C_2 C_2w f_y \right) \rho \overline{u^2_p} \Delta x \Delta y}{A_e}, \\
\mu_{11}^E = \frac{\left( 2 - \frac{4}{3} C_2 + \frac{8}{3} C_2 C_2w f_x + \frac{2}{3} C_2 C_2w f_y \right) \rho \overline{u^2_E} \Delta x \Delta y}{A_e},
\]

\[
\mu_{11}^{P} = \frac{1}{2} (\mu_{11}^p + \mu_{11}^E), \quad (3.88)
\]

with \( \mu_P \) replacing \( \mu_{11} \) in an unsteady state case.

In deriving equations (3.87) and (3.88), an element of uncertainty is that the level of apparent viscosity is a strong function of the manner in which the source in the \( \overline{u^2} \) is transformed into the equivalent quasi-linear form via

\[
S_{\overline{u^2}} \leftarrow S_C + S_P \overline{u^2} \rho.
\]

(3.89)

To appreciate the nature of the problem, it must first be pointed out that \( \mu_{11} \) depends strongly on \( A_P \) via equation (3.84). But \( A_P \) contains \( S_P \) and this value depends on how the terms encountered in \( S_{\overline{u^2}} \) are treated. It is a general objective to maximize the magnitude of the (negative !) fragment \( S_P \). If any term in \( S_{\overline{u^2}} \) is found to be negative but does not contain \( \overline{u^2} \) as a factor, it may nevertheless be allocated to \( S_P \) by dividing that term by \( \overline{u^2} \).
CHAPTER 3. COMPUTATIONAL FLUID DYNAMICS

at the previous iterate, and then adding the result to $S_p$. Because $S_p$ varies greatly from node to node, the averaging process equation (3.87) may yield a value for $A_e$ which, if fed into equation (3.88), will give an inappropriate level of $\mu_{11}^p$. Thus, it would be desirable to construct a scheme which allows $\mu_{11}$ (and other apparent viscosities) to be determined without reference to the discretization process and its details. Focusing attention to the differential equation governing $\bar{u}^2$, it may be written in the following compressed form:

$$C_{11} - D_{11} = -A - B + \left(P_{11} + \Phi_{11} - \frac{2}{3}\rho e + A + B\right), \quad (3.90)$$

with

$$A = \frac{\rho e}{k} \left( C_1 + 2C_{1w}f_x \bar{u}^2 \right), \quad (3.91)$$

and

$$B = \left(2 - \frac{4}{3}C_2 + \frac{8}{3}C_2C_{2w}f_x + \frac{2}{3}C_2C_{2w}f_y\right) \frac{\rho \bar{u}^2 \partial u}{\partial x}, \quad (3.92)$$

Combination of equations (3.90), (3.91) and (3.92) then yields:

$$-\rho \bar{u}^2 \frac{\partial u}{\partial x} = \mu_{11} \frac{\partial u}{\partial x} + \frac{k}{\varepsilon} \left( C_{11} - D_{11} - (P_{11} + \Phi_{11} - \frac{2}{3}\rho e + A + B) \right), \quad (3.93)$$

with

$$\mu_{11} = \frac{2 - \frac{4}{3}C_2 + \frac{8}{3}C_2C_{2w}(4f_x + f_y)}{C_1 + 2C_{1w}f_x} \frac{\rho \bar{u}^2}{\varepsilon}. \quad (3.94)$$

Note that $\mu_{11}$ in equations (3.93)-(3.94) is associated with the differential gradient; in contrast, $\mu_{11}^p$ in equation (3.84) is applicable to the gradient approximation at grid-node
The derivation of \( \mu_{22} \) follows a path analogous to equations (3.90)-(3.94). Hence, only the end result is given below:

\[
\mu_{22} = \frac{2 - \frac{4}{5}C_2 + \frac{4}{5}C_2C_{2w}(4f_y + f_x) \rho k \nu^2}{C_1 + 2C_{1w}f_y}.
\] (3.95)

Attention is now focused to the interpolation formula for the shear stress \( \overline{u'v'} \). A treatment consistent with the one applied in relation \( \overline{u'^2} \) would involve extracting a viscosity \( \mu_{12} \) by reference to \( \partial u/\partial y \) and \( \partial v/\partial x \). This is not possible, however, because the fragments multiplying \( \partial u/\partial y \) and \( \partial v/\partial x \) in the stress model are not identical. Hence, here, \( \overline{u'v'} \) is only sensitized to one of the two strains; which one is chosen is dictated by the direction of the derivative of the shear stress. Since in the \( u \)-momentum equation the shear-stress gradient is \( \partial \overline{u'v'}/\partial y \), it is natural to relate \( \rho \overline{u'v'} \) to \( \partial u/\partial y \), leading to the stability-promoting second-order derivative \( \frac{\partial}{\partial y}(\mu_{12} \frac{\partial u}{\partial y}) \). The same rule can be applied to the \( v \)-momentum equation, resulting in the diffusion term \( \frac{\partial}{\partial x}(\mu_{21} \frac{\partial v}{\partial x}) \). In order to derive the two apparent viscosities \( \mu_{12} \) and \( \mu_{21} \) from the \( \overline{u'v'} \)-equation, this equation is first written as follows:

\[
C_{12} - D_{12} = -\frac{d^2}{\partial y^2} \left[ C_1 + \frac{3}{2}C_{1w}(f_x + f_y) \right] \overline{u'v'}
- \left[ 1 - C_2 + \frac{3}{2}C_2C_{2w}(f_x + f_y) \right] \left( \nu^2 \frac{\partial u}{\partial y} + \nu^2 \frac{\partial v}{\partial x} \right).
\] (3.96)

Then equation (3.96) can be reformulated in either of the two forms:

\[
-\rho \overline{u'v'} = \mu_{12} \frac{\partial u}{\partial y} + \frac{k C_{12} - D_{12} + [1 - C_2 + \frac{3}{2}(f_x + f_y)] \rho \nu^2 \frac{\partial v}{\partial x}}{C_1 + \frac{3}{2}C_{1w}(f_x + f_y)},
\] (3.97)

or

\[
-\rho \overline{u'v'} = \mu_{21} \frac{\partial v}{\partial x} + \frac{k C_{12} - D_{12} + [1 - C_2 + \frac{3}{2}(f_x + f_y)] \rho \nu^2 \frac{\partial u}{\partial y}}{C_1 + \frac{3}{2}C_{1w}(f_x + f_y)},
\] (3.98)
where

\[
\begin{align*}
\mu_{12} &= \frac{1 - C_2 + \frac{3}{2}C_2C_{2w}(f_x + f_y)}{C_1 + \frac{3}{2}C_{1w}(f_x + f_y)} \left( \frac{\rho k^{7/2}}{\varepsilon} \right), \\
\mu_{21} &= \frac{1 - C_2 + \frac{3}{2}C_2C_{2w}(f_x + f_y)}{C_1 + \frac{3}{2}C_{1w}(f_x + f_y)} \left( \frac{\rho k^{7/2}}{\varepsilon} \right),
\end{align*}
\]

(3.99) (3.100)

To extend the above concepts to the general curvilinear environment, attention is next focused on the \( u \) and \( v \) momentum equations written in terms of the general coordinates \((\xi, \eta)\):

\[
\begin{align*}
\frac{\partial}{\partial \xi} (U\rho u) + \frac{\partial}{\partial \eta} (V\rho u) = & \quad -\frac{\partial}{\partial \xi} (p + \rho \overline{u^2})\eta + \frac{\partial}{\partial \eta} (p + \rho \overline{u'^2})\eta \xi + \frac{\partial}{\partial \xi} (\rho \overline{u'v'})\eta - \frac{\partial}{\partial \eta} (\rho \overline{u'v'})\eta \xi, \\
\frac{\partial}{\partial \xi} (U\rho v) + \frac{\partial}{\partial \eta} (V\rho v) = & \quad \frac{\partial}{\partial \xi} (p + \rho \overline{v^2})\eta \xi - \frac{\partial}{\partial \eta} (p + \rho \overline{v'^2})\eta \xi - \frac{\partial}{\partial \xi} (\rho \overline{u'v'})\eta + \frac{\partial}{\partial \eta} (\rho \overline{u'v'})\eta \xi,
\end{align*}
\]

(3.101) (3.102)

It is clear from equations (3.101)–(3.102) that no physical (second-order) diffusion terms arise naturally. In order to extract apparent viscosities from the Reynolds-stress equations in terms of the \((\xi, \eta)\) coordinate system, a tedious, but otherwise rather straightforward, manipulation of the transformed equations, analogous to that in the Cartesian framework, may be carried out. Interestingly enough, the final expressions are identical to equations (3.94), (3.95), (3.99) and (3.100), except that the wall-related damping function in the pressure-strain model assumes different forms. Thus, for a single \( x \)-directed wall in the Cartesian framework, \( f_x \) is given by \( f_x = \frac{k^{5/2}/\varepsilon}{C_{1w}} \); while, in contrast, \( f_x \) along a curved
surface becomes \( f_y = n_2^2 \left( \frac{v^{y^2}}{C_i n_s} \right) \) (recall equation (3.81)). Introduction of the apparent viscosity into equations (3.101)-(3.102) leads to

\[
\frac{\partial}{\partial \xi} \left[ U \rho \phi - \left( \frac{q^\phi_1}{J} \right) \frac{\partial \phi}{\partial \xi} \right] + \frac{\partial}{\partial \eta} \left[ V \rho \phi - \left( \frac{q^\phi_2}{J} \right) \frac{\partial \phi}{\partial \eta} \right] = J S_\phi, \tag{3.103}
\]

For \( \phi = u \):

\[
q^u_1 = \mu_1 y^2_\eta + \mu_2 x^2_\eta, \quad q^u_2 = \mu_1 y^2_\xi + \mu_2 x^2_\xi,
\]

\[
J S_u = -\frac{\partial}{\partial \xi} \left( \rho \bar{u} \bar{v} \right) x_\eta + \frac{\partial}{\partial \eta} \left( \rho \bar{u} \bar{v} \right) y_\xi + \frac{\partial}{\partial \xi} \left( \rho \bar{u} \bar{v} \right) y_\eta - \frac{\partial}{\partial \eta} \left( \rho \bar{u} \bar{v} \right) x_\eta, \tag{3.104}
\]

where

\[
p^{u\xi} = p + \rho \bar{u} \bar{v} - \left( \frac{\mu_1 y_\eta}{J} \right) \frac{\partial u}{\partial \xi}, \quad p^{u\eta} = p + \rho \bar{u} \bar{v} - \left( \frac{\mu_1 y_\xi}{J} \right) \frac{\partial u}{\partial \eta}, \tag{3.105}
\]

\[
\bar{u} \bar{v} = \bar{u} \bar{v} - \left( \frac{\mu_1 y_\eta}{J} \right) \frac{\partial u}{\partial \xi}, \quad \bar{u} \bar{v} = \bar{u} \bar{v} - \left( \frac{\mu_1 y_\xi}{J} \right) \frac{\partial u}{\partial \eta}. \tag{3.106}
\]

For \( \phi = v \):

\[
q^v_1 = \mu_2 y^2_\eta + \mu_2 x^2_\eta, \quad q^v_2 = \mu_2 y^2_\xi + \mu_2 x^2_\xi,
\]

\[
J S_v = -\frac{\partial}{\partial \xi} \left( \rho \bar{u} \bar{v} \right) x_\eta - \frac{\partial}{\partial \eta} \left( \rho \bar{u} \bar{v} \right) y_\xi - \frac{\partial}{\partial \xi} \left( \rho \bar{u} \bar{v} \right) y_\eta + \frac{\partial}{\partial \eta} \left( \rho \bar{u} \bar{v} \right) y_\eta, \tag{3.107}
\]

where

\[
p^{v\xi} = p + \rho \bar{u} \bar{v} - \left( \frac{\mu_2 y_\eta}{J} \right) \frac{\partial v}{\partial \xi}, \quad p^{v\eta} = p + \rho \bar{u} \bar{v} + \left( \frac{\mu_2 y_\xi}{J} \right) \frac{\partial v}{\partial \eta}, \tag{3.108}
\]

\[
\bar{u} \bar{v} = \bar{u} \bar{v} + \left( \frac{\mu_2 y_\eta}{J} \right) \frac{\partial v}{\partial \xi}, \quad \bar{u} \bar{v} = \bar{u} \bar{v} - \left( \frac{\mu_2 y_\xi}{J} \right) \frac{\partial v}{\partial \eta}. \tag{3.109}
\]
With pressures now augmented by the normal-stress contributions via equations (3.105) and (3.108), the cell-face velocities, previously given by equations (3.40) and (3.41), are now modified as follows:

\[
\begin{align*}
    u_e &= (1 - f^-_e) \left\{ \frac{A_p}{A_e} [u_p - (1 - \alpha) u_p^*] - \alpha \frac{(\rho^0/\Delta t)p}{A_e} u_p^* - \frac{D U_p^e}{A_e} (p_w - p_e) E P \right\} \\
    &+ f^-_e \left\{ \frac{A_E}{A_e} [u_E - (1 - \alpha) u_E^*] - \alpha \frac{(\rho^0/\Delta t)E}{A_e} u_E^* - \frac{D U_E^e}{A_e} (p_w - p_e) E E \right\} \\
    &+ (1 - \alpha) u_e^* + \alpha \frac{(\rho^0/\Delta t)E}{A_e} u_E^* + \frac{\alpha D U_E^e}{A_e} (p_P - p_E) E E,
    \tag{3.110}
\end{align*}
\]

\[
\begin{align*}
    u_n &= (1 - f^-_n) \left\{ \frac{A_p}{A_n} [u_P - (1 - \alpha) u_P^*] - \alpha \frac{(\rho^0/\Delta t)p}{A_n} u_P^* - \frac{D U_P^n}{A_n} (p_s - p_n) P n \right\} \\
    &+ f^-_n \left\{ \frac{A_N}{A_n} [u_N - (1 - \alpha) u_N^*] - \alpha \frac{(\rho^0/\Delta t)N}{A_n} u_N^* - \frac{D U_N^n}{A_n} (p_s - p_n) N n \right\} \\
    &+ (1 - \alpha) u_n^* + \alpha \frac{(\rho^0/\Delta t)N}{A_n} u_N^* + \frac{\alpha D U_N^n}{A_n} (p_P - p_N) N n,
    \tag{3.111}
\end{align*}
\]

Similarly the velocities \( v_e \) and \( v_n \) become:

\[
\begin{align*}
    v_e &= (1 - f^-_e) \left\{ \frac{A_p}{A_e} [v_p - (1 - \alpha) v_p^*] - \alpha \frac{(\rho^0/\Delta t)p}{A_e} v_p^* - \frac{D V_p^e}{A_e} (p_w - p_e) E P \right\} \\
    &+ f^-_e \left\{ \frac{A_E}{A_e} [v_E - (1 - \alpha) v_E^*] - \alpha \frac{(\rho^0/\Delta t)E}{A_e} v_E^* - \frac{D V_E^e}{A_e} (p_w - p_e) E E \right\} \\
    &+ (1 - \alpha) v_e^* + \alpha \frac{(\rho^0/\Delta t)E}{A_e} v_E^* + \frac{\alpha D V_E^e}{A_e} (p_P - p_E) E E,
    \tag{3.112}
\end{align*}
\]

\[
\begin{align*}
    v_n &= (1 - f^-_n) \left\{ \frac{A_p}{A_n} [v_P - (1 - \alpha) v_P^*] - \alpha \frac{(\rho^0/\Delta t)p}{A_n} v_P^* - \frac{D V_P^n}{A_n} (p_s - p_n) P n \right\} \\
    &+ f^-_n \left\{ \frac{A_N}{A_n} [v_N - (1 - \alpha) v_N^*] - \alpha \frac{(\rho^0/\Delta t)N}{A_n} v_N^* - \frac{D V_N^n}{A_n} (p_s - p_n) N n \right\} \\
    &+ (1 - \alpha) v_n^* + \alpha \frac{(\rho^0/\Delta t)N}{A_n} v_N^* + \frac{\alpha D V_N^n}{A_n} (p_P - p_N) N n,
    \tag{3.113}
\end{align*}
\]
where

\[ DU^\xi = y_\eta, \quad DU^\eta = -y_\xi, \quad DV^\xi = -x_\eta, \quad DV^\eta = x_\xi. \quad (3.114) \]

The relationship between velocity- and pressure-corrections is represented by:

\[ u'_e = \frac{\alpha DU^\xi}{A_e} (p'_p - p'_E)^u_\xi, \quad u'_n = \frac{\alpha DU^\eta}{A_e} (p'_p - p'_N)^u_\eta, \]

\[ v'_e = \frac{\alpha DV^\xi}{A_e} (p'_p - p'_E)^v_\xi, \quad v'_n = \frac{\alpha DV^\eta}{A_e} (p'_p - p'_N)^v_\eta. \quad (3.115) \]

A disadvantage of expressions (3.115) is that their insertion into the continuity-correction equation does not lead to an equation in terms of the isotropic pressure correction \( p' \). However, because all terms in the \( p' \)-equation eventually decline to zero, the fragments giving rise to the anisotropic pressures in equation (3.115) may be omitted without ultimate consequence, and the usual \( p' \)-equation is recovered, which means that velocity perturbations are held to be unrelated to normal-stress perturbations. This would appear to suggest that the definition of \( p'^\xi \), \( p'^\eta \), \( p'^\xi \), and \( p'^\eta \) serve no useful purpose. This is not so, however, since all fragments in these modified pressures are subjected to the Rhie & Chow interpolation process. Hence, the fourth-order diffusion introduced into equations (3.110)–(3.113) arises not only from pressure but from the normal-stress-related fragments.

Finally, it should be mentioned that the source-term linearization equation (3.89) and under-relaxation factor \( \alpha \) are important ingredients for stability-promoting measures when the stress closure is used. The former requires a careful local discrimination between positive and negative fragments with the latter being allocated to \( S_p \). As regards
the under-relaxation factor, experience has shown that the value required in the stress equations is of order 0.3.

3.7 Wall-Reflection Terms

The combined model of Shir and Gibson & Launder, introduced in the previous section were said to have been derived so as to be coordinate invariant. It should, therefore, follow that they can be applied to any wall-oriented system such as $(\xi, \eta)$ system in Figure 3.6. This is an issue which needs careful examination. In terms of the transformed $(\xi, \eta)$ system in Figure 3.6, $\Phi_{ij}^{wall}$ be may expanded as follows:

\[
\Phi_{11}^{wall} = C_{1w} \frac{\rho E}{k} \left( -2u'^2 f_x + v'^2 f_y - u'v' f_{xy} \right) + C_{2w} \left( -2\Phi_{112} f_x + \Phi_{222} f_y - \Phi_{122} f_{xy} \right),
\]

\[
\Phi_{22}^{wall} = C_{1w} \frac{\rho E}{k} \left( -2u'^2 f_x - v'^2 f_y - u'v' f_{xy} \right) + C_{2w} \left( \Phi_{112} f_x - 2\Phi_{222} f_y - \Phi_{122} f_{xy} \right),
\]

\[
\Phi_{12}^{wall} = -1.5C_{1w} \frac{\rho E}{k} \left[ (u'^2 + v'^2) f_{xy} + u'v'(f_x + f_y) \right] -1.5C_{2w} \left[ (\Phi_{112} + \Phi_{222}) f_{xy} + \Phi_{122} (f_x + f_y) \right].
\]

where

\[
f_x = n_1^2 f_\eta, \quad f_y = n_2^2 f_\eta, \quad f_{xy} = n_1 n_2 f_\eta,
\]

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
The stresses in Figure 3.5. are Cartesian, i.e. associated with the system \((e_1, e_2)\). If the above forms are valid, then it should be possible to express \(\Phi_{ij}^{wall}\) in terms of the local \((e'_1, e'_2)\) system, and upon transforming the wall-oriented stresses to their Cartesian counterparts, the forms of equations (3.116)-(3.118) should be recoverable. To investigate this correspondence, it is useful to first review transformation operations allowing the stress in one system to be derived from those in the other system.

Consider next an infinitesimal tetrahedron in Cartesian coordinates as shown in Figure 3.7, which is under the action of a surface force. The surface force imposed on a small element of area in a continuous body depends not only on the magnitude of the area but...
also on its orientation. Denoting the stress vector by $t^{(n)}$, where $n$ identifies the orientation of that area, and utilizing Newton’s second law, the state in Figure 3.7 can be described by the following equation:

$$t^{(n)} \Delta S - t^{(i)} \Delta S_i = \text{body force} \quad (3.121)$$

Shrinking the tetrahedron to a point (in which case the body force can be neglected) together with $\Delta S_l = \Delta S n. e_i$ from Green theorem, one obtains:

$$t^{(n)} = (n. e_i) t^{(i)} = n. \left( \varepsilon_i \otimes t^{(i)} \right) \quad \text{stress dyadic} \quad (3.122)$$

Attention is here drawn to the fact that although the stress vector $t^{(n)}$ depends on the unit normal $n$, the stress dyadic at a point is invariant. Decomposition of vector $t^{(i)}$ into components along the $x_j$-direction:

$$t^{(i)} = \tau_{ij} e_j, \quad (3.123)$$

allows the stress dyadic to be expressed in terms of two different orthogonal coordinates:

$$\tau_{ij} e_i \otimes e_j = \tau'_{ij} e'_i \otimes e'_j. \quad (3.124)$$

Suppose two set of base vectors $e_i$ and $e_j$ are related to each other by:

$$e'_i = g_{ij} e_j, \quad (3.125)$$

where $g_{ij}$ denotes the direction cosines, i.e. $g_{ij} = e_i \cdot e_j$. Then equation (3.124) can be reduced to:

$$(\tau_{ij} - g_{ik} \tau'_{kl} g_{lj}) e_i \otimes e_j = 0,$$
which implies

$$\tau = \overline{G}^T \tau' \overline{G}. \quad (3.126)$$

Similarly, it can be shown that

$$\tau' = \overline{G} \tau \overline{G}^T. \quad (3.127)$$

Let $e_1 = (t_1, t_2)$ and $e_2 = (n_1, n_2)$, then the matrix $G$ corresponding to Figure 3.5 is expressed as:

$$G = \begin{pmatrix} t_1 & t_2 \\ n_1 & n_2 \end{pmatrix}, \quad (3.128)$$

and the stress tensor in the Cartesian system, $\tau_{ij}$, can be represented relative to that in the transformed domain, $\tau'_{ij}$, by:

$$\begin{pmatrix} \tau_{11} & \tau_{12} \\ \tau_{12} & \tau_{22} \end{pmatrix} = \begin{pmatrix} t_1^2 \tau'_{11} + 2t_1n_1 \tau'_{12} + n_1^2 \tau'_{22} & t_1t_2 \tau'_{11} + (t_1n_2 + t_2n_1) \tau'_{12} + n_1n_2 \tau'_{22} \\ t_1t_2 \tau'_{11} + (t_1n_2 + t_2n_1) \tau'_{12} + n_1n_2 \tau'_{22} & t_2^2 \tau'_{11} + 2t_2n_2 \tau'_{12} + n_2^2 \tau'_{22} \end{pmatrix} \quad (3.129)$$

Since $\Phi_{ij1}^{wall}$ and $\Phi_{ij2}^{wall}$ are linear in stresses, (3.129) is applicable. Taking $\Phi_{ij1}^{wall}$ as an example, $\Phi_{ij1}^{wall}$ to $\Phi_{ij1}^{wall}$ can be related by

$$\begin{pmatrix} \Phi_{111} & \Phi_{121} \\ \Phi_{121} & \Phi_{221} \end{pmatrix}^{wall} = \begin{pmatrix} t_1^2 \Phi_{111}' + 2t_1n_1 \Phi_{121}' + n_1^2 \Phi_{221}' & t_1t_2 \Phi_{111}' + (t_1n_2 + t_2n_1) \Phi_{121}' + n_1n_2 \Phi_{221}' \\ t_1t_2 \Phi_{111}' + (t_1n_2 + t_2n_1) \Phi_{121}' + n_1n_2 \Phi_{221}' & t_2^2 \Phi_{111}' + 2t_2n_2 \Phi_{121}' + n_2^2 \Phi_{221}' \end{pmatrix}^{wall}$$
For the case $i = j = 1$, $\Phi_{111}^{\text{wall}}$ can be expanded in the form of:

$$\Phi_{111}^{\text{wall}} = C_{1w} \frac{\rho e}{k} \left[ (t_1^2 - 2n_1^2)(\bar{v}^2)' - 3t_1n_1(\bar{u}'v') \right] f_\eta$$  \hspace{1cm} (3.131)

In recognition of $(t_1, t_2) \equiv (n_2, -n_1)$ from orthogonality, $(\bar{v}^2)'$ and $(\bar{u}'v')'$ in the above equation can be written by employing (3.127) as:

$$(\bar{v}^2)' = n_1^2\bar{u}^2 + 2n_1n_2\bar{u}'\bar{v} + n_2^2\bar{v}^2,$$ \hspace{1cm} (3.132)

$$(\bar{u}'v')' = n_1n_2(\bar{u}^2 - \bar{v}^2) + (n_2^2 - n_1^2)\bar{u}'\bar{v}.$$

Substituting (3.132) and (3.133) into (3.131) in conjunction with the requirement $n_1^2 + n_2^2 = 1$ for a unit vector, (3.131) is further simplified to:

$$\Phi_{111}^{\text{wall}} = C_{1w} \frac{\rho e}{k} \left[ -2n_1^2\bar{u}^2 + n_2^2\bar{v}^2 - n_1n_2\bar{u}'\bar{v} \right] f_\eta,$$ \hspace{1cm} (3.134)

which is exactly the same as equation (3.116). Similar proofs can be performed for $\Phi_{221}^{\text{wall}}$ and $\Phi_{121}^{\text{wall}}$ without any difficulty. Analogous arguments apply to $\Phi_{ij2}^{\text{wall}}$ because the IPM version used herein relates $\Phi_{ij2}$ to the stresses in a linear manner. Hence, Gibson & Launder's model, when expressed by forms corresponding to equations (3.116)–(3.118), applies to curved walls.

3.8 Boundary Conditions

Boundary conditions considered herein are mainly for incompressible flow having elliptic characteristics. These include inlet, wall, plane of symmetry, and exit.
3.8.1 Inlet

At an inlet, all properties governed by related transport equations must be prescribed, i.e. Dirichlet conditions are applied. Ideally, all conditions should be available from experimental measurements or exact analytical considerations. In practice, this is virtually never possible, and assumptions must be made for certain properties, based on intuition or order-of-magnitude considerations. Typically, the velocity normal to the inlet is known, while the lateral velocity is assumed to be zero. With velocity prescribed, no conditions are needed for the pressure. However, with regard to turbulence quantities, it is rarely the case that data are available for more than one turbulence intensity or turbulence energy. In such circumstances, a sensible approach is, for example, to apply

\[
\frac{\bar{u}_j^2}{u_k^2} = \frac{2}{3}k \quad \text{no summation implied,}
\]

\[
\bar{u}_i\bar{u}_j = 0 \quad \text{for } i \neq j. \quad (3.135)
\]

Since the rate of turbulence dissipation is virtually never available from experiment, a standard practice is to refer to the relationship between \( \varepsilon \) and the mixing length, applicable to energy-equilibrium conditions.

3.8.2 Wall

Walls pose particular challenges in the context of turbulence-flow computations, because spatial variations of the near-wall turbulence structure are intense due to the combined influence of viscosity and wall-induced anisotropy. When low-Reynolds-number models
are applied, numerical integration encompasses the entire near-wall region including the
viscous sublayer. Hence, in this case, the implementation of wall conditions is straight­
forward, consisting simply of imposing no-slip and impermeability relations providing
the wall is ‘smooth’. The treatment is more difficult when log-law-based “wall laws” are
adopted in conjunction with high-Re models to bridge the viscous sublayer and when the
wall is ‘rough’. Particular difficulties arise when Reynolds-stress modelling is applies in
conjunction with curved wall; it is this aspect which is of particular interest herein. Atten­
tion is directed to a general near-wall volume abutting a curved wall, as shown in Figure
3.8. The point P is assumed to be in the log-law region, with the log variation assumed
to prevail normal to the wall at any tangential velocity position. The resultant shear force
\( F_{\text{shear}} \) acting on the cell’s southern face \( \text{Area}^s \) is:

\[
F_{\text{shear}} = \tau_w \text{Area}^s, \tag{3.136}
\]

where

\[
\tau_w = \frac{\rho_{\text{pp}}^{1/2} C_{\mu}^{1/4} \kappa}{\ln(E^* l_n k_p^{1/2}/\nu)}, \tag{3.137}
\]

and

\[
u' = u - u^n, \quad l_n = (r_p - r_s)n. \tag{3.138}
\]

Here \( l_n \) is the normal distance away from the wall, and \( u' \) and \( u^n \) denote the tangential and
normal velocity components, respectively. The unit normal vector pertaining to Figure
3.8 is \( n = \nabla \eta / |\nabla \eta| \) or in expanded form:

\[
(n_1, n_2, n_3) = \frac{(n_x, n_y, n_z)}{\sqrt{n_x^2 + n_y^2 + n_z^2}}.
\]  

(3.139)

Hence, the components of \( u' \) in equation (3.138) can be written as:

\[
(u'_x, u'_y, u'_z) = (u, v, w) - (u n_1 + v n_2 + w n_3) n,
\]  

(3.140)

with

\[
u'_x = (1 - n_1^2) u - n_1 n_2 v - n_2 n_3 w,
\]  

(3.141)

\[
u'_y = (1 - n_2^2) v - n_1 n_2 u - n_2 n_3 w,
\]  

(3.142)

\[
u'_z = (1 - n_3^2) w - n_1 n_3 u - n_2 n_3 v.
\]  

(3.143)
Once the tangential velocity components are resolved, the coefficient $A_S$ in the discretized equation pertaining to the near-wall cell is first set to zero, and then the source $S_C$ is modified in such a manner as to explicitly include the shear force imposed on the southern cell face as follows:

for the $x$-momentum equation:

$$S_C \leftarrow S_C - \frac{\rho_p k_p^{1/2} c^1/4 \kappa}{\ln(E^* l_n k_p^{1/2} / \nu)} u'_x A_{era}^x,$$  \hspace{1cm} (3.144)

for the $y$-momentum equation:

$$S_C \leftarrow S_C - \frac{\rho_p k_p^{1/2} c^1/4 \kappa}{\ln(E^* l_n k_p^{1/2} / \nu)} u'_y A_{era}^y,$$  \hspace{1cm} (3.145)

for the $z$-momentum equation:

$$S_C \leftarrow S_C - \frac{\rho_p k_p^{1/2} c^1/4 \kappa}{\ln(E^* l_n k_p^{1/2} / \nu)} u'_z A_{era}^z,$$  \hspace{1cm} (3.146)

where

$$A_{era}^x = \sqrt{\eta_x^2 + \eta_y^2 + \eta_z^2}.$$  \hspace{1cm} (3.147)

To implement equations (3.144)–(3.146), the correct value of $k_p$ is needed. In the near-wall cell, this is, essentially, governed by the balance between volume-averaged production and dissipation. Note that, both must be evaluated in a manner consistent with the log-law variation in the cell. If the entire cell is assumed to reside within the log-law region, the average $k$-production arises as (Launer [95]):

$$\overline{P_k} = \frac{\ln(l_n/l_v)}{\rho_p \nu l_n k_p^{1/2}} (\tau_w.\tau_w).$$  \hspace{1cm} (3.148)
With \( \varepsilon = 2\nu k/l_n^2 \) in the viscous sublayer and \( \varepsilon = k^{3/2}/(\kappa l_n) \) in the log-law region, the average value of \( \varepsilon \) becomes:

\[
\bar{\varepsilon} = \frac{\rho \mu^2 p}{l_n} \left( \frac{2\nu}{l_n k_p^{1/2}} + \ln\left(\frac{l_n}{l_p}\right) \right).
\] (3.149)

Since the \( \bar{\varepsilon} \) is unconditionally positive (but preceded by a minus sign), iterative stability can be enhanced by the replacement:

\[
S_p \leftarrow S_p - \frac{\rho \mu^2 p}{k_p} j.
\] (3.150)

The aforementioned modifications in the near-wall region suffice for the implementation of \( k-\varepsilon \) model. The extension of the above treatment to the Reynolds-stress model is less straightforward than might seem at the first sight. In a Cartesian framework, the average near-wall stress productions are well approximated by:

\[
\bar{P}_{11} = 2\bar{P}_k, \quad \bar{P}_{22} = 0, \quad \bar{P}_{33} = 0,
\] (3.151)

while the shear stress itself is given by:

\[
-\rho \mu \dot{v}^2 \approx \tau_w.
\] (3.152)

A further difficulty arises due to the additional average pressure-strain terms, \( \Phi_{ij} \), which need to be evaluated and, unlike the case of turbulence energy, they contribute substantially to the balance equations. Note that the pressure-strain terms contain products of stresses and strains, and the variation of the former across the sublayer is both uncertain and highly influential to the averaging process. Considerable further complications arise...
in a non-Cartesian environment because of the tensorial nature of the stresses, productions and \( \Phi_{ij} \), and the consequent complex transformation involved in determining productions and contributions of \( \Phi_{ij} \) in terms of wall-oriented coordinates. Thus, the \( k \)-equation, incorporating the production and diffusion terms appropriate to the second-moment closure, is solved rather than the equations for \( \overline{u'^2}, \overline{v'^2} \) and \( \overline{w'^2} \), where equations (3.148) and (3.150) are used. Then all near-wall values of the Reynolds-stress components are specified in accordance with wall-oriented Reynolds stresses followed by a local coordinate transformation as follows (see Lien & Leschziner [105]):

\[
\begin{align*}
\overline{u'^2}_p &= (\overline{u'^2})' t_1^2 + (\overline{v'^2})' n_1^2 + (\overline{u'v'})' t_1 t_1 n_1 \\
\overline{v'^2}_p &= (\overline{u'^2})' t_2^2 + (\overline{v'^2})' n_2^2 + (\overline{u'v'})' t_2 t_2 n_2 \\
\overline{u'v'}_p &= (\overline{u'^2})' t_1 t_2 + (\overline{v'^2})' n_1 n_2 + (\overline{u'v'})' (t_1 n_2 + t_2 n_1)
\end{align*}
\] (3.153-3.155)

where

\[
(\overline{u'^2})' = 1.098k_p, \quad (\overline{v'^2})' = 0.247k_p, \quad (\overline{u'v'})' = -0.255k_p. \quad (3.156)
\]

3.8.3 Wall-Function-Based Boundary Conditions

The Reynolds-stress model used herein is applicable to high-Re-number forms only. In consequence, the log-law of the wall needs to be used in some manner to arrive at boundary conditions for the stress equations. The simplest set of wall function appropriate to a near-wall layer in local equilibrium may be written (see Patankar & Spalding [123]):

\[
\frac{u}{u_r} = \frac{1}{k} \ln \left( \frac{E u_r}{\nu} \right), \quad k = C_{\mu}^{-1/2} u_r^2, \quad \varepsilon = \frac{u_r^3}{k \nu}. \quad (3.157)
\]
It can be shown that, in local equilibrium, the stresses may be related to the turbulence energy by

\[ \overline{u'_i u'_j} = c_{ij}k, \quad (3.158) \]

where \( c_{11}, c_{22} \) relative to \( u^2 \) and \( v^2 \) are

\[
c_{11} = \frac{4C_1 + 2C_1^2 - 4C_1C_2 + 2C_1C_2C_w + 6C_{1w} + 6C_1C_{1w} - 6C_{1w}C_2}{3C_1(C_1 + 2C_{1w})}, \quad (3.159)
\]

\[
c_{22} = \frac{2(-1 + C_1 + C_2 - 2C_2C_w)}{3(C_1 + 2C_{1w})}, \quad (3.160)
\]

respectively. The coefficient \( c_{12} \) pertaining to the shear stress \( u'v' \) can be derived from its related equation as follows:

\[
P_{12}(1 - C_2 + 1.5C_2C_w) - \frac{\rho \varepsilon}{k} (C_1 + 1.5C_{1w}) \overline{u'v'} = 0. \quad (3.161)
\]

Replacement of \( P_{12} \) by \( -\overline{v'^2} \rho u^2 / (\kappa \nu) \) in conjunction with \( \varepsilon = u_i^2 / (\kappa \nu) \) and \( c_{22} \) above yields

\[
\overline{u'v'} = \frac{k}{\sqrt{\left\{ \frac{1 - C_2 + 1.5C_2C_w}{C_1 + 1.5C_{1w}} \right\} \left\{ \frac{2(-1 + C_1 + C_2 - 2C_2C_w)}{3(C_1 + 2C_{1w})} \right\}}. \quad (3.162)
\]

If the standard values for coefficients \( C_1, C_2, C_{1w} \) and \( C_{2w} \) are adopted, equations (3.159), (3.160) and (3.162) become

\[ c_{11} = 1.098, \quad c_{22} = 0.247, \quad c_{12} = -0.255. \quad (3.163) \]

The treatment is over-simplified in situations in which the near-wall layer departs from equilibrium. In such circumstances, a more refined treatment must be adopted. One level
of refinement is the use of $k^{1/2}$ rather than $u_\tau$ as the velocity scale. This replacement is particularly advantageous near reattachment points where $u_\tau \approx 0$, so that $k = C_\mu^{-1/2} u_\tau^2$ is far from true. In this case turbulent velocity scale does not scale with $\sqrt{\tau_w/\rho}$. A second level of refinement is to divide the near-wall layer into two sublayers: a viscous sublayer of thickness $y_v$ and a fully turbulent layer above it, as shown in Figure 3.9. In the viscous sublayer the turbulent stress is held to be zero, so that

$$\frac{u}{u_\tau} = \frac{y u_\tau}{v}, \quad u_\tau = \sqrt{\frac{\tau_w}{\rho}}. \quad (3.164)$$

While in the fully turbulent region,

$$\frac{u k^{1/2}}{\tau_w/\rho} = \frac{1}{\kappa^*} \ln \left( E^* \frac{y k^{1/2}}{v} \right), \quad (3.165)$$

where $\kappa^* = C_\mu^{1/4}$ $\kappa$. In order to match the velocities in equations (3.164) and (3.165) at $y_v$,

$E^*$ is required to be:

$$E^* = \frac{\exp \left( \kappa^* y_v k_v^{1/2} / v \right)}{y_v k_v^{1/2} / v}. \quad (3.166)$$

Within a finite-volume framework, use of wall laws requires the near-wall cell to be such that its centroid, $P$, in Figure 3.9 is well into the log-law layer, i.e. $y^+ \approx 60 - 100$. Since the near-wall turbulence energy is required in equation (3.165), its value at $P$, namely $k_P$, needs to be determined, but this requires the solution of the cell-averaged $k$-equation. Hence, the cell-averaged production and dissipation must be determined. This might appear to be a numerical artifact, but is really closely intertwined with the physical assumptions made in relation to the near-wall stress and energy variations as shown in
Figure 3.9. To obtain cell-averaged values for generation and dissipation, they must be integrated across the cell. In the viscous sublayer, it can be assumed that

\[ k = k_v \left( \frac{y}{y_v} \right)^2, \quad \varepsilon = \frac{2\nu k_v}{y_v^2}, \quad -u'v' = 0, \] (3.167)

while in the fully turbulent region

\[ k = k_p, \quad \varepsilon = \frac{k^{3/2}}{C_l y}, \quad -u'v' = \tau_w, \] (3.168)

(Shown by dashed lines in Figure 3.9), or

\[ k = \left( \frac{k_n - k_v}{y_n - y_v} \right) y + \left( \frac{k_p - k_N}{y_p - y_N} \right), \quad \varepsilon = \frac{k^{3/2}}{C_l y}, \quad -u'v' = \tau_w + (\tau_n - \tau_w) \frac{y}{y_n}, \] (3.169)

(Shown by solid lines in Figure 3.9). With the former option, the average production \( \overline{P}_k \) and dissipation \( \overline{\varepsilon} \) can be obtained as:

\[ \overline{P}_k = \frac{1}{y_n} \int_{y_v}^{y_n} \tau_w \frac{\partial u}{\partial y} dy = \tau_w \left( \frac{u_n - u_v}{y_n} \right) = \frac{\tau_w^2}{\rho \kappa y_n k_p^{1/2}} \ln \left( \frac{y_n}{y_v} \right), \] (3.170)
which is simple and, thus, widely used. The solutions pertaining to the latter option are cumbersome and can be found in the article by Chieng & Launder [27].

Now, with regard to dissipation, it is standard practice not to solve the related equation, but to use the explicit relation

\[ \varepsilon_p = \frac{k_p^{3/2}}{C_l y_p}, \]  

which is consistent with the log-law and has been used above to evaluate the cell-averaged dissipation rate entering the near-wall \( k \)-equation. The above consideration relating to the cell-averaged production and dissipation have been pursued in terms of wall-oriented quantities. It is thus finally necessary to indicate how these are implemented in terms of the Cartesian stresses. All principles may be elucidated by focusing, for simplicity, on the local equilibrium conditions equations (3.157)–(3.163); quite analogous considerations apply to the two-layer approach.

The Reynolds stresses in equation (3.158) are viewed, first, as being decomposed in a local framework with its coordinates tangential and perpendicular to the wall. Then, a stress transformation mentioned in the previous section is employed to calculate all related Reynolds stresses in the Cartesian coordinates, which are subsequently fixed in the near-wall cell as boundary conditions. This practice in detail may be described, by reference to Figure 3.9, in the following sequences:

1. The wall shear stress, used to represent momentum fluxes through the solid surface,
is given by:

\[ \tau_w = \frac{\rho k_p^{1/2} C_{\mu}^{1/4} \kappa}{\ln(E^* l_n k_p^{1/2}/\nu)} (n_2 u_p - n_1 v_p) \]  \hspace{1cm} (3.173)

where the normal distance \( l_n = r_p n \).

2. The average production and dissipation in \( k \)-equation is expressed in the following forms:

\[ \overline{P_k} = \frac{\rho (n_2 u_p - n_1 v_p)^2 k_p^{1/2} C_{\mu}^{1/4} \kappa}{\ln(E^* l_n k_p^{1/2}/\nu)} l_n \quad \overline{\epsilon} = \frac{k_p^{3/2}}{C_l l_n} \]  \hspace{1cm} (3.174)

3. The near-wall value of dissipation rate in \( \epsilon \)-equation is fixed, via \( \epsilon_p = k_p^{3/2}/(C_l l_n) \).

4. Following the stress transformation procedure mentioned in (3.129), all Reynolds stresses in the vicinity of wall are fixed by the expressions below:

\[ \overline{u'^2} = (\overline{u'^2})' t_1^2 + (\overline{v'^2})' n_1^2 + 2(\overline{u'v'})' t_1 n_1, \]  \hspace{1cm} (3.175)

\[ \overline{v'^2} = (\overline{u'^2})' t_2^2 + (\overline{v'^2})' n_2^2 + 2(\overline{u'v'})' t_2 n_2, \]  \hspace{1cm} (3.176)

\[ \overline{u'v'} = (\overline{u'^2})' t_1 t_2 + (\overline{v'^2})' n_1 n_2 + (\overline{u'v'})' (t_1 n_2 + t_2 n_1). \]  \hspace{1cm} (3.177)

where

\[ (\overline{u'^2})' = 1.098 k_p, \quad (\overline{v'^2})' = 0.247 k_p, \quad (\overline{u'v'})' = -0.255 k_p. \]  \hspace{1cm} (3.178)

5. Since \( k \)-equation is solved, \( \overline{w'^2} \) can be extracted, if necessary by

\[ \overline{w'^2} = 2k_p - \overline{u'^2} - \overline{v'^2}. \]  \hspace{1cm} (3.179)
Finally, for a rough wall, following Sajjadi & Aldridge [138], the constant $E$ in equation (156) is replaced with a roughness coefficient evaluated as in Krishnappan [86] by the expression

$$E = \frac{\exp(k B_s)}{k_s^+}$$

(3.180)

Here $B_s$ is determined empirically as a function of $k_s^+ = u_t k_s / \nu$, where $k_s$ is the Nikuradse's sand grain roughness. For convenience in the numerical implementation, the Nikuradse [119] has been curve-fitted by the following expression

$$B_s(k_s^+) = (5.5 + 2.5 \ln k_s^+) \exp[-0.062 (\ln k_s^+)^3]$$

$$+ 8.5 \left[ 1 - \exp[-0.062 (\ln k_s^+)^3] \right]$$

(3.181)

as shown in Figure 3.10.

3.8.4 Plane or Symmetry

At plane of symmetry, the normal gradient of all conserved properties vanish, a condition which is equivalent to $\partial \phi / \partial n = 0$.

3.8.5 Fluid Exit

A fluid exit does not normally arise naturally and must be defined at some position to restrict the solution domain. If the exit plane is placed sufficiently far downstream of the disturbed region, the condition $\partial u / \partial y = \partial v / \partial x = 0$ may be prescribed with little loss of accuracy. A more difficult case is one in which fluid exits a cavity or plenum through a gap in the wall into stagnant surroundings. Here, the flow is highly curved at the exit,
and the appropriate hydrodynamic boundary condition is an invariant value of stagnation pressure, from which the exit velocity would need to be determined. For any other scalar property, a zero-gradient condition may still be acceptable.

3.8.6 Periodic Condition

Like an exit, a periodic boundary is artificial and designed to restrict the size of the solution domain. Periodicity simply implies an anti-symmetric transfer of information. In practice, this can be achieved by an implicit coupling of the inlet with its mirror image.
along the exit. However, an easier practice will be to use the inlet boundary and updating conditions there by copying the conditions along the exit after every iteration, subject to a global mass-flux constraint.
4.1 Theoretical background

4.1.1 The Poisson equations

Grid generation is a procedure for the calculation of the Cartesian coordinates \((x,y,z)\) of each grid node, i.e., a process of finding a relationship between computational coordinates \((\xi, \eta, \zeta)\) and Cartesian coordinates \((x,y,z)\).

For most fluid flow applications, the concentration of mesh points and orthogonality in the boundaries must be carefully controlled. Skewness of grid lines should also be avoided, since it leads to poor discretization of the differential equations, which in turn yields low convergence and accuracy. Numerical grid generation using Laplace differential equations (and Poisson differential equations) is mostly used due to the associated smoothness of the grids produced by these methods. The Poisson system has the following formulation

\[
\begin{align*}
\xi_{xx} + \xi_{yy} + \xi_{zz} &= P_c + P_i \\
\eta_{xx} + \eta_{yy} + \eta_{zz} &= Q_c + Q_i \\
\zeta_{xx} + \zeta_{yy} + \zeta_{zz} &= S_c + S_i
\end{align*}
\]

(4.1)

In this set of Poisson equations, the terms \(P_c, Q_c\) and \(S_c\) are forcing terms which allow the control of the grid lines clustering near the boundaries. In turn, \(P_i, Q_i\) and \(S_i\) are forcing terms used to transmit the grid spacing into the interior of the domain. The need for these
terms is well evident if one examines Figure 4.1, where a grid was generated over a two-
dimensional sinusoidal hill. In this grid, no forcing terms (or control functions) were
used, \textit{i.e.}, a solution was obtained by converting equation (4.1) into \textit{Laplace} equations.
The smoothing effect of these equations lead to a uniform grid distribution inside the
domain.

As a result, the distance of grid lines to the boundaries increases in convex regions
and the non-uniform grid distribution in the boundaries is not transmitted into the interior
of the domain, leading to a higher degree of grid skewness.

Since the unknowns in equation (4.1) are the Cartesian coordinates, the solution of
these equations must be carried out in the computational space. For that purpose, the
dependent and independent variables are interchanged, yielding

\begin{figure}[h]
\centering
\includegraphics[width=0.8\textwidth]{grid.png}
\caption{Grid generated about a two dimensional hill with no control functions.}
\end{figure}
\[ \alpha_{11} r_{\xi}^2 + \alpha_{22} r_{\eta}^2 + \alpha_{33} r_{\zeta}^2 + 2(\alpha_{12} r_{\xi \eta} + \alpha_{13} r_{\xi \zeta} + \alpha_{23} r_{\eta \zeta}) = \]

\[-J^2 \left[ (P_c + P_l) r_{\xi} + (Q_c + Q_l) r_{\eta} + (S_c + S_l) r_{\zeta} \right] \quad (4.2)\]

where

\[ r = (x, y, z) \]

\[ \alpha_{ij} = J^2 (\nabla \xi_i, \nabla \xi_j) \quad (4.3) \]

\[ \xi_1 = \xi; \quad \xi_2 = \eta; \quad \xi_3 = \zeta \quad (4.4) \]

The metric relations \( \alpha_{ij} \) in equation (4.3) are computed using the following equation written in the computational space as

\[ \alpha_{ij} = \sum_{m=1}^{3} A_{mi} A_{mj} \quad (4.5) \]

where \( A_{mj} \) is the co-factor of the \( (m, i) \) element in the matrix

\[
\begin{bmatrix}
  x_\xi & x_\eta & x_\zeta \\
  y_\xi & y_\eta & y_\zeta \\
  z_\xi & z_\eta & z_\zeta
\end{bmatrix}
\quad (4.6)
\]

The Jacobian of the transformation is given by the determinant of matrix equation (4.6), \textit{i.e.}

\[
J = \begin{vmatrix}
  x_\xi & x_\eta & x_\zeta \\
  y_\xi & y_\eta & y_\zeta \\
  z_\xi & z_\eta & z_\zeta
\end{vmatrix} \quad (4.7)
\]
4.1.2 The control functions: Steger and Sorenson

Steger & Sorenson [154] proposed, for two dimensions, an efficient way for achieving a pre-specified grid clustering of grid lines near the boundaries, only requiring imposition of Dirichlet boundary conditions in two opposite boundaries. Furthermore, orthogonality at the boundaries is automatically achieved, independently of the geometry.

MESH3D employs the original proposal of Steger and Sorenson SS, with the necessary modifications for three dimensions.

Let us assume that a three dimensional grid is to be generated between two opposite boundaries defined by $\zeta = \zeta_1$ and $\zeta = \zeta_{\text{max}}$ (c.f. Figure 4.2).

The terms $P_c, Q_c$ and $S_c$ are functions that allow the control of the clustering of the $\zeta$ and $\eta$ lines ($S_c$), as well as the orthogonality of the $\zeta$ lines ($P_c$ and $Q_c$) near boundaries $\zeta = c^\epsilon$. If these characteristics are to be achieved near $\zeta = \zeta_1$ (Figure 4.2), then an exponential decay into the interior of the domain is imposed to the control functions.

Figure 4.2: Schematic representation of the physical space near the surface $\xi = \zeta_1$. 

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
$P_c, Q_c$ and $S_c$

\begin{align}
P_c &= P_1 e^{-a(\zeta - \zeta_1)} \\
Q_c &= Q_1 e^{-b(\zeta - \zeta_1)} \\
S_c &= S_1 e^{-c(\zeta - \zeta_1)}
\end{align}

(4.8) \hspace{1cm} (4.9) \hspace{1cm} (4.10)

where $a, b$ and $c$ are decay factors and $P_1, Q_1$ and $S_1$ are the control functions evaluated at the boundary.

To determine $P_1, Q_1$ and $S_1$, equation (4.2) are evaluated at $\zeta = \zeta_1$, yielding the following set of equations for these control functions

\begin{align}
P_1 &= \frac{(y_\zeta z_\eta - z_\zeta y_\eta)R_1 + (z_\zeta x_\eta - x_\zeta z_\eta)R_2 + (x_\zeta y_\eta - y_\zeta x_\eta)R_3}{f^3} \\
Q_1 &= \frac{(y_\zeta z_\eta - z_\zeta y_\eta)R_1 + (z_\zeta x_\eta - x_\zeta z_\eta)R_2 + (x_\zeta y_\eta - y_\zeta x_\eta)R_3}{f^3} \\
S_1 &= \frac{(z_\zeta y_\eta - y_\zeta z_\eta)R_1 + (x_\zeta z_\eta - z_\zeta x_\eta)R_2 + (y_\zeta x_\eta - x_\zeta y_\eta)R_3}{f^3}
\end{align}

(4.11)

where

\begin{align}
R_1 &= \alpha_{11}(x_1)\xi + \alpha_{22}(x_1)\eta + \alpha_{33}(x_1)\zeta + 2[\alpha_{12}(x_1)\xi \eta + \alpha_{23}(x_1)\eta \zeta + \alpha_{31}(x_1)\zeta \xi] \\
x_1 &= x; \quad x_2 = y; \quad x_3 = z
\end{align}

(4.12)

Note that, in this derivation, control functions $P_1, Q_1$ and $S_1$ were set to zero.

In the previous equations, all the derivatives $\xi$ and $\eta$ are known, since the location ($x, y$ and $z$ coordinates) of the grid points on the surface $\zeta_1$ is specified as a boundary condition. The derivatives in $\zeta$ are computed as a function of the (imposed) distance $\Delta s$
of the first grid points \( \zeta = \zeta_2 \) to the surface \( \zeta = \zeta_1 \). This distance is specified \textit{a priori} and may be written as a function of the co-variant metrics as follows

\[
\Delta s = \sqrt{(x_\zeta'^2) + (y_\zeta'^2) + (z_\zeta'^2)}
\] (4.13)

The other two conditions needed for determination of the derivatives in \( \zeta \) are provided by the requirement that \( \zeta \) lines intersect the surface in a nearly orthogonal fashion. This requirement is satisfied as long as the co-variant base vectors \( \hat{r}_\zeta \) and \( \hat{r}_\zeta \), as well as \( \hat{r}_\eta \) and \( \hat{r}_\zeta \), are orthogonal, \textit{i.e.}

\[
r_\zeta \cdot r_\zeta = (x_\zeta \hat{\imath} + y_\zeta \hat{j} + z_\zeta \hat{k}) \cdot (x_\zeta \hat{\imath} + y_\zeta \hat{j} + z_\zeta \hat{k}) = x_\zeta^2 + y_\zeta^2 + z_\zeta^2 = 0
\] (4.14)

\[
r_\eta \cdot r_\zeta = (x_\eta \hat{\imath} + y_\eta \hat{j} + z_\eta \hat{k}) \cdot (x_\zeta \hat{\imath} + y_\zeta \hat{j} + z_\zeta \hat{k}) = x_\eta x_\zeta + y_\eta y_\zeta + z_\eta z_\zeta = 0
\]

The combination of equations (4.13) and (4.14) provides the unknown derivatives in \( \zeta \) as follows

\[
x_\zeta' = \frac{(z_\eta y_\zeta - y_\eta z_\zeta) \Delta s}{A}; \quad y_\zeta' = \frac{(x_\eta z_\zeta - z_\eta x_\zeta) \Delta s}{A}; \quad z_\zeta' = \frac{(y_\eta x_\zeta - x_\eta y_\zeta) \Delta s}{A}
\] (4.15)

where

\[
A = \sqrt{(z_\eta y_\zeta - y_\eta z_\zeta)^2 + (x_\eta z_\zeta - z_\eta x_\zeta)^2 + (y_\eta x_\zeta - x_\eta y_\zeta)^2}
\] (4.16)

Evaluation of the second \( \zeta \) derivatives at \( \zeta = \zeta_1 \) equation (4.12), is made using a one-sided second order discretization (Steger & Sorenson [154]), as follows

\[
r_{\zeta\zeta} = \left( \frac{-7r_1 + 8r_2 - r_3}{2\Delta \zeta^2} \right) - \frac{3r_{\zeta\zeta}}{\Delta \zeta^2}
\] (4.17)
where the subscript 1, 2, 3 indicates the position of the grid node relative to the boundary, along the $\zeta$ line.

Similar equations are written for imposition of the grid clustering at other boundaries.

The equations derived previously allow the determination of the forcing terms $P_\zeta, Q_\zeta,$ and $S_\zeta,$ as a function of the imposed $\Delta s$ distance. The coefficients $a$ and $b$ in equations (4.8)–(4.10), by acting on the $\zeta$ lines, determine how deep inside the computational domain the orthogonality condition is transmitted. The coefficient $c,$ in the same equation, controls the attraction of the $\xi$ and $\eta$ lines to the boundary. Typical values for $a, b$ and $c$ are in the range of 0.2 to 1. As an example, Figure 4.3 depicts a grid generated for a 2D sinusoidal hill. It is well evident in this figure, the constant spacing between the first surface of nodes and the boundary, as well as the intersection of the $\zeta$ lines with the hill in an orthogonal fashion.

In Figure 3.4, the value used for the decay coefficient $c$ was higher than in the pre-
vious cases, yielding a lower clustering of the grid lines near the boundary (note that, nevertheless, the distance $\Delta s$ remains practically unchanged). The effect of the coefficient $a$ (coefficient $b$ is inactive, since this is a 2D case) may observed in Figure 4.5. The grid depicted in this figure was generated using the same parameters as for the grid of Figure 4.3, except for the value of coefficient $a$, which was higher in Figure 4.3. It may be observed that, in Figure 4.5, the grid lines tend to keep the direction prevailing near the surface much deeper inside the domain.

4.1.3 The control functions: Thomas and Middlecoff

The grids previously presented were generated incorporating source terms designed to allow the transmission of the grid spacing into the interior of the domain. The need for these terms derives from the fact that Laplace equations (and Poisson equations without special formulation) tend to produce a homogeneous grid spacing inside the domain. This may be appreciated by examining Figure 4.6, which represents a grid generated without
Figure 4.5: Grid generated about a two dimensional hill \((a = 0.2, c = 0.4)\).

Figure 4.6: Grid generated about a two dimensional hill, without transmission of the grid spacing into the interior of the domain \((a = 0.5, c = 0.4)\).

One may note how this grid has a higher degree of skewness than the previous ones, which is a feature that should be avoided. The formulation of the control functions that allow to overcome this problem was proposed by Thomas & Middlecoff [158], in the
following manner

\[ P_t = \frac{\alpha_{11}\phi}{J^2} \]  (4.18)

\[ Q_t = \frac{\alpha_{22}\psi}{J^2} \]  (4.19)

\[ S_t = \frac{\alpha_{33}\omega}{J^2} \]  (4.20)

where

\[ \phi = \frac{r_\xi r_\xi \zeta}{r_\xi r_\xi} \]  (4.21)

\[ \psi = \frac{r_\eta r_\eta \eta}{r_\eta r_\eta} \]  (4.22)

\[ \omega = \frac{r_\zeta r_\zeta \zeta}{r_\zeta r_\zeta} \]  (4.23)

The control functions \( P_t, Q_t \) and \( S_t \) transmit the \( \xi, \eta \) and \( \zeta \), and spacing, respectively, into the interior of the domain. If, for instance, the \( \xi \) spacing is to be transmitted along the \( \zeta \) direction, the term \( \phi \) should be computed at the locations \( \zeta_j \) and \( \zeta_{\text{max}} \). These values are then interpolated into the interior of the domain. It is important to note that, unlike the term \( \phi, \alpha_{11}, \) and \( J^2 \) are not computed at the boundaries and then interpolated. Rather, these terms are evaluated at each node and then equation (4.18) is applied to compute the control function \( P_t \). If the \( \xi \) grid spacing is to be transmitted along the \( \eta \) direction as well, the values of \( \phi \) in each node of the interior of the domain are obtained by simply summing up the contributions obtained from the interpolations of \( \phi \) between the two pairs of boundaries.

The effect of Thomas and Middlecoff (TM) control functions may be adjusted by applying a multiplying factor lower than unity.
4.1.4 Solution of the equations

Substituting equations (4.18)-(4.20) in equation (4.2) and rearranging the terms we obtain

\[ \alpha_{11} r_{\xi \xi} (J^2 P_c + \alpha_{11} \phi) + \alpha_{22} r_{\eta \eta} + (J^2 Q_c + \alpha_{11} \psi) r_{\eta} + \alpha_{33} r_{\zeta \zeta} + (J^2 R_c + \alpha_{11} \omega) r_{\zeta} + 2(\alpha_{12} r_{\xi \eta} + \alpha_{23} r_{\eta \zeta} + \alpha_{31} r_{\zeta \xi}) = 0 \]  

(4.24)

which are the equations that allow the computation of the coordinates of each node in the interior of the domain. In these equations, the second derivatives are evaluated using central differences. The first derivatives are evaluated using forward or backward differences, depending on the sign of the corresponding coefficient. Then, equation (4.24) are cast in the following general form

\[ a_P r_P = a_E r_E + a_W r_W + a_N r_N + a_S r_S + a_T r_T + a_B r_B + b \]  

(4.25)

where the subscripts \( E, W, N, S, T, B \) stand for the computational relative position of the nodes: East, West, North, South, Top and Bottom, respectively, and

\[ r_P = (x, y, z)_{i,j,k} \]  

(4.26)

\[ r_E(x, y, z)_{i+1,j,k}; \quad r_W(x, y, z)_{i-1,j,k} \]  

(4.27)

\[ r_N (x, y, z)_{i,j+1,k}; \quad r_S (x, y, z)_{i,j-1,k} \]  

(4.28)

\[ r_T (x, y, z)_{i,j,k+1}; \quad r_B (x, y, z)_{i,j,k-1} \]  

(4.29)

Exemplifying for the East-West direction, the coefficients affecting the neighbour points are computed as follows

If \( J^2 P_c + \alpha_{11} \phi > 0 \), \( a_E = \alpha_{11} + (J^2 P_c + \alpha_{11} \phi) \) and \( a_W = \alpha_{11} \)  

(4.30)
If \((J^2 P_c + \alpha_{11} \phi) < 0\), \(a_E = \alpha_{11}\) and \(a_W = \alpha_{11} - (J^2 P_c + \alpha_{11} \phi)\) \(\quad (4.31)\)

The coefficient for the central location is

\[
a_P = a_E + a_W + a_N + a_S + a_T + a_B
\]

\(\quad (4.32)\)

and the source term is given by

\[
b = 2(\beta_1 r_{\xi \eta} + \beta_2 r_{\eta \xi} + \beta_3 r_{\zeta \xi})
\]

\(\quad (4.33)\)

Sub-relaxation of control functions

TM control functions do not need sub-relaxation. SS control functions need to be strongly sub-relaxed at the beginning of the iteration process. Sub-relaxation may then be reduced up to 0.25. The process used in MESH3D is to increase the relaxation factor \(\gamma\) in an exponential way along the iterative process, according to the following function

\[
\gamma = \frac{10 \left[ \frac{\beta}{\gamma_{\text{max}}} \right] n}{10^8 \gamma_{\text{max}}}
\]

\(\quad (4.34)\)

where \(\beta\) represents the rate of increase, \(n\) is the present iteration number and \(\gamma_{\text{max}}\) is the maximum sub-relaxation coefficient, reached at iterations. The dependence on \(\gamma\) with \(\beta\) is represented in the graphic of Figure 4.7, for \(n_{\text{max}} = 100\) and \(\gamma_{\text{max}} = 0.25\). Higher values of \(\beta\) (not exceeding 4) are recommended for problems with lower values of \(\Delta s\) (clustering distance), which are more prone to stability problems during the early stages of the iterative process.
4.1.5 Relaxation of the iterative process

Solution of equation (4.25) is performed using a Point Successive Over Relaxation method. The optimum relaxation parameter is computed at every point in the grid following the procedure proposed by Ehrlich [42] as follows. We first rewrite equation (4.25) in the following vector form

\[-a_P r_P + a_E r_E + a_W r_W + a_N r_N + a_S r_S + a_T r_T + a_B r_B = -b\]  \hspace{1cm} (4.35)

The complex eigenvalues of these equations are

\[\mu = \mu_r + \mu_i = \frac{2}{a_P} \left( \sqrt{a_E a_W \cos \frac{\pi}{l_{\max}}} + \sqrt{a_N a_S \cos \frac{\pi}{j_{\max}}} + \sqrt{a_T a_B \cos \frac{\pi}{k_{\max}}} \right)\]  \hspace{1cm} (4.36)

where \(\mu\) and \(\mu_i\) are the real and the imaginary part of \(\mu\), respectively. In fact, in the present case, the imaginary part of (30) does not exist, since positive coefficient are always
ensured. We next define

\[ A = \mu_r^2 + \mu_i^2 \]  
(4.37)

\[ B = \mu_r^2 - \mu_i^2 \]  
(4.38)

\[ C = A^2 - B^2 \]  
(4.39)

\[ D = A^2 - B \]  
(4.40)

\[ E = \sqrt{C + D^2} \]  
(4.41)

\[ F = \sqrt[3]{C} \]  
(4.42)

and

\[ \bar{\kappa} = \frac{(3D + E)F \sqrt{E - D} - (3D - E)F \sqrt{E + D} + A^2 + 3B^2 - 4A^2B}{A^2D} \]  
(4.43)

Finally, the relaxation parameter \( \kappa \) is given by

\[ \kappa = \begin{cases} 
\frac{\left( \bar{\kappa} - \sqrt{\bar{\kappa}^2 + 4\bar{\kappa}} \right)}{2} & \text{if } D > 0 \\
\frac{\left( \bar{\kappa} + \sqrt{\bar{\kappa}^2 + 4\bar{\kappa}} \right)}{2} & \text{if } D < 0
\end{cases} \]  
(4.44)

The computed values of \( \kappa \) can sometimes be a little too large, causing instabilities in the iterative process. To circumvent this, its values are multiplied by a limiting factor, whose default value is 0.7.
CHAPTER 4. GRID GENERATION FOR CURVILINEAR GEOMETRIES

Solution process

The initial field of variables (grid coordinates $x, y, z$) necessary to start the iterative process, is calculated through linear interpolation of the boundary values into the interior of the domain, and a few sweeps are performed before the introduction of the control functions. Control functions are introduced gradually, through the application of a sub-relaxation which is increasing (and approaching 1) with the number of iterations, according to equation (4.34).

No convergence criterion was implemented as a means for deciding to stop the iterations. Since the computer times are modest, the maximum number of iterations to be performed is specified a priori. The check for the convergence is made by monitoring the coordinates of a selected grid point.
Virtual Reality Simulation is currently being considered as a potential tool for the design and evaluation of environments before they are created, or recreated in a different context. Computer based visualization offers an enhanced capability for gaining insight and is an effective means for design communication and presentation. It provides an opportunity to verify hunches, analyze possibilities and gives unlimited scope for further experimentation. Virtual environments enable the user to walk through and interact with various design components.

Plato, in his famed Republic, introduced "The Allegory of the Cave" through which he was able to explore the concepts of reality and human perception. In his exploration, Plato used the analogy of a man who defined the foundation of his reality by the dancing shadows of firelight, which shone upon the walls of his cave. From this analogy, Thomas DeFanti and Dan Sandin conceived the idea of the Cave Automatic Virtual Environment (CAVE) in the spring of 1991. Graduate candidate Carolina Cruz-Neira and several other collaborators endeavored to realize this dream in the later half of 1991. Working at the Electronic Visualization Laboratory (EVL) at the University of Illinois in Chicago, they developed their first prototype of the CAVE and unveiled it as part of the Showcase feature at the ACM SIGGRAPH convention in 1992. SIGGRAPH is an annual meeting hosted by the Association for Computing Machinery (ACM) and is devoted to the art and science of computer-generated graphics. See Figure 5.1 for an example 3D visualization.
The emerging technology of virtual reality and immersive techniques will change the way data is viewed and analyzed, as well as the way models will be developed and the results will be interpreted. Using virtual reality methods to create and navigate through an artificial world based on the dataset provides an actual sense of presence to the user. Medical exploration and intervention at the cell and gene levels are facilitated in a virtual environment since its micro- and macro- scaling features allow surgeons to work as if the areas were expanded. Virtual medical facilities, virtual equipment and virtual patients

Figure 5.1: Example CAVE immersive visualization system displaying a terrain data set.
CHAPTER 5. IMMERSIVE VISUALIZATION

give practitioners and students more opportunities for hands-on experience.

The 21st century now offers an infinite amount of opportunity for us humans to explore the vast depths of science and technology. The CAVE offers a vast number of benefits for architects, pathologists, anesthesiologists, surgeons, doctors, engineers and even military personnel. The hypothesis for this project would be that AutoCAD models could be visualized in CAVE Immersive Environments. The conversion process through NuGraf would have to be investigated and researched. If the hypothesis is accepted, then complex AutoCAD models could be drawn and visualized in the CAVE. For example, a pathologist could walk through a cell membrane to gain a better understanding of the different components like the phospholipids, integral proteins and cholesterol that make up the cell membrane.

The Cornell Theory Center (CTC) and VRCO worked together to create the first ever immersive display system running the CAVELib on a cluster of Windows 2000 workstations. This breakthrough is one of the latest advancements in the ability to drive an immersive system with commodity hardware. Using Dell Personal Computers equipped with 3Dlabs Wildcat graphics cards, and Giganet networking, VRCO worked to extend the CAVELib to take advantage of these devices and made displaying virtual environments a reality with low-end systems.

With a focus on service, research and development, and technical innovation, CTC supports scientists working in Computational Finance, Computational Materials, and Computational Biology. 3D scientific visualization on the desktop is a workday tool.
for these researchers and they benefit greatly from stereo immersive virtual reality environments in which they can explore and share their data with peers. Until recently, access to these expensive, high-end systems has been limited by cost. With the advent of the Windows-based CAVE environment, the cost of installation and operation (total cost of ownership) for these systems promises to drop so dramatically that the stereo CAVE might soon join desktop VR as a standard tool for research.

"We expect that the development of cost effective solutions for immersive virtual reality environments, such as the Windows version of CAVElib from VRCO, will bring these systems into more widespread use. And we want to introduce students to the tools they will need as academic, government, and industry researchers after graduation," says CTC executive director Linda Callahan.

Presently at CTC, the necessary hardware to run CAVE is just a few Windows PCs. The PC hardware is off-the-shelf, and VRCO's CAVElib software does the difficult synchronization tasks for you, making those PCs look like any other distributed CAVE. This allows quick access to the standard library of visualization applications and offers low cost, good speed and good scalability.

In recent years, immersive visualization systems have emerged with different implementations worthy of special names. For example, some systems have been developed for use in the area of networked virtual reality. Collaborative Virtual Environments (CVEs) involve the use of networked virtual reality systems to support cooperation between groups of people. Also, the terms Distributed Virtual Reality (DVR) and Dis-
CHAPTER 5. IMMERSIVE VISUALIZATION

Distributed Interactive Virtual Reality (DIVE) have been created to describe immersive visualization systems with distributed computer resources. AccessGrid (AG) technology is similar to DIVE which allows groups of people to interact and share data over the internet. The Internet2 project uses the term Tele-Immersion to describe a 3D virtual environment where two or more people in different geographic locations can come together to interact.

3D reconstruction for tele-immersion is performed in stereo, which means two or more cameras rapidly capture images of the same object where distance calculations from each camera are continuously performed, and projects the images into a computer simulated environment, as a replicate to real-time movement.

Prototypical example software, such as KnotPlot, now use immersive visualization to better understand physical knot theory. KnotPlot has been ported to run in the Immersive Media Lab (IML) at the New Media Innovation Centre (NewMIC) using CAVELib. This new version of KnotPlot opens up exciting possibilities. For example, in the field of physical knot theory (knots considered as tangible physical objects), researchers are interested in the surfaces of self-contact of a tightly pulled knot. By being able to easily move around and peer into such a knot in a Virtual Reality (VR) environment, researchers can investigate knot theories much easier. See Figure 5.2

The advent of high performance computers (HPC) has allowed materials researchers to model large 3D nanostructures. Properties associated with these 3D structures can be sufficiently complex where researchers can benefit from immersive virtual environments (VEs) in their analysis and interpretation of HPC model results. Because immersive
VEs are expensive and not as convenient compared to the researchers desktop computers, desktop VE-simulators have become important for day-to-day routine analysis. Recently desktop computer graphic cards have significantly improved and consequently emphasized the use of desktop VE-simulators when analyzing large structures. In spite of these advances at the desktop, immersive VEs have continued to prove invaluable in special cases. Although both immersive VEs and desktop VE-simulators are useful, analysis is typically limited to a single researcher.

With high-speed networks, immersive VEs and desktop VE-simulators can be combined into a share collaborative workspace. The first shared VE applications were created using a new Application Programming Interfaces (API) called DIVERSE. To foster col-
laborative development the DIVERSE API was licensed GNU-GPL/LGPL. These collab­
orative VE applications were designed to enhance researchers interpretation and analysis
of HPC results. For several years now material researchers have used commercial graph­
ical tools at their desktop computers to interpret and analyze properties associated with
large 3D nanostructures predicted by their high performance computer models. Many
3D nanostructures that can be visually interpreted and analyzed at desktop computers can
also be analyzed and interpreted in immersive virtual environments such as a CAVE. In
some cases immersive VEs can be more insightful from the researchers viewpoint. The
CAVE is a multi-person, room-sized, high-resolution, 3D video and audio environment.
In the current configuration, graphics are rear projected in stereo onto three walls and the
floor, and viewed with stereo glasses, see Figure 5.1. Because it is more convenient for
researchers to work at their desktop computers, there was a need to extend visual anal­
ysis done at the desktop to more insightful immersive VEs when necessary. Hence VE
applications were developed that scaled from the desktop computer to an immersive VE
such as a CAVE. These VE applications were also networked together into a collaborative
working environment which enhanced researchers interpretation and analysis. Here we
summarize the development and use of these collaborative VE applications. Creation of
nanostructure visualization applications were based on two different software APIs:

1. The CAVE-libraries which were created at the Electronic Visualization Laboratory
of the University of Illinois.

2. DIVERSE which was created at the University Visualization and Animation Group
at Virginia Tech. The CAVE-Libraries libraries were developed when the CAVE itself was first created.

In collaboration with National Center for Supercomputing Applications (NCSA), Atomview, Collaborative CAVE Console (CCC), and CCC_atom were applications developed using the CAVE-Libraries. For a discussion of Atomview, CCC, and CCC_atom, the reader is referred to the Proceedings of the 2003 SCS High Performance Computing (HPC) Symposium 2003. Another VE, Device Independent Virtual Environments: Reconfigurable, Scalable, and Extensible (DIVERSE) was also created at Virginia Technology Foundation. The DIVERSE API was developed at Virginia Tech in collaboration with the National Institute for Standards and Technology (NIST). DIVERSE was designed with a common user interface to interactive graphics and/or VE programs. Using DIVERSE, the same program can be run on CAVE, Reconfigurable Advanced Visualization Environment (RAVE), Immersa-Desk, head mounted display (HMD), desktop, and laptop without modification with an emphasis on desktop VE simulation. To enhance collaboration between different users on the Internet, DIVERSE was also designed to share data generated from different I/O (Input/Output) devices such as USB hand held navigation devices, e.g. joysticks, pocket-PCs or six degree-of-freedom (DOF) tracking systems that share users head/wand positions. For example, these DOFs can then be used to draw a visual representation of another users position in the shared VE. To enhance the collaborative development of VE applications DIVERSE was licensed as GNU-GPL/LGPL. The current version of DIVERSE, which currently runs on SGI-IRIX and GNU-Linux.
CHAPTER 5. IMMERSIVE VISUALIZATION

operating systems was written using SGI-Performer and is called DPF. A recent beta release of DIVERSE, rewritten in OpenGL, which is called DGL, will be used to port to other operating systems, e.g. Windows and Macintosh. Several DIVERSE DPF applications were created in collaboration with NIST and with funding from Lockheed Martin to create a generic collaborative virtual design environment. For a discussion of Atomview, CollabTools and DXwand the reader is referred to the Proceedings of the 2003 SCS High Performance Computing Symposium 2003.

5.1 Immersive Visualization Technical Difficulties

Physical models, such as atmospheric, phase transition, sediment transport, etc., typically have solutions involving partial differential equations. Usually these models are time-dependent but not always. From a mathematical viewpoint, the solutions involve continuous variables that computers cannot represent. As stated in Chapter 2, these physical models must be discretized for use on a computer. The solutions generated are not exact, but approximate and involve representing the model domain with a discrete set of points called a grid or mesh. Depending on model parameters, a tetrahedral mesh is used that meets the Delaunay property (Cavendish [22]). Delaunay tesselations are important due to their favorable features from the computational point of view. See 5.3 for an 2D example of a refined mesh that meets the Delaunay property. Several numerical algorithms have been introduced to solve and make efficient rendering volume data sets (Albertelli & Crawfis [2], Grosso et al. [51], Haber et al. [55], Hanrahan [58], Hesselink et al. [62],
Even though much work has been done on improving rendering performance of rectilinear volumes, volume rendering of non-rectilinear (i.e. curvilinear and unstructured) grids remains an open problem (Hong [67]). A curvilinear grid can be the result of non-linear transformations on a rectilinear grid of cubic voxels. This approach has the effect of warping around a complex shaped object while preserving the grid topology. The curvilinear grid has the same implicit connectivity as the rectilinear grid, yet unlike the rectilinear grid, the 3D vertex grid locations must be explicitly defined. As a result, each quadrilateral cell face is not necessarily planar and each six sided cell is not necessarily convex. These technical concepts are needed to render a data set as the example shown of Figure 5.4 Advances in graphics hardware and firmware are making VE systems more capable and cost effective to mass produce. Lower system cost will provide feasibility to
more and more researchers to build or purchase a complete system. Also, with lower costs even commercial industry and eventually the general public will have access to immersive visualization to support their business and entertainment needs. In the next chapter we discuss our own RAVE system and components.
Immersive visualization is gaining greater acceptance in scientific research due to the advantages of allowing the scientist to develop a deeper understanding of the problem being studied. With the emergence of computer graphics research came many computer hardware related limitations. Early computers lacked the computational power and memory space needed to produce displays even for 2D data sets (Hill [64]). The emergence of immersive visualization concepts involves creating displays of 3D and higher order data sets requiring even higher performance computer and graphics hardware. A immersive visualization system needs high performance hardware because of these main factors: high computation load of physical models, computational requirements of simultaneously rendering multivariate data sets through multiple projectors, and providing navigation tools for the user to interact with the data set.

Developed at EVL, the CAVE enables scientists to interact with virtual worlds that have been created from observed data or simulations. The main purpose of the CAVE is to link users in the CAVE with remote users for shared collaborative virtual environments. The EVL CAVE system is a high-resolution projection-screen virtual reality system powered by a Silicon Graphics Onyx parallel-processor computer. The screens are arranged in a 10-foot cube with computer-generated images projected on three walls and a floor. A viewer wears a 6-degrees of freedom head tracker device and stereo-shutter glasses so that the correct projections and perspectives are presented as the viewer moves inside the
CAVE. This creates a stereoscopic effect where the depth information encoded in the virtual scene was restored and conveyed to the eyes of those using it. Some CAVE systems only use stereo-shutter glasses. A wand held by the viewer allows interaction with the virtual environment. Six to ten people can stand in the CAVE and view the projection, while one person with a headset and computer wand controls the simulation’s perspective. Also available is CAVE simulator software, which allows one to develop applications for the CAVE on local platforms, then load the applications on the CAVE for immersive visualization.

6.1 Our System

CAVE systems evolved into RAVE systems by designing the screens so that they can be rotated from a CAVE configuration all the way to a 30 foot wide wall screen. Figures 6.1 and 6.2 shows two different configurations of our system.

1. Three 8’ tall by 10’ wide vertical projection walls One 8’ by 10’ projection floor.

2. Four Christie Digital Mirage 2000 DLP projectors, displaying 1280 x 1024 @ 96Hz resolution.

3. Ten pairs of Stereographics Crystal Eyes III LCD eyewear Three extra long-range IR stereo sync emitters.

4. InterSense IS-900 acoustic motion tracking system MiniTrax wand and head tracking devices.
Figure 6.1: RAVE configured as a standard CAVE system.

Figure 6.2: RAVE configured as a 30 foot flat wall.
5. Four DELL computers, each has: 2 Pentium Xeon 2.0 GHz processors, 1 3Dlabs WildCat III 6210 Video Card, and 4GB RAM.
7.1 Introduction

In Chapter 3 we discussed the formal theory behind fluid flow including finite volume discretization schemes, boundary conditions, convection schemes, pressure correction algorithms, reynolds-stress modeling, wall reflection algorithms, and more. Now we introduce turbulence from a qualitative perspective first, then present algorithmic details. Consider the turbulence depicted in Figure 7.1. Studies show a few basic observations can be made:

1. Turbulence is random. The properties of the fluid, such as density, pressure, and velocity, at any given spatial or temporal point cannot be predicted. However, statistical properties, such as time and space averages, correlation functions, and probability functions, show regular behavior. Turbulent fluid motion is stochastic.

Figure 7.1: Turbulence introduced into flow field by flow around an obstacle.
2. Turbulence decays without energy input. Turbulence must be driven or else it decays, returning the fluid to a laminar state.

3. Turbulence displays scale-free behavior. On all length scales larger than the viscous dissipation scale but smaller than the scale on which the turbulence is being driven, the appearance of a fully developed turbulent flow is the same.

4. Turbulence displays intermittency. Outer fluctuations occur more often than chance predicts.

### 7.2 Oscillatory Turbulent Flow

The oscillatory boundary layer flow over rippled sand beds has very complicated features which involves different types of turbulence. The main source of turbulence at the sea bed is related to tidal and wave action (Sajjadi et al. [140]). The purpose of most practical calculations in turbulent flows is to obtain ensemble average statistics such as the mean values of velocity $U_i$, pressure $p$, temperature $\theta$, and moments $\bar{u}_i\bar{u}_j$, $\bar{p}\bar{u}_i$, etc., given certain boundary and initial conditions. The central problem in any computation of turbulent flows is to adopt an appropriate and accurate closure scheme (Launder [97]). As stated in Chapter 2, this dissertation focuses on the body of research related to immersive visualization using direct numerical simulation modeling techniques. In this chapter, we narrow our focus to developing a stable numerical model for calculating oscillatory turbulent boundary layer flow over a 3D domain. Figure 7.2 shows a prototypical 3D computational domain for visualizing the complex oscillatory turbulent flow model used.
in our research. Although such models are improving and providing better results when
compared with experiments, however there still remain questions about their validity in
various regions of the flow field, such as the near-wall anisotropies that are inherent in
flows near obstacles (Moin et al. [116]). Equally important in computation turbulent
flows is the appropriate choice of numerical methodology, which must be adopted to ob­
tain an accurate solution. The closure methodology adopted for any turbulent flows are
usually achieved with comparison with other turbulent models, experimental data and of­
ten with Direct Numerical Simulation (DNS) of turbulence (Kim et al.[73]), which gives
strikingly similar results to those observed experimentally.

Understanding the structure of turbulence is essential to understanding practical appli­
cations that depend on local fluctuations of eddy motions. These local fluctuations have
more to do with the local field statistics than the ensemble statistics. The reason is ensem­
ble statistics provide information about the state of the entire domain. Therefore, there must be a focus on the structure of turbulence in order to develop models that can better represent local fluctuations. The numerical model must accurately model how sediment is lifted from the bed into suspension. We would like to use the RAVE and visualization techniques to improve current theories and models involving sediment transport in wave dominated environments. Our results can be applied to improving coastal engineering models of sand transport. These models are of particular interest to researchers needing to predict sediment transport rates.

7.2.1 Present Research Objectives

Algorithms and methods from Chapter 3 will be implemented to produce a new algorithm that is physically accurate and computationally efficient. Our investigation will focus on two principle approaches:

1. A DNS simulation of the oscillatory flow over flat and undulating surfaces.


This dual approach is intended to combine a method that is closer to the underlying physics of turbulent flow (the DNS simulation), but which is still too computationally expensive for direct use in engineering calculations, with an approach which is capable of such calculations (DSM) but which requires significantly greater assumptions to model the turbulent flow. The DNS will provide information on the turbulent correlations in such
flows that can be used to check the DSM turbulence modeling. It is proposed here to apply
the method of DNS to the study oscillatory flows over flat and undulating surfaces. The
investigators will then use the DNS results to develop more accurate closure models for
turbulent correlations, which arise in the DSM approach to describing turbulence. These
improved closure models will then be incorporated in the general 3D non-orthogonal
curvilinear DSM software. Particle tracking will be included in the DSM software and
the pickup, transport and deposition of sand particles will be investigated with the aim
of understanding the physics of this process and to develop simple parameterizations that
can be used in Engineering applications. The DSM model will also be used in order to
study the formation and subsequent migration of sand ripples. The DNS will be based
on an existing software designed to run over undulating topography, but will be adapted
to handle time dependent boundary conditions so as to allow an oscillatory flow to be
modeled. Such a flow over a flat smooth bed has been approached via the Direct Nu-
merical Simulation (DNS) technique (Spalart et al.[152]). However, the use of DNS over
a rough undulating surface, with the roughness elements due to sand grains, would be a
novel application and a significant challenge in it’s own right. The DNS software will be
used to study the large-scale vortices created in the lee of sand ripples which are ejected
when the flow reverses. This mechanism has been observed, and provides a mechanism
for entraining sediment high into the boundary layer flow and is the dominant mechanism
governing sand transport for oscillatory flow over rippled beds (Sajjadi [139]). There
have been a number of attempts to model this process, but none of these studies have led
to a complete physical explanation of the mechanism involved. From the DNS results, the ensemble average velocities and statistics of the mean and fluctuating quantities at each phase of the oscillation will be determined.

DSM calculations solve for the mean flow and need significantly fewer computer resources, but require closure models to account for the effects of turbulence. It is proposed that the results from the DNS calculation be used to improve an existing software developed to study flow and sediment transport over rippled beds (Sajjadi [138]). Budgets of production, dissipation pressure strain, at different stations along the sand ripple obtained from the DNS technique, will be compared with observation and with DSM calculations (Buckles et al. [20]). Initial calculations and experiments concerning formation and lift up of vorticies using a DSM closure model have been completed and reported in previous research (Sajjadi [139]). It will be necessary to include the transport of particles. In this proposal a discrete Lagrangian approach will be adopted which is based on the equation of motion of single particles and mean random drag forces caused by flow turbulence. The pickup and release of particles will be based on empirical studies relating rate of particle erosion to bed shear stress. Deposition of particles will occur due to the fall velocity. Given the model statistics for the velocity field over an undulating bed, the trajectories of particles are then simulated. This requires assumptions about the lift off (mainly a result of surface velocity fluctuations) and deposition. The methods are quite standard for level surfaces but their application to simulate the growth of the ripples is quite new and will require flow and trajectory calculations of the sand ripple growth. In the follow-on phase
of the project, the dynamics of sand ripple migration will be studied using the DSM software. This will entail the calculation of the net balance of erosion minus deposition at each point along the sand ripple profile and an adjustment of the mesh at that point.

The computational procedure is based on a generalized curvilinear coordinate system that is tessellated to form triangular shaped finite elements. These finite elements are used to implement a collision detection algorithm needed to impose boundary conditions. Our model considers time and length scales of the turbulent flows and calculates a low-order statistical description and general form of eddy structures using local mean velocity gradients. First, in order to simplify the computational aspects of the model, the data set needs to be re-sampled so that the model's runtime is reasonably small. The next section describes a new algorithm used to accomplish the re-sampling.

7.2.2 Re-sampling vector fields for better visualization

A number of techniques are available for re-sampling the results of a spatial measurement (Chetverikov [25]). Most regular grid techniques perform re-sampling from one regular grid to another, even though the data is on an irregular grid points. Due to the volume of data to be processed, and thus the high order of computational complexity, a quick and easy re-sampling algorithm is needed. A new approach, the bilinear interpolation scheme illustrated in Figure 7.3 (Chetverikov [25]). Given a regular grid point \( p_g \), the algorithm interpolates the velocity vector of point \( p_g \) by including points \( p_i, i = 1, 2, .., k \), that lie within a specified distance \( S \) from \( p_g \). The algorithm is constructed as follows: Let the velocity vectors of points \( p_i \) be defined by \( v_i \), and let \( d_i^2 \) define the squared distance from
Let define the mean of over all $i$. From these definitions the following quantities can be introduced:

$$\alpha_i = e^{\left(-\frac{d_i^2}{d^2}\right)} \quad \beta_i = \frac{\alpha_i}{\sum_{i=1}^{k} \alpha_i}, \text{ where } 0 < \alpha_i, \beta_i \leq 1 \quad \text{and} \quad \sum_{i=1}^{k} \beta_i = 1 \quad (7.1)$$

Thus, the interpolated velocity at point $p_g$ is computed as:

$$v_g = \sum_{i=1}^{k} \beta_i v_i \quad (7.2)$$

This interpolation scheme is used to compute how each sand particle moves through the domain. In the computational domain the particles Cartesian coordinates and volume cell index parameters are stored. The cell index parameters are used to efficiently determine which finite element cell the particle belongs to as well as provide a foundation to implement the collision detection algorithm. Collision detection becomes important particularly when a particle approaches the sand bed. Ideally, the bottom profile must be used to determine where and how fast the particle is reflected. More work is needed to accurately model surface interaction effects. For this work, an completely elastic collision model was used which means that a sand particle that collides with the surface will stick

---

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
to the surface for the current iteration. Subsequent model iterations may allow the sand particle to become suspended once again in the water column.
A PROTOTYPICAL APPLICATION

In this chapter, we present the implementation details of a prototypical application using the algorithms and techniques presented in chapters 2 through 7. The application was written in the C++ programming language using Microsoft's Visual Studio Integrated Development Environment. We chose Microsoft's Visual Studio because of extensive prior programming experience using the environment and the fact that our RAVE system was designed as a Microsoft Windows platform. The software architecture of the application is organized into domain initialization and processing sections. Adaptations to other development environments and platforms can be accomplished without any major problems.

8.1 Domain Initialization

The domain size and other simulation parameters are controlled by an input file that the application loads before the data structures are created. The purpose of this design is to allow scientists to study different datasets provided the datasets are stored in an acceptable format. For our prototype application, we use an example dataset cited in previous work. This dataset consists of 122 grid points in the horizontal, or $x$, direction, 40 grid points in the vertical, or $y$, direction, and 30 grid points in the $z$ direction which is orthogonal to the $x$ and $y$ axis and is interpreted as being normal to the viewing area for 3D graphics when projected onto a 2D plane. The $x,y,z$ grid parameters are used to initialize the computational environment and load the flow vector dataset. During the loading process, the raw
vector data spatial coordinates and corresponding velocity components are interpolated to fit the non-regular, adaptive grid spatial coordinates using the techniques described in Chapter 4. See Figures 8.1 and 8.2 for a graphical depiction of the grid generated.

The dataset includes 30 time frames each one consisting of velocity components, \( u, v, w \), for each \( x, y, z \) grid point. Each time frame represents \( \Delta t = 0.1269 \) seconds, or a total of 3.8 seconds in real-time. See Figures 8.3 and 8.4 for a graphic depiction of the flow vectors for different time frames. The flow vectors in these figures display only the back plane, \( z = 0 \), for viewing ease. Additionally, common computations needed for sand particle movement calculations are calculated and stored in a data structure in order to reduce the computational load on the processor during simulation.
8.2 Processing

This section describes how each time frame is processed and displayed to the user. In Chapter 6, we presented details how graphic images are displayed in stereographic mode.
Displaying each time frame involves the following:

1. Draw the left, right, and back vertical sections of the domain.

2. Draw the wave surface

3. Draw the sand particles.

The user has access to an on-screen menu allowing for viewing of the major computational components. See Figure 8.5 for the menu that provides user interactive controls for the application. The menu allows the user to study effects by turning on or off the various components used in the model computations. The left and right vertical planes are easy to draw and only serve to establish the extents of the domain. The back vertical plane includes drawing the adaptive grid lines. The water waves are created by using solid-fill
elements from the top grid points. The bottom, or ocean floor, is created using the bottom grid points and used to construct rectilinear area elements. These area elements are triangulated by adding an diagonal edge which are then used to implement a collision detection algorithm. See Figure 8.6 for a graphical depiction of the normal vectors computed for each area element. These normal vectors are an important part of the collision detection algorithm.

To begin, sand particles are generated and placed at random barycentric locations within each triangular bottom area element. These sand particles are then subjected to the turbulent flow and domain restrictions as a function of position and time. More specifically, for each sand particle, a new location is computed based on the current location of the particle and neighboring fluid flow vectors. The new cartesian location is accepted for
each time step provided the sand particle does not collide with the bottom. In the event a collision does occur, the new location is rejected and replaced by the barycentric collision point on the bottom. To satisfy physical model parameters discussed in Chapter 3, the collision is treated as completely elastic. Therefore when a collision exists, the sand particle’s location moves from the old location to the computed barycentric collision point. See Figures 8.7 through 8.12 for a graphical depiction of model results at selected time frames.
Figure 8.7: 3D sand particles at rest.

Figure 8.8: 3D sand particles when lefted.
CHAPTER 8. A PROTOTYPICAL APPLICATION

Figure 8.9: 3D sand particles lefted and suspended.

Figure 8.10: 3D sand particles when falling.
Figure 8.11: 3D sand particles when falling and deposited.

Figure 8.12: 3D sand particles deposited near the bottom surface.
CHAPTER 9

CFD SIMULATIONS FOR

SEDIMENT TRANSPORT IN DREDGE LANE

9.1 Introduction

The management of aggregate extraction areas and their associated dredge lanes relies on a good understanding of the potential impacts of the dredge tracks (generated by the action of the drag-head) and the wider sediment transport regime. Anecdotal evidence indicates that dredge tracks are associated with apparently enhanced accumulations of sand. If dredge tracks are acting as potential sand sinks, this has implications for both the management and restoration of the site and sediment transport through the aggregate extraction region. The objective here is to investigate tidal flows and suspended sediment transport associated with dredge tracks in dredging lanes arising from aggregate extraction, with the ultimate aim of providing advice on minimizing changes to sediment transport.

The first task is validating the existing computational fluid dynamics (CFD) algorithm described in earlier chapters in order to simulate tidal flows and suspended sediment transport over bed features typical of aggregate extraction dredging. Following modifications are made to the algorithm in order to include pressure-gradient forcing and to enable rough boundaries to be simulated. A sediment-transport module involving re-suspension, transport and deposition is also developed, allowing budgets of material to be computed.

Model results are then compared with analytic solutions and with previous model studies...
over flat beds.

For the validation of the algorithm, a series of simplified scenarios is developed that could be modelled using the CFD algorithm. These included the dredge-track orientation with respect to the principal axis of the tidal ellipse, dredge-track morphology and particle size.

From first principles, field observations and established models of sub-aqueous bedform development (Van Rijn [?], Middleton et al. [114], Ashley et al. [3]), there is an identifiable suite of changes which are likely to occur, in some form or another, associated with the tracks made by dredgers (Figure 9.1). These include:

(1) lowering of relief of the tracks and within the dredging lanes;

(2) preferential infilling of the lanes by sands at one end of the lanes (upstream with respect to net bed sediment transport);

(3) the formation of a suite of mobile sedimentary bedforms at the downstream end of the lane, consisting of sand released by one of more of the dredging processes. The bedforms form a sequence away from the dredging area, the precise nature of which is dependent upon the local degree of sediment availability and mobility.

These changes in the texture of the bed and in the natural patterns and rates of bed sediment transport exert a primary control on the nature and timing of changes in the benthic biota, both during the dredging processes, and following the cessation of dredging. As well as influencing biological recoverability of the dredged region, there are also likely
impacts for depositional regions further along the sand transport path that will experience a decrease in sand supply. Whilst the general impacts upon the natural sedimentary regime can be hypothesized, the detail (timing, magnitudes & durations) of such impacts requires testing.

In this dissertation we apply the computational fluid dynamics (CFD) model, described in Chapters 3 and 7, to examine the effect on suspended sediment transport of the presence of dredge tracks or tracks. The work extends classic work on modelling turbulent tidal flows over flat beds (Johns [76], Davies et al. [36], Davies & Jones [37], Chapalain et al. [23]) to the non-flat case, including suspended sediment transport. Initially it was envisioned that a full three dimensional (3-D) time dependent model would be considered, but after problems with obtaining periodic tidal velocity solutions in 3-D, the effort was focused on the two-dimensional (2-DHV) case.

The primary objective of this dissertation is to investigate tidal flows and sediment transport over dredge tracks in dredging lanes arising from aggregate extraction, with the aim of providing advice on minimizing changes to sediment transport.

In investigating this issue, the approach used has modified an existing CFD (Computational Fluid Dynamics) model validated for flow over bed features and has applied it to the case of tidal flows through dredge tracks within dredging lanes. This tool has been used to investigate tidally driven sediment transport into, within and out of a dredge track. Various dredging and environmental scenarios which influence the impacts of dredging on sedimentation have been developed and applied using the CFD model. The relevant
parameters involved and varied are:

- Dredge track orientation relative to the tidal currents;
- Tidal current speed;
- Sediment grain size.

9.2 Sediment transport model

We adopt the hydrodynamic algorithm described earlier in this thesis and further incorporate a sediment transport model. The hydrodynamic model solves the non-hydrostatic three-dimensional Reynolds-averaged Navier-Stokes equations on a three-dimensional curvilinear finite-volume mesh.

A full description of the formulation and numerical implementation, apart from the suspended sediment module, is given in Chapter 7. The suspended load module, which is significantly enhanced, is described later. The principal features relevant for this application are that the model:

1. solves the model for the full dynamic pressure field;

2. uses a turbulent kinetic energy based turbulence model;

3. uses a general boundary-fitted mesh.

Together, these enable the simulation of complex flow structures such as Lee vortices and secondary motions potentially generated by non-uniform seabed features resulting
from dredging operations. However, the model does not include the rise and fall of the free surface with the tide, but most of the sediment would be expected to be concentrated near the bed, and it has been assumed here that the interactions with the free surface have a negligible effect.

The software is modified to simulate turbulent tidal flows and suspended sediment transport over stretches of seabed at horizontal scales comparable to dredge track dimensions (horizontal extent of the model is 6 m) and in water depths of the order of 15 m. This required a number of modifications to the original model, as described next.

(a) Two general approaches to driving flows in boundary-layer models are: (1) by prescribing velocities at the boundaries, or (2) by applying an external pressure gradient force, and using zero gradient conditions at the boundaries. The original model used method (1), however, for tidal flows, the approach using pressure gradient forcing is most appropriate so that results are not affected by the particular velocity profiles chosen. The model was adapted to allow this.

(b) Initial runs were carried out assuming the bed was smooth, but in reality the bed substrate is mainly gravel. Initially, computational problems arose when large values of bed roughness were used. This was resolved by changing the way in which the velocity boundary conditions were prescribed at the seabed.

(c) The original implementation of the pressure correction algorithm was not found to produce re-circulation in the downward lee of steps in bathymetry. Some time was
required to resolve this and necessitated a reduction from a 3-D to a 2-D solution, but retaining all three velocity components. Thus, the solution was obtained on a slice through a nominal 3-D domain with the assumption that the flow is fully developed in the along track direction. This should be a reasonable approximation away from the beginning and end of a track.

(d) It was found necessary to simplify the turbulence model from a two equation \( k-\varepsilon \) model to a one-equation \( k \) model because the two-equation model did not produce stable solutions when used with highly compressed meshes. Within the time constraints of this project, it was not possible to investigate this further. In the one-equation mode, the turbulent length scale is prescribed rather than calculated. For boundary-layer flows of the type being simulated here, well-accepted forms of the length scale exist (Blackadar [13]) and these were adopted.

(e) The implementation of suspended-sediment load solved the transport equation

\[
\frac{\partial C}{\partial t} + \frac{\partial (vC)}{\partial y} + \frac{\partial (wC)}{\partial z} = -w_{f}\frac{\partial C}{\partial w} + \frac{\partial}{\partial y}\left( v_{T} \frac{\partial C}{\partial y} \right) + \frac{\partial}{\partial z}\left( v_{T} \frac{\partial C}{\partial z} \right)
\]

(9.1)

where \( y \) and \( z \) are cross-track and vertical directions respectively, with corresponding velocity components \( v \) and \( w \). The sediment fall velocity is \( w_{f} \), and eddy diffusion coefficient is \( v_{T} \). Gradients in the \( x \) direction, along the dredge track, were assumed to be zero. The numerical scheme was enhanced to make it conserve mass, with budgets kept
of material in suspension and on the bed, and consistent with the fluxes calculated into
and out of the computational domain.

The boundary conditions at the bed was a flux condition

\[-v_f \frac{\partial C}{\partial z} = w_f c_b\]  \hspace{1cm} (9.2)

where the reference concentration was assumed to be given by

\[c_b = \frac{\gamma_0 S}{(1 + \gamma_0 S)}\]  \hspace{1cm} (9.3)

with the excess stress defined by

\[S = \max[(\tau - \tau_{cr})/\tau_{cr}, 0]\]  \hspace{1cm} (9.4)

where \(\tau_{cr}\) is value of bed shear stress \(\tau\) value below which suspended load was assumed
to be zero. The relation of Van Rijn [163] was used to determine when material was in
suspension, given by

\[\tau_{cr} = \begin{cases} 4w_f/D_s, & D_s < 10 \\ 0.4v_s, & \text{otherwise} \end{cases}\]  \hspace{1cm} (9.5)

where

\[D_s = D\left[(\rho_s/\rho - 1)/g\nu^2\right]^{1/3}\]  \hspace{1cm} (9.6)
is a non-dimensional particle diameter, in which the $\rho_s, \rho, \nu$ and $g$ are the sediment and water densities, kinematic viscosity and acceleration due to gravity respectively. When the local shear stress is below $\tau_{cr}$, any transport was assumed to take place by bedload and was not simulated by the model. The sediment pick-up function was modified to enforce the condition that the erosive bed flux became zero as the bed supply was exhausted, to ensure that the quantity of bed sediment remained $\geq$ zero.

Under some circumstances suspended sediment can act to stratify the flow (Soulsby & Wainwright [151]). Although this can be simulated by including buoyant production terms in the turbulence model, the criterion of Soulsby & Wainwright [151] indicated that the effect was insignificant, so that in the present situation these terms were not incorporated into the model.

Software was written in Visual Studio C++ to generate computational meshes with tracked boundaries and enhanced grid resolution near the bed. Visual Studio C++, OpenGL, and CAVElib were developed to visualize model output as line plots, contour plots and animations.

9.3 Results

For our modelling task we begin with validation of the model output by comparing it against exact analytic solutions (for simple cases) and against previously published work for the full tidal simulations.
9.3.1 Mesh Generation

The 2-D curvilinear mesh used for the simulations with the tracked bed is shown in Figure 9.2. Compression has been applied to pack the mesh more densely at the bed in order to resolve near-bed velocity and sediment concentration profiles. The CFD implementation allows for very general meshes and is not limited to the requirement that they be orthogonal. For model validation against analytic solutions and with previous results by other workers, a flat mesh (not shown) was generated with similar compression of the vertical mesh.

9.3.2 Steady-State Results

Steady-state solutions are of interest because analytic results (Appendix A) can often be obtained with which to compare with numerical results. A steady-state numerical solution was obtained by applying a constant pressure gradient forcing and running the model until equilibrium. A flat-bed mesh was used, and the same equilibrium vertical profile (as a function of height above the bed) was obtained at each bed location in the domain. An arbitrary bed location was chosen and numerical and analytic results were compared for velocity and horizontal shear stress.

The horizontal shear stress should show a linear decrease, from a maximum at the bed to a prescribed value, usually zero, at the sea surface. The maximum value at the bed is a function of the applied pressure gradient and the water depth (equation A.3, Appendix A) and independent of the form of the vertical eddy viscosity. The Figure 9.3 shows the
exact result compared with the numerical calculation. The model reproduces the required linear dependence and bed stress. There is a small discrepancy at the top boundary, which is due to the numerical software implementing the zero stress boundary condition at a point just above the boundary rather than precisely at it.

Taking the turbulent length-scale as proportional to distance from the bed, and retaining only production and dissipation terms in the turbulent kinetic energy equation leads to the mixing length form for the vertical eddy viscosity (equation A.4, Appendix A). For this case, an exact analytic solution for the steady-state mean velocity profile can be obtained (equation A.6, Appendix A). The numerical solution for the vertical velocity profile in the Figure 9.4 shows some discrepancy in the magnitude compared to the analytic result. Again this may be connected with implementation of the boundary condition at the surface. Nevertheless, the results were deemed sufficient to verify the basic implementation of the velocity and turbulent kinetic energy equations in the software.

An exact expression for the suspended sediment profiles in the steady state case was calculated for constant vertical eddy viscosity (equation A.9, Appendix A) and compared with numerical results. The Figure 9.5 shows extremely good agreement between numerical and exact results for two particle sizes.

9.3.3 Tidal Flow Results

Although steady state results are a useful test case, the situation of interest is oscillatory tidal flows. Although some analytic results for simplified versions of this case can be derived, the solutions are sufficiently complex that it was decided instead to make a
qualitative comparison with existing numerical solutions. The paper by Davies & Jones [37] gives detailed predictions for tidal flows and was used as a benchmark for validation of the present software. The calculations of Davies and Jones used a different value for one of the key turbulence model constants (Cm) and we adjusted this, and the turbulence model, to be as close to their formulation as possible. Therefore the turbulent length scale was prescribed using the Blackadar [13] form. Calculations were run in 3-D over a flat mesh. No significant horizontal variation in profiles occurred across the numerical mesh and an arbitrary point was therefore chosen for comparison.

The results are shown in Figure 9.6 to Figure 9.9. Profiles of the horizontal mean velocity in Figure 9.6 and shear stress in Figure 9.7 for all stages of the tidal cycle appear to be in excellent agreement with those found in Davies & Jones [37]. For the turbulent kinetic energy in Figure 9.8 agreement is also very good, apart from near the top boundary where the present model has turbulence levels decreasing toward zero, while the Davies and Jones results show a residual accumulation of energy. This difference is reflected in the turbulent viscosity in Figure 9.9 where the present model shows lower values for this quantity near the top boundary. Nevertheless, the comparisons show the present model to be capable of reproducing the key features of tidal boundary layers produced by other studies. Variations can probably be attributed to differences in the numerical scheme and the implementation of boundary conditions.
9.3.4 Results Over Dredge Tracks

Given that the model is able to reproduce the velocity and turbulence found by previous workers over flat beds, the model was applied to calculations over a tracked bed. All scenarios considered the bed substrate to consist of fixed gravel, overlain with quantities of the transportable sand. The bed was assumed to be covered with fixed 2 cm diameter gravel. The $z_0$ value was determined assuming the relation $z_0 = 2.5D/30$ (Soulsby [150]).

Sediment transport calculations were undertaken for sand grain diameters of 100, 250 and 500 $\mu$m (i.e. very fine sand, fine-medium sand and medium-coarse sand), using tidal current regimes corresponding to the scenarios discussed in section 9.4. In brief, these are:

- Principal M2 tidal axis aligned nearly parallel with the track, at a 10° angle.
- Principal M2 tidal axis aligned at 45° to the dredge track.
- Principal M2 tidal ellipse aligned at right angles to the track.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Track orientation</th>
<th>Peak speed ($ms^{-1}$)</th>
<th>Eccentricity</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0°</td>
<td>0.9</td>
<td>0.0</td>
</tr>
<tr>
<td>1</td>
<td>10°</td>
<td>1.01</td>
<td>0.17</td>
</tr>
<tr>
<td>2</td>
<td>45°</td>
<td>0.84</td>
<td>0.10</td>
</tr>
<tr>
<td>3</td>
<td>90°</td>
<td>0.80</td>
<td>0.15</td>
</tr>
</tbody>
</table>

*Table 9.1: Ellipse parameters for scenarios.*

To verify that the decrease in bed shear stress found for the angled case was also present when flow was parallel to the dredge track, a further calculation was carried out...
with the flow aligned exactly along the track. However, without cross flow, it was not possible to introduce sediment into the calculation so no corresponding calculation of sediment transport is available for this run. Velocities were adjusted to produce peak bed stress values of about 2.5 N\( m^{-2} \), for all ellipse orientations. In the model simulations, this corresponds to peak near-surface speeds of around 1 m\( s^{-1} \) and would correspond to spring tide conditions at a number of gravel extraction sites.

Forcing at a single M2 component was considered giving a symmetric tide apart from slight distortions generated from non-linear interactions in the model.

Within the track, the near bed cross-track velocity \( V \) in Figures 9.10, 9.11 and 9.12 for the 10°, 45° and 90° orientations respectively showed a small region of recirculation for all orientations of the tidal ellipse, but more pronounced at greater angles.

A snapshot of the turbulent kinetic energy distributions for each of the ellipse orientations Figures 9.13, 9.14, and 9.15 showed, apart from the 90° orientation case, an increase in the level of turbulent energy occurring at the lip of the track, and decreased values within the track. In these snapshots the cross flow goes from left to right and the right turbulence is most enhanced at the right hand lip of the track. Enhanced production of turbulence is associated with production arising from both vertical and horizontal velocity shear. Along the bottom of the track, turbulence kinetic energy values decreased due to decrease in the velocity shear. Away from the bed, levels of turbulence were enhanced in the region over the track compared to the ambient conditions on either side. The 90° orientation case (Figure 9.15) is somewhat different and shows enhanced turbulence.
within the track and as well as the right hand track lip. Although not visible in the plot, the turbulence levels within the track very near the near the bed values are diminished. This is illustrated in the next series of plots of the bed shear stress distribution, which, under conditions of local equilibrium is directly proportional to the turbulent kinetic energy (equation A.7, Appendix A).

The bed shear stress distribution across the track is shown for each ellipse orientation, including the situation with no cross flow (Figures 9.16, 9.17, 9.18 and 9.19 for 0°, 10°, 45° and 90° track orientations respectively). As suggested by the turbulent kinetic energy plots, the results show significantly diminished values within the track and enhanced shear stress at the lip. This is also the case when the flow is oriented at 90° (Figure 9.19). Where there is cross flow present, the side of the track facing the incoming cross flow shows enhanced bed shear stress. Conversely, the situation with flow ellipse aligned entire down the track (Figure 9.16) shows a symmetric stress distribution but still with significantly decreased values within the track.

Sediment was assumed to enter the domain upstream in the cross-track direction. The upstream direction alternates with the cross-track tidal velocity. Downstream a zero derivative boundary was applied. The input profile used a power law distribution based on the fall velocity and time varying bed shear stress at the boundary. The applied profile was zero when the bed shear stress fell below the criteria for suspension, but otherwise sediment supply was not assumed to be limited.

A snapshot of the sediment distribution for each orientation of the tidal ellipse and
each particle size is shown in Figure 9.20 to Figure 9.28. For 500 m particles, the suspended load was observed to be confined to a very thin layer near the bed and, qualitatively at least, acted rather like a bedload layer.

9.4 Dredging Scenarios

9.4.1 Synthesis Of Dredging Practice

One of the key aims of this project is to analyse dredging practice and synthesis this into a simple model with which the CFD software could be based. Three marine dredging locations have been assessed: Area 107 in the Outer Wash, Hastings Shingle Bank in the eastern English Channel and Owers Bank off the Shoreham coastline. (This location was suggested by Dr. S. Sajjadi and Aldridge.)

1. Area 107.

A side-scan sonar image (Figure 9.29) of the seabed taken in 2000 shows that the majority of the dredge tracks lie parallel with the major tidal axis with only a few tracks at approximately 10°- 20° to the main tidal axis. The interactions between the natural sediment transport environment of the Wash and the dredged tracks can be observed. Firstly, an area of megaripples (marked megaripples A) shows that at the time of the sonar image there is little interaction between the megaripples as there is no infilling of the tracks or distortion of the megaripples field. Conversely, a sandwave (marked Sandwave A) has partially filled the dredge tracks while another sandwave upstream of sandwave A has completely filled the dredge track.

A side-scan sonar image taken from Hastings Shingle Bank in 1999 shows high-intensity dredging in the southern line of sub area Y, the so-called Lane 9b (Figure 9.30). Multiple passes of the dredge head have resulted in an accumulation of dredge tracks resulting in no distinct individual tracks. All dredge tracks are parallel to the major tidal axis.

3. Owers Bank.

A side-scan (Figure 9.31) and swath bathymetry (Figure 9.32) survey undertaken in 2004 at the Owers Bank dredge site off Shoreham shows the physical impacts of many different dredging techniques, including static and trailer suction dredging. The prominent features are static dredge pits (typically 5 m deep and 35-40 m in diameter) where anchored dredgers have either swung on the tide using modest power and full rudder to move the dredge head gently across the target resource. More recent trailer-suction dredging is also evident especially in the north of the license area. Here the dredge tracks tend to be less controlled by the major tidal axis and hence angles with respect to the tidal axis are larger. Wide turns are also evident at the end of each dredger run across the license area.

In conclusion, the evidence indicates that most trailer-suction dredging activity is undertaken near-parallel to the major tidal axis (within 20°). Where intense dredging occurs, individual dredge tracks become less pronounced and shallow depressions are cre-
ated with small ridges (height 20 cm) parallel to the tidal axis. At the end of dredge lanes, the drag head is either lifted off the seabed before returning along previous lines or wide turns made resulting in curves on the seabed of large radius. Individual dredge tracks are approximately 3-4 m wide with a depth of up to 50 cm. Lips of positive relief can be observed on the edges of the tracks created by the drag-head as it moves through the substrate.

9.4.2 Simplified Scenarios

It was suggested by Dr. S. Sajjadi and Aldridge, under consultation with the marine aggregate extraction industry through BMAPA regulators at CEFAS Burnham-on-Crouch in UK and other stakeholders, that a series of simple scenarios to be developed in order to investigate the potential physical impacts of dredge lane with respect to suspended sediment transport. Note that, at this stage, it is only required a physical models for suspended sediment transport to be developed under the following scenarios:

- **Scenario 1** - this is the simplest scenario and involves near-rectilinear tidal currents acting on a series of parallel, straight dredge tracks. This also represents, to a first order, that found in reality (e.g. Hastings);

- **Scenario 2** - the straight tracks are orientated at 45° to the principal tidal axis;

- **Scenario 3** - the straight tracks are orientated at right angles to the flow.

9.4.3 Results

Results for each scenario are illustrated

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
(a) with a snapshot of the bed state at the last velocity minimum (slack water), and

(b) with the time series of the total material on the bed. All bed quantities are expressed as a percentage of the total amount of sediment that entered the domain during the simulation.

Scenario 1 near-rectilinear tidal currents acting nearly parallel to a straight dredge track

The snapshot of bed accumulation for all particle sizes for a track at 10° to the tidal ellipse is shown in Figure 9.34 to Figure 9.35 for 100, 250 and 500 μm particle sizes. Significant retention of the input material occurs for both the 500 and 250 μm particle sizes, whilst for the 100 μm sediments, retention is negligible. This is reflected in the time series of total bed accumulation (Figure 9.36) indicating retention of 20% for 500 μm particles over the simulation period.

Scenario 2 a straight track orientated at 45° to the principal tidal axis

The snapshot of bed accumulation for all particle sizes for a track at 45° to the tidal ellipse is shown in Figure 9.37 to Figure 9.39 for 100, 250 and 500 μm particle sizes. Significant retention of the input material occurs for both the 500 and 250 μm particle sizes, whilst for the 100 μm sediments, retention is negligible. This is reflected in the time series of total bed accumulation (Figure 9.40) indicating retention of the order of 25% for both 250 and 500 μm particles over the simulation period.

Scenario 3 a straight track orientated at right angles to the principal tidal axis

The snapshot of bed accumulation for all particle sizes for a track at 90° to the tidal ellipse is
CHAPTER 9. SIMULATIONS FOR SEDIMENT TRANSPORT IN DREDGES

shown in Figure 9.41 to Figure 9.43 for 100, 250 and 500 μm particle sizes. Significant retention of the input material occurs for both the 500 and 250 μm particle sizes, while for the 100 μm sediments, retention is negligible. This is reflected in the time series of total bed accumulation (Figure 9.44) indicating retention about 20 for both 250 and 500 μm particles over the simulation period.

The results are summarised in Figure 9.45 showing significant retention by the tracked bed for 250 and 500 μm sediments for all three track orientations. Conversely, retention of the 100 μm sediments was negligible.

9.4.4 Analysis

Sediment Retention

The results clearly indicate a mechanism for retention of coarser-grained sands by dredge track features which appears insensitive to track orientation with respect to the tidal ellipse. Reduced bed stress within the track compared to the surrounding flat regions causes the preferential retention of sediment within the track. This mechanism should be quite robust for particle sizes for which the in-track bed stress remains below the critical erosion stress for significant portions of the tidal cycle. Where this is not the case, retention may still occur if the suspended sediment load is hydrodynamically limited (that is, limited by the flow speed and turbulence) rather than supply limited. For the hydrodynamically limited situation, the suspended load outside the track is in equilibrium with the flat bed conditions and on moving across a track, sediment deposition would be expected to occur because the decrease in turbulence levels can no longer support the ambient suspended
sediment load. However, for sufficiently fine material, the simulations indicate that the reduced shear stress within the track will not retain material to any significant degree.

Although the simulations considered suspended load only, the same mechanism would be expected to pertain to bed load transport as well. In this case, retention within tracks may be enhanced by the slope effect contribution whereby bedload flux would be diminished in up-gradient directions. With regard to bedload transport, use of standard formula would be open to question, because bedload transport across gravel extraction sites is likely to occur as isolated thin patches of sand moving over gravel. This is rather different from the case used to derive most empirical bedload relationships which assume an unlimited supply of sand moving over a bed.

9.5 Conclusions

Model

This project has produced a robust CFD numerical model capable of describing the velocity flows within complex areas of morphology such as found in aggregate extraction areas. These flows fields have been utilized within numerical suspended sediment models to predict re-suspension and deposition of sediment within the model domain.

The results of the CFD simulations clearly indicate a mechanism for retention of sands by dredge tracks, arising from the reduced bed shear stress within the tracks compared to the surrounding higher flat regions. Three points of worthy of note:

- This mechanism appears to be relatively insensitive to track orientation with respect to the main tidal axis.
• The mechanism is size selective, preferentially trapping coarser particles.

• It is plausible that a similar mechanism will hold for particles transported as bed-load, as well as the suspended load case considered here.

Dredging Scenarios

The preliminary practical implications of this work are that suspended sediment transport associated with the dredge track is relatively insensitive to the orientation of the dredge tracks compared to the principal tidal axis. Some aspects of dredge track morphology are probably more significant, and in particular, isolated dredge tracks with a large depth/width ratio (i.e. deep and narrow) are more likely to accumulate sediment than tracks with a small depth/width ratio (i.e. shallow and wide).

One of the potential main deliverables from this project was practical "Best Practice Guidance" for the marine aggregate extraction industry on the management of dredging lanes to minimize environmental impact, if the work’s findings were found to be significant. A limited number of scenarios have been investigated to date, so that it has been considered inappropriate to create such guidance notes at this stage.

9.5.1 Implications For Managing The Impact of Dredging On Sediment Transport

With regard to the affects on suspended sediment transport, our results indicate that the morphology of dredge tracks is probably more important than their orientation relative to the tide. Thus, shallow wider tracks would be expected to have less impact on suspended sediment transport than deep narrow tracks. There is little evidence to indicate
that changing current dredging practice by altering the orientation of dredge tracks will reduce sediment transport and hence environmental impact.

Clearly, these conclusions have implications for the management of dredge sites, especially regarding their potential ‘recovery’. In terms of recovery of dredge tracks after cessation of dredging, the results indicate that shallow, wide tracks are more likely to be stripped of mobile sand across their surface and thus return to a bed-armored condition. However, areas with deep, narrow and isolated dredge tracks are more likely to accumulate sand which could remain there.

These initial results might be applied, with caveats, to the broader scale, because the overall depressions with low aspect ratios (i.e. shallow and wide) might be less likely to retain sediment transported by suspension. However, further work would be required to test this hypothesis, because, for example, the precise sedimentary processes likely to be operating on a wide linear depression (e.g. 5 m deep and 300 m wide) will be different to those operating at a smaller scale (e.g. 0.5 m deep and 3 m wide).

Further, the precise bed irregularities present in such contrasting features are likely to have different significance with respect to the potential for biological (re-) colonization of the bed. A CFD approach such as the one used here could be used to help explore mitigation measures regarding dredging patterns and to examine the sedimentary consequences of measures performed following the cessation of dredging (e.g. ‘raking’ of the dredge lane).
9.5.2 Future Work

- Further runs of the current model will allow more complex and hence more realistic simulations to be undertaken, and could include development of the 2-D CFD model into a full 3-D model, ultimately to allow real bathymetry to be simulated, including irregular-shaped dredge tracks, static dredge pits and irregularly shaped dredging areas.

- Generalized inferences on suspended sediment transport have been generated by this project, and, together with bedload transport, these need to be converted into implications for the benthos and long-term benthic 'recovery'. A potential first stage in this approach is to use the framework developed under the Cefas 'rehabilitation' projects (Boyd et al. [14], Cooper et al. [30]), which is focused on the 'recovery' of benthic biology.

- The results should be verified using a wider search of suitably high-resolution side-scan sonar imagery, swath bathymetry, photographic evidence and particle size data from within dredge tracks. If no such data exists, a modest fieldwork programme could be designed to be bolted onto existing monitoring and research programs.

- At present, the CFD software only includes pick-up and deposition of sediment and suspended transport on tidal timescales. Further modifications could include bedload transport and scenarios involving storm-generated bed shear stress through the action of waves and from storm-generated changes to residual transport direc-
tions. The bedload transport addition is especially important because many of the hydrographic regimes around dredge sites are in a zone where sediment transport is controlled by both suspended and bedload components.

- The model can be used to test various scenarios of active management of dredging sites after cessation of dredging in order to accelerate restoration of the site in terms of sediment transport, sediment coverage, morphology and, given integration with a suitable benthic biological module, the biology.

Figure 9.1: Hypothetical distribution of macro-bedforms around an individual dredging lane. Net bed sediment transport is towards the left. A-A' and B-B' are sections across the dredging lane and show the accumulation of individual dredge tracks. The red line shows the bed profile immediately after dredging ceases, green a short period later (months) and the orange lines a long period later (years).
Figure 9.2: The 2-D curvilinear (non-orthogonal) mesh (Horizontal scale in metres).
CHAPTER 9. SIMULATIONS FOR SEDIMENT TRANSPORT IN DREDGES

Figure 9.3: Numerical (crossed line) and analytic (solid line) profiles of horizontal shear stress \( \left( \frac{m^{-2}}{s^{-2}} \right) \), with bed roughness \( z_0 = 0.01 \) m. Crosses show position of numerical grid points.

Figure 9.4: Numerical (crossed line) and analytic (solid line), profiles of mean stream-wise velocity \( (ms^{-1}) \), with bed roughness \( z_0 = 0.01 \) m. Crosses show position of numerical grid points.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Figure 9.5: Numerical versus analytic solution for suspended sediment profiles for 100 and 200 μm particles (vertical scale expanded to emphasise nearbed values). Solid lines are the analytic solution and the stars the numerical solution.
Figure 9.6: Vertical profiles of mean velocity (m s^-1) over a tidal cycle. a) from Davies & Jones [37]. b) calculated by present model. Note, profiles from the present and previous study were not plotted at exactly the same phase through the tidal cycle as the phase of outputs profiles was not specified in Davies and Jones. The results indicate that the two sets of calculations are consistent with one another.
Figure 9.7: Vertical profiles of shear stress (Nm$^{-2}$) over a tidal cycle, a) from Davies & Jones [37], b) Calculated by present model. Note, profiles from the present and previous study were not plotted at exactly the same phase through the tidal cycle as the phase of outputs profiles was not specified in Davies and Jones. The results indicate that the two sets of calculations are consistent with one another.
Figure 9.8: Vertical profiles of turbulent kinetic energy ($m^2s^{-2}$) over a tidal cycle, a) from Davies & Jones [37], b) Calculated by present model. Note, profiles from the present and previous study were not plotted at exactly the same phase through the tidal cycle as the phase of outputs profiles was not specified in Davies and Jones. The results aim to show the consistency of the two sets of calculations rather than to make a quantitative comparison.
Figure 9.9: Vertical profiles of turbulent viscosity \( (m^2 s^{-1}) \) over a tidal cycle, a) from Davies & Jones [37], b) Calculated by present model. Note, profiles from the present and previous study were not plotted at exactly the same phase through the tidal cycle as the phase of outputs profiles was not specified in Davies and Jones. The results aim to show the consistency of the two sets of calculations rather than to make a quantitative comparison.
Figure 9.10: Nearbed bed cross-track velocity ($V$) for $10^\circ$ orientation

Figure 9.11: Nearbed bed cross-track velocity ($V$) for $45^\circ$ orientation
Figure 9.12: Nearbed bed cross-track velocity ($V$) for 90° orientation

Figure 9.13: Turbulent Kinetic Energy (TKE) for track orientation 10° - U tidal velocity oscillates back and forth into and out of the page with the cross-track velocity ($V$) going across the page
Figure 9.14: Turbulent Kinetic Energy (TKE) for track orientation 45°. U tidal velocity oscillates back and forth into and out of the page with the cross-track velocity (V) going across the page.

Figure 9.15: Turbulent Kinetic Energy (TKE) for track orientation 90°. U tidal velocity oscillates back and forth into and out of the page with the cross-track velocity (V) going across the page.
Figure 9.16: Bed shear distribution. Track orientation $0^\circ$ 

Figure 9.17: Bed shear distribution. Track orientation $10^\circ$
CHAPTER 9. SIMULATIONS FOR SEDIMENT TRANSPORT IN DREDGES

Figure 9.18: Bed shear distribution. Track orientation 45°

Figure 9.19: Bed shear distribution. Track orientation 90°
CHAPTER 9. SIMULATIONS FOR SEDIMENT TRANSPORT IN DREDGES

Figure 9.20: Log$_{10}$ sediment concentrations for 10° and 100 μm.

Figure 9.21: Log$_{10}$ sediment concentrations for 10° and 250 μm.
Figure 9.22: Log$_{10}$ sediment concentrations for 10° and 500 μm.

Figure 9.23: Log$_{10}$ sediment concentrations for 10° and 100 μm.
Figure 9.24: \( \log_{10} \) sediment concentrations for 45° and 250 \( \mu \)m.

Figure 9.25: \( \log_{10} \) sediment concentrations for 45° and 500 \( \mu \)m.
Figure 9.26: Log \(_10\) sediment concentrations for 90° and 100 µm.

Figure 9.27: Log \(_10\) sediment concentrations for 90° and 250 µm.
Figure 9.28: $\log_{10}$ sediment concentrations for 90° and 500 µm.

Figure 9.29: Dredge tracks observed using Side-scan sonar from Area 107 in the Outer Wash.
CHAPTER 9. SIMULATIONS FOR SEDIMENT TRANSPORT IN DREDGES

Figure 9.30: Side-scan sonar image from Hastings Single bank showing intensive tide parallel dredging with Sub Area Y of the Licence area in 1999.

Figure 9.31: Side-scan sonar image from Owers Bank off Shoreham in 2004 showing new and old dredge tracks. The new dredge track (line stretching centre left curving to east) also shows "lips" either side of the dredge tracks due to the draghead.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Figure 9.32: Swath Bathymetry showing dredge tracks and dredge pits on the Owers Bank off Shoreham in 2004.

Figure 9.33: Snapshot of bed sediment distribution for scenario 1 for 100 μm sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).
Figure 9.34: Snapshot of bed sediment distribution for scenario 1 for 250 µm sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).

Figure 9.35: Snapshot of bed sediment distribution for scenario 1 for 500 µm sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).
Figure 9.36: Time series of bed retention for scenario 1 for various particle sizes.

Figure 9.37: Snapshot of bed sediment distribution for scenario 2 for 100 μm sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).
CHAPTER 9. SIMULATIONS FOR SEDIMENT TRANSPORT IN DREDGES

Figure 9.38: Snapshot of bed sediment distribution for scenario 2 for 250 μm sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).

Figure 9.39: Snapshot of bed sediment distribution for scenario 2 for 500 μm sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).
CHAPTER 9. SIMULATIONS FOR SEDIMENT TRANSPORT IN DREDGES

Figure 9.40: Time series of bed retention for scenario 2 for various particle sizes:

Figure 9.41: Snapshot of bed sediment distribution for scenario 3 for 100 μm sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).
Figure 9.42: Snapshot of bed sediment distribution for scenario 3 for 250 μm sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).

Figure 9.43: Snapshot of bed sediment distribution for scenario 3 for 500 μm sand (blue line shows the seabed elevation and the black line percentage retention from the CFD model).
**Figure 9.44:** Time series of bed retention for scenario 3 for various particle sizes.

**Figure 9.45:** Percentage retention for all scenarios and particle sizes

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
CHAPTER 10

CONCLUSION

This work has primarily been focused on designing and building a visualization environment to aid scientists in investigating problems of interest related to computational fluid dynamics and sediment transport. The RAVE visualization system not only provides 3D immersive viewing capabilities, but also an environment for analyzing computational models. Scientists can use the environment developed here to compare model results with actual data and therefore, use this information to adjust model design and parameters for a more accurate depiction. Furthermore, once these more accurate models have been developed, our visualization environment may be used for predictive analysis and recommendations. The following figures show visualization results using a desktop computer equipped with a 3.0 GHz Pentium 4 processor and 1.0 GB of memory. Figures 10.1 - 10.7 show aerodynamic model results of a rectangular object placed in a constant velocity wind field. The yellow spheres represent air molecules in the wind field and how their movement would be affected by the pressure gradients introduced by the presence of the rectangular object. The figures also show cross-sectional views of the velocity wind field at selected locations along the horizontal axis. Additional viewing perspectives can be directed by the user. Also, other user-selectable display techniques, such as flow vectors, stream lines, etc., may be used to create more views. The figure 10.7 shows the visualization result in the CAVE environment.
Figure 10.1: Wind turbulence at 90° view angle.

Figure 10.2: Wind turbulence at 45° view angle.
Figure 10.3: Wind turbulence around an object at 45° view angle.

Figure 10.4: Wind turbulence around an object at 180° view angle.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
CHAPTER 10. CONCLUSION

Figure 10.5: Wind turbulence at the end of an object at 135° view angle.

Figure 10.6: Wind turbulence at the end of an object at 135° view angle.
Figures 10.8 - 10.15 depict results of a sand ripple bed sediment transport model. In this case, a near-shore ocean profile is mathematically generated with virtual sand particles randomly deposited over the bottom surface. A time-dependent ocean wave velocity field was applied over the domain. Each figure shows how sediment transport is effected by the sediment transport model. Animation is a key factor that cannot be represented here, but these images indicate how eddy forces are generated by the current shape of the ocean floor. Also, only a fraction of the domain has sand deposits due to a limited amount of computer memory. However, current results allow for the 3D simulation of approximately 400,000 sand particles using only desktop computing power. The figure 10.15 shows the visualization result in the CAVE environment.
Figure 10.8: Ripple bed sand particles at rest.

Figure 10.9: Ripple bed sand particles when lefted.
Figure 10.10: Ripple bed sand particles when lefted and suspended.

Figure 10.11: Ripple bed sand particles when lefted and suspended at close-up view.
Figure 10.12: Ripple bed sand particles when falling.

Figure 10.13: Ripple bed sand particles when falling and deposited.
Figure 10.14: Ripple bed sand particles when falling and deposited near the bottom surface.

Figure 10.15: Sediment transport results on the RAVE.
The next ten figures show a few of the details from applying the sediment transport model used in the previous example to one scenario from Chapter 9 where a dredge channel is located in the presence of a wave velocity field. Figure 10.16 shows a global view of the entire water column. In this example, the water is 60 feet deep, the dredge channel approximately 20 feet wide, and the wave height approximately 2-3 feet. Figures 10.17 shows the grid generated for this example.

Figures 10.18 - 10.25 show sediment transport results as a function of time where the direction of ocean wave impact is horizontally perpendicular to the dredge channel. Additional results can be simulated incorporating results from Chapter 9 where the angle between the dredge channel and the velocity field is less than 90°.
Figure 10.17: Grid generation.

Figure 10.18: Dredge sand particles at rest.
Figure 10.19: Dredge sand particles at leading wave.

Figure 10.20: Dredge sand particles at trailing wave.
Figures 10.21 and 10.22 show the flow vectors at two selected moments in time.
The figure 10.23 shows the domain with the normal vectors displayed which are one component needed to compute collisions of the sand particles with the ocean floor. Finally, the figures 10.24 and 10.25 show the visualization results in the CAVE environment.
Figure 10.24: Sand particle results on the RAVE.

Figure 10.25: Flow vector results on the RAVE.
10.1 Applications and Future Directions

The industrial relevance of this project could be for predicting the behavior of coastal defense, navigation, and waste disposal, fisheries, oil and gas extraction, power generation (from waves and tides), search and rescue operations, response to oil spills, other pollution incidents and mine burial. In addition, there is long term concern about the impacts of climatic change and sea level rise for coastal defense. The recognition of the need to manage coastal and shelf seas requires a major extension of our existing predictive capability to take account of the complex interactions that take place between motions in the sea and sediment transport.
APPENDIX A

AN ANALYTICAL SOLUTION FOR MIXING-LENGTH MODEL

The $U$ momentum balance at steady state assuming no stream-wise change is simple a balance between the pressure forcing and the vertical turbulent diffusion.

\[
-\frac{1}{\rho} \frac{\partial P}{\partial x} + \frac{\partial}{\partial z} \left( \nu_T \frac{\partial U}{\partial z} \right) = 0
\]  \hspace{1cm} (A.1)

Here $U$ is the stream-wise velocity, is the eddy viscosity, $P$ is the applied pressure force, $\rho$ is the water density, with $z$ is measured positive upward from the bed and $x$ represent the positive stream-wise direction. For a constant pressure gradient $P_x = \frac{\partial P}{\partial x}$ the momentum balance can be immediately integrated from $z = z_0$ to $z = h$, where $h$ is the water depth, to give

\[
\nu_T \frac{\partial U}{\partial y} = -h \frac{P_x}{\rho} \frac{1}{h} (1 - z/h)
\]  \hspace{1cm} (A.2)

The left hand side of the above is the definition of (minus) the horizontal shear stress $\tau$ and this relation shows that the stress distribution is must be a linear function of height, decreasing from a maximum at the bed independent of any turbulence viscosity. The maximum value at the bed is given as a function of the applied pressure gradient and water depth as

\[
\tau_b = h |P_x| / \rho
\]  \hspace{1cm} (A.3)
When the transport terms are neglected in the turbulent kinetic energy equation, the model resorts to a mixing length formulation. In this case, the turbulent viscosity is given by

$$\nu_T = l^2 \left| \frac{dU}{dz} \right|$$  \hspace{1cm} (A.4)

where $l$ is the turbulent length scale. A reasonable assumption for the turbulent length scale near a solid boundary is

$$l = \kappa z$$  \hspace{1cm} (A.5)

where $\kappa = 0.4$ is the von-Karman constant. Assuming this relationships for $l$ holds at all levels and substituting into (A.3), and (A.4) into (A.2) enables a solution to be found for the velocity profile of the form

$$U(z) = \sqrt{\frac{4|P_x|h}{\kappa^2 \rho}} \left( \sqrt{1 - z/h} - \sqrt{1 - z_0/h} - \tanh^{-1} \sqrt{1 - z/h} + \tanh^{-1} \sqrt{1 - z_0/h} \right)$$  \hspace{1cm} (A.6)

This satisfies a no-slip boundary condition at $z = z_0$ and $U'(z) = 0$ at $z = h$. The value of $z_0$ is set depending on the assumed roughness of the bed. The relationship (A.5) is unlikely to hold right up to the free surface so the resulting expression for the velocity profile (A.6) is not necessarily the most physically realistic that can be obtained. Nevertheless, as an exact solution with which to compare against model predictions, it provides a useful benchmark. The mixing length formulation is equivalent to assuming local equilibrium of turbulence in which case the turbulent kinetic energy and horizontal shear stress $\tau$ are related by

$$k = \rho^{-1} C_{\mu}^{-1/2} |\tau|$$  \hspace{1cm} (A.7)
where $C_\mu = 0.09$ is a standard turbulent model constant. For suspended sediments in the steady state case, the vertical balance

$$\frac{d}{dz} \left( w_f C + v_T \frac{dC}{dz} \right) = 0$$

(A.8)

is easily solved for constant eddy viscosity to provide an exact solution to compare with numerical results. A solution satisfying $C(0) = 1$ and $C(h) = 0$ is

$$C(z) = \frac{1 - \exp[w_f(h-z)/v_T]}{1 - \exp[w_f h/v_T]}$$

(A.9)
The results produced by the code EUAC are shown in the following output. It is important to observe that the font and spacing allow only a 100 character display.

```
imax = 17  GRID POINTS
fm = 0.400000  FREE STREAM MACH NUMBER
niter = 600  NUMBER OF ITERATIONS
gm = 1.400000  GAMMA
ts = 0.9700000  CFL NUMBER
uf = 0.3937496  FREE STREAM VELOCITY
rf = 1.293984  FREE STREAM DENSITY
pf = 0.8956145  FREE STREAM PRESSURE

INITIAL CONDITION

<table>
<thead>
<tr>
<th>i</th>
<th>x</th>
<th>ar</th>
<th>u(1)</th>
<th>u(2)</th>
<th>u(3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1.0000E+00</td>
<td>3.1416E+00</td>
<td>4.0652E+00</td>
<td>1.6007E+00</td>
<td>7.3493E+00</td>
</tr>
<tr>
<td>2</td>
<td>-8.7500E-01</td>
<td>3.0939E+00</td>
<td>4.0035E+00</td>
<td>1.5764E+00</td>
<td>7.2378E+00</td>
</tr>
<tr>
<td>3</td>
<td>-7.5000E-01</td>
<td>2.9603E+00</td>
<td>3.9305E+00</td>
<td>1.5083E+00</td>
<td>6.9251E+00</td>
</tr>
<tr>
<td>4</td>
<td>-6.2500E-01</td>
<td>2.7657E+00</td>
<td>3.5788E+00</td>
<td>1.4091E+00</td>
<td>6.4699E+00</td>
</tr>
<tr>
<td>5</td>
<td>-5.0000E-01</td>
<td>2.5447E+00</td>
<td>3.2928E+00</td>
<td>1.2965E+00</td>
<td>5.9529E+00</td>
</tr>
<tr>
<td>6</td>
<td>-3.7500E-01</td>
<td>2.3329E+00</td>
<td>3.0187E+00</td>
<td>1.1886E+00</td>
<td>5.4574E+00</td>
</tr>
<tr>
<td>7</td>
<td>-2.5000E-01</td>
<td>2.1605E+00</td>
<td>2.7957E+00</td>
<td>1.1008E+00</td>
<td>5.0542E+00</td>
</tr>
<tr>
<td>8</td>
<td>-1.2500E-01</td>
<td>2.0491E+00</td>
<td>2.6515E+00</td>
<td>1.0440E+00</td>
<td>4.7935E+00</td>
</tr>
<tr>
<td>9</td>
<td>0.0000E+00</td>
<td>2.0105E+00</td>
<td>2.6017E+00</td>
<td>1.0244E+00</td>
<td>4.7035E+00</td>
</tr>
<tr>
<td>10</td>
<td>1.2500E-01</td>
<td>2.0491E+00</td>
<td>2.6515E+00</td>
<td>1.0440E+00</td>
<td>4.7935E+00</td>
</tr>
<tr>
<td>11</td>
<td>2.5000E-01</td>
<td>2.1605E+00</td>
<td>2.7957E+00</td>
<td>1.1008E+00</td>
<td>5.0542E+00</td>
</tr>
<tr>
<td>12</td>
<td>3.7500E-01</td>
<td>2.3329E+00</td>
<td>3.0187E+00</td>
<td>1.1886E+00</td>
<td>5.4574E+00</td>
</tr>
<tr>
<td>13</td>
<td>5.0000E-01</td>
<td>2.5447E+00</td>
<td>3.2928E+00</td>
<td>1.2965E+00</td>
<td>5.9529E+00</td>
</tr>
<tr>
<td>14</td>
<td>6.2500E-01</td>
<td>2.7657E+00</td>
<td>3.5788E+00</td>
<td>1.4091E+00</td>
<td>6.4699E+00</td>
</tr>
<tr>
<td>15</td>
<td>7.5000E-01</td>
<td>2.9603E+00</td>
<td>3.9305E+00</td>
<td>1.5083E+00</td>
<td>6.9251E+00</td>
</tr>
<tr>
<td>16</td>
<td>8.7500E-01</td>
<td>3.0939E+00</td>
<td>4.0035E+00</td>
<td>1.5764E+00</td>
<td>7.2378E+00</td>
</tr>
<tr>
<td>17</td>
<td>1.0000E+00</td>
<td>3.1416E+00</td>
<td>4.0652E+00</td>
<td>1.6007E+00</td>
<td>7.3493E+00</td>
</tr>
</tbody>
</table>

CONVERGED SOLUTION

<table>
<thead>
<tr>
<th>i</th>
<th>x</th>
<th>ar</th>
<th>u(1)</th>
<th>u(2)</th>
<th>u(3)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
### APPENDIX B. COMPUTER RESULTS

<table>
<thead>
<tr>
<th>i</th>
<th>a(i)</th>
<th>Mach</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.1416E+00</td>
<td>4.0000E-01</td>
</tr>
<tr>
<td>2</td>
<td>3.0939E+00</td>
<td>4.0771E-01</td>
</tr>
<tr>
<td>3</td>
<td>2.9603E+00</td>
<td>4.3129E-01</td>
</tr>
<tr>
<td>4</td>
<td>2.7657E+00</td>
<td>4.7197E-01</td>
</tr>
<tr>
<td>5</td>
<td>2.5447E+00</td>
<td>5.3142E-01</td>
</tr>
<tr>
<td>6</td>
<td>2.3329E+00</td>
<td>6.1095E-01</td>
</tr>
<tr>
<td>7</td>
<td>2.1605E+00</td>
<td>7.0938E-01</td>
</tr>
<tr>
<td>8</td>
<td>2.0491E+00</td>
<td>8.1594E-01</td>
</tr>
<tr>
<td>9</td>
<td>2.0106E+00</td>
<td>8.7728E-01</td>
</tr>
<tr>
<td>10</td>
<td>2.0491E+00</td>
<td>8.1594E-01</td>
</tr>
<tr>
<td>11</td>
<td>2.1605E+00</td>
<td>7.0938E-01</td>
</tr>
<tr>
<td>12</td>
<td>2.3329E+00</td>
<td>6.1095E-01</td>
</tr>
<tr>
<td>13</td>
<td>2.5447E+00</td>
<td>5.3142E-01</td>
</tr>
<tr>
<td>14</td>
<td>2.7657E+00</td>
<td>4.7197E-01</td>
</tr>
<tr>
<td>15</td>
<td>2.9603E+00</td>
<td>4.3129E-01</td>
</tr>
<tr>
<td>16</td>
<td>3.0939E+00</td>
<td>4.0771E-01</td>
</tr>
<tr>
<td>17</td>
<td>3.1416E+00</td>
<td>4.0000E-01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>i</th>
<th>r(1)</th>
<th>r(2)</th>
<th>r(3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0000E+00</td>
<td>6.5565E-07</td>
<td>0.0000E+00</td>
</tr>
<tr>
<td>2</td>
<td>1.9073E-06</td>
<td>0.0000E+00</td>
<td>3.8147E-06</td>
</tr>
<tr>
<td>3</td>
<td>0.0000E+00</td>
<td>-2.0266E-06</td>
<td>0.0000E+00</td>
</tr>
<tr>
<td>4</td>
<td>3.8147E-06</td>
<td>-8.3447E-07</td>
<td>7.6294E-06</td>
</tr>
<tr>
<td>5</td>
<td>1.9073E-06</td>
<td>2.1458E-06</td>
<td>3.8147E-06</td>
</tr>
<tr>
<td>6</td>
<td>-3.8147E-06</td>
<td>-3.2187E-07</td>
<td>-1.1444E-05</td>
</tr>
<tr>
<td>7</td>
<td>9.5367E-07</td>
<td>5.0068E-06</td>
<td>3.8147E-06</td>
</tr>
<tr>
<td>8</td>
<td>-4.7684E-06</td>
<td>-6.0648E-06</td>
<td>-7.6294E-06</td>
</tr>
<tr>
<td>9</td>
<td>1.9073E-06</td>
<td>3.7253E-07</td>
<td>3.8147E-06</td>
</tr>
<tr>
<td>10</td>
<td>-3.8147E-06</td>
<td>-5.0068E-06</td>
<td>-1.5259E-05</td>
</tr>
</tbody>
</table>

*Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.*
<table>
<thead>
<tr>
<th></th>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>2.8610E-06</td>
<td>4.8876E-06</td>
<td>1.9073E-05</td>
</tr>
<tr>
<td>12</td>
<td>3.8147E-06</td>
<td>3.2187E-06</td>
<td>3.8147E-06</td>
</tr>
<tr>
<td>13</td>
<td>-9.5367E-07</td>
<td>7.1528E-07</td>
<td>-7.6294E-06</td>
</tr>
<tr>
<td>14</td>
<td>-9.5367E-07</td>
<td>-1.6689E-06</td>
<td>0.0000E+00</td>
</tr>
<tr>
<td>15</td>
<td>-9.5367E-07</td>
<td>2.0862E-06</td>
<td>1.1444E-05</td>
</tr>
<tr>
<td>16</td>
<td>6.6757E-06</td>
<td>-5.6624E-07</td>
<td>0.0000E+00</td>
</tr>
<tr>
<td>17</td>
<td>0.0000E+00</td>
<td>0.0000E+00</td>
<td>0.0000E+00</td>
</tr>
</tbody>
</table>


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.


