ABSTRACT

Title of Thesis: FLUID AND PARTICLE DYNAMICS IN AN AEROSOL VIRTUAL IMPACTOR

Marwan L. Charrouf, Master of Science, 2004

Thesis Directed by: Professor Richard V. Calabrese
Department of Chemical Engineering

The collection and characterization of chemical and biological aerosols is essential to many areas of particle research such as toxicological studies, pollutant sampling, and biohazard assessment. This work presents the simulation of a low cutpoint, high volume aerosol sampling device known as the "virtual impactor". A steady state, three dimensional RANS type calculation is done using the FLUENT^{TM} computational fluid dynamics code to predict the turbulent flow field inside the device. Particle collection efficiency and wall losses are then obtained by solving the particle equation of motion governed by drag for mono-dispersed samples of spherical particles in the 0.1-0.4 μm diameter range.

Predictions of the mean fluid velocity field with the incompressible Reynolds stress model and the compressible k-epsilon turbulence model are relied upon for conducting particle tracking calculations. FORTRAN 90 computer code is developed to solve the particle equation of motion using an implicit second order accurate time integration scheme. In addition, a multi-variate, scattered point interpolation method is implemented to obtain the fluid velocity at a position away from an Eulerian mesh point.

It is found that “adaptive” drag law models are necessary to correctly account for slip and compressibility. The results indicate the trends observed in the experiments, and a 50% cutpoint diameter between 0.250 and 0.275 μm. Recommendations for improved modeling in future work are made.

FLUID AND PARTICLE DYNAMICS IN AN AEROSOL VIRTUAL IMPACTOR

by

Marwan L. Charrouf

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Science
2004

Advisory Committee:

Professor Richard V. Calabrese, Chair
Professor Sheryl H. Ehrman
Professor Panagiotis Dimitrakopoulos
Professor James Gentry
Dr. M. B. (Arun) Ranade

Acknowledgments

My deepest gratitude goes to my thesis adviser Dr. Richard V. Calabrese, who warmly welcomed me into his research group, and guided me in the teachings of fluid mechanics and turbulence. His encouragement to switch to Linux, and his generosity in providing top of the line computers to handle the multitude of computationally intensive applications performed for this research is greatly appreciated. I have to thank Dr. Arun Ranade for getting me started on the virtual impactor literature, and for involving me in the ongoing experimental side of this project.

I would also like to thank my colleague Karl Kevala for his support during my early days of learning FLUENT™, and for the valuable conversations we had about simulations and fluid flow in general.

Much appreciation is due to the members of my advisory committee; thank you for taking the time to critique my work, and for your patience.

Last but not least, my parents who are the reason I am what I am today. I am eternally indebted to your endless kindness and noble sacrifices. Thank you for understanding my quest for knowledge, and for holding our family together in times of hardship.

Table of Contents

Acknowledgments ii

List of Tables vii

List of Figures viii

List of Symbols xvi

1. Introduction

1.1 Problem Statement 1

1.2 Overview of Virtual Impactors 3

1.3 Contributions 5

1.4 Organization of the Thesis 6

2. Background and Related Work

2.1 Experimental Investigation of Virtual Impactors 8

2.2 Theoretical and Numerical Studies of Virtual Impactors 12

2.3 Approaches to Two Phase Turbulent Flows 14

3. Simulation Components

3.1 Compressibility 19

3.2 Device Geometry 19

3.3 Computational Meshes 21

3.4 Boundary and Operating Conditions 24

3.5 Discrete Phase Properties 26

4. Governing Equations

4.1 Fluid Flow 27

4.1.1 Reynolds Averaged Navier Stokes Equations 29

4.1.2 The Ҡ-∈ Closure 30

4.1.3 The RSM Closure 34

4.1.4 The Numerical Scheme 36

4.1.4a Discretization 37

4.1.4b Wall Functions 39

4.2 Particulate Flow 42

4.2.1 Particle Equation of Motion 42

4.2.2 The Drag Coefficient 44

4.2.3 Random Walk Model: Stochastic Tracking 51

4.2.4 Numerical Solution of the Particle Equation 54

4.2.4a FLUENT'S Implementation 54

4.2.4b AeroTrack: The Developed Code 55

4.2.4b.1 Trapezoidal Scheme 55

4.2.4b.2 Multi-Variate Scattered Interpolation 57

4.2.4b.2-i Motivation 57

4.2.4b.2-ii Theory 59

4.2.4b.2-iii Proposed Algorithm 63

4.2.4b.3 Adaptive Drag Coefficient 70

4.2.4b.4 AeroTrack Task Flow 70

5. Discussion of Numerical Approaches

5.1 Convergence of the Fluid Flow Solution 73

5.2 Accuracy of the Particle Tracking Algorithms 82

5.2.1 Time Stepping 83

5.2.2 Number of Particles 87

5.2.3 Interpolation Tests 92

5.2.3a Bi-Variate Functions 92

5.2.3b Tri-Variate Functions 102

5.2.3c Prescribed Velocity Field 104

5.2.3d Virtual Impactor Simulation Data 106

5.3 Summary 110

6. Simulation Results

6.1 Properties of the Fluid Flow Solution 111

6.1.1 Ƙ-ϵ vs. RSM 112

6.1.2 Incompressible vs. Compressible Flow 124

6.1.3 Coarse vs. Fine Grid 131

6.1.4 1st vs. 2nd Order Discretization 136

6.1.5 Motion of Fluid Particles 145

6.1.6 Vorticity 151

6.1.7 Summary 155

6.2 The Discrete Phase Results 156

6.2.1 Incompressible Flow Field 157

6.2.2 Compressible Flow Field 168

6.2.3 The Drag Coefficient Revisited 180

6.2.3a More Realistic Models for the Drag Coefficient .. 185

6.2.4 Incompressible vs. Compressible Flow 189

6.2.5 Comparison with Experiment 198

6.2.6 Summary 204

7. Conclusions and Recommendations

7.1 Summary 207

7.2 Conclusions 208

7.3 Recommendations 210

A. Solution of the Particle ODE 212

B. Solution of the Least Squares in 2D 215

C. Solution of the Least Squares in 3D 217

References 221

List of Tables

3.1 Computational mesh statistics 22

4.1 Classification of flow regimes from [1] 45

5.1 Listing of virtual impactor simulations 72

5.2 Mean velocity difference between “converged” and “pre-converged” solutions 82

5.3 Mean and standard deviation of collection efficiency and wall losses for 0.1 μm particles using three different sample sizes, simulation 1 FLUENT^{TM} stochastic tracking with Stokes-Cunningham 91

5.4 Interpolation errors for bivariate test functions 99

5.5 Error Norms for modified Shepard Algorithms on Franke's node set. *Reported in [47] 100

5.6 Interpolation errors for trivariate test functions 103

5.7 Absolute interpolation errors for numerical data, 185 random cells, simulation 4 107

5.8 Relative interpolation errors for numerical data, 185 random cells, simulation 4 108

5.9 RMS of relative interpolation error for different Renka parameters, 185 random cells, simulation 4 109

6.1 Summary of drag coefficient models 157

List of Figures

1.1 Overview of a virtual impactor 3

1.2 Actual and ideal collection efficiency for a virtual impactor 5

2.1 Semi-empirical dependence of $St_{50}^{1/2}$ on Q_m/Q_T from [15] 10

2.2 Influential nozzle design parameters 11

2.3 Map of regimes of interaction between particles and turbulence from [10] 16

3.1 Dimensions of simulated virtual impactor (XY plane) 20

3.2 3D coarse mesh 22

3.3 Nozzle cross-section (coarse mesh Δ=W/50) 23

3.4 Nozzle cross-section (finer mesh Δ=W/100) 24

4.1 Drag coefficients for a 0.1 μm particle, Stokes = $24/Re_p$, Oseen = eqn. 4.2.12, Stokes-Cunningham = eqn. 4.2.6, Compressible = eqn. 4.2.9 (insert shows entire y-axis scale) 50

4.2 Drag coefficients for a 0.35 μm particle, Stokes = $24/Re_p$, Oseen = eqn. 4.2.12, Stokes-Cunningham = eqn. 4.2.6, Compressible = eqn. 4.2.9 (insert shows entire y-axis scale) 51

4.3 Algorithm for computing dynamic radii 64

4.4 2x2 cell structure with 7 data nodes & a query point P 68

4.5 Cell grid on top of computational mesh points 69

4.6 Flowchart of AeroTrack execution 71

5.1 Convergence of simulation 1 74

5.2 Schematic of “Virtual Impaction Region” 76

5.3 Contours of the norm of velocity difference between the final iteration sets (simulation 1) 76

5.4 Convergence of simulation 2 77

5.5 Contours of the norm of velocity difference between the final iteration sets (simulation 2) 77

5.6 Convergence of simulation 3 78

5.7 Contours of the norm of velocity difference between the final iteration sets (simulation 3) 78

5.8 Convergence of simulation 4 79

5.9 Contours of the norm of velocity difference between the final iteration sets (simulation 4) 80

5.10 Convergence of simulation 5 81

5.11 Contours of the norm of velocity difference between the final iteration sets (simulation 5) 81

5.12 Particle paths using $\Delta L=1.0e^{-05}$ (red), $1.0e^{-06}$ (blue), and $1.0e^{-07}$ m (magenta) {simulation 4} 84

5.13 Particle paths using $\Delta L=1.0e^{-05}$ (red), $1.0e^{-06}$ (blue), and $1.0e^{-07}$ m (magenta) - zoom level 1 - {simulation 4} 85

5.14 Particle paths using $\Delta L=1.0e^{-05}$ (red), $1.0e^{-06}$ (blue), and $1.0e^{-07}$ m (magenta) - zoom level 2 - {simulation 4} 86

5.15 Particle approaching a wall using $\Delta L=1.0e^{-05}$ (red), $1.0e^{-06}$ (blue), and $1.0e^{-07}$ m (magenta) - zoom level 1 - {simulation 4} 87

5.16 Efficiency and wall losses curve for two sample sizes, simulation 1 FLUENT™ mean tracking with Stokes-Cunningham 89

5.17 Collection efficiency of 0.1 μm particles for three different sample sizes, simulation 1 FLUENT™ stochastic tracking with Stokes-Cunningham drag law 90

5.18 Wall losses for 0.1 μm particles for three different sample sizes, simulation 1 FLUENT™ stochastic tracking with Stokes-Cunningham drag law 87

5.19 33x33 uniform grid (blue) with a set of 100 random nodes (red) 94

5.20 Bivariate test function F_1 (Equation 5.3) 95

5.21 Bivariate test function F_2 (Equation 5.4) 96

5.22 Bivariate test function F_3 (Equation 5.5) 97

5.23 Bivariate test function F_8 (Equation 5.6) 98

5.24 33x33 uniform grid (blue) with Franke's 100 nodes (red) 101

5.25 Interpolation error versus resolution for the interpolation of trivariate functions F_1 (black), F_2 (red), & F_3 (blue). Renka (solid lines), linear (dashed lines) 103

5.26 Interpolation error versus resolution for the interpolation of sinusoidal velocity fields using linear, Renka, cubic, & spline interpolation 105

5.27 Random set of interpolation nodes in a virtual impactor grid 109

6.1 Midplane contours of velocity magnitude (m/s), simulation 1 113

6.2 Midplane contours of velocity magnitude (m/s), simulation 2 114

6.3 Schematic of throat cross sections at y=0, y=-0.5W, y=-W, y=-1.5W, and y=-2W (from top to bottom) 115

6.4 Dimensionless streamwise velocity profile in throat, simulation 1 (black), simulation (red) 115

6.5 Mean velocity vectors (m/s), flow separation, simulation 1 117

6.6 Mean velocity vectors (m/s), flow separation, simulation 2 117

6.7 Streamwise velocity profile at nozzle and beyond, y=-W, y=-1.5W, y=-2W, simulation no. 1 (solid lines), simulation no. 2 (dashed lines) 118

6.8 Schematic of expansion nozzle cross sections at x = 0.75W, x = 1.125W, and x = 1.5W (from left to right) 119

6.9 Cross-stream velocity profile of exiting fluid, simulation 1 120

6.10 Cross-stream velocity profile of exiting fluid, simulation 2 120

6.11 Contours of the norm of velocity difference between the predictions of simulations 1 and 2, scaled by U_{avg} 122

6.12 Absolute pressure as a function of streamwise distance from nozzle at x=0, simulation 2 (insert shows pre-nozzle data) 123

6.13 Absolute pressure as a function of cross-stream distance, midpoint of virtual impaction zone, simulation 2 124

6.14 Midplane contours of velocity magnitude (m/s), simulation 3 126

6.15 Mean velocity vectors (m/s), flow separation, simulation 3 127

6.16 Cross-stream velocity profile of exiting fluid, simulation 3 127

6.17 Streamwise velocity profile at nozzle and beyond, y=-W, y=-1.5W, y=-2W, simulation no. 1 (solid lines), simulation no. 3 (dashed lines) 128

6.18 Dimensionless streamwise velocity profile in throat, simulation 1 (black), simulation 3 (red) 129

6.19 Contours of the norm of velocity difference between the predictions of simulations 1 and 3, scaled by U_{avg} 130

6.20 Contours of absolute pressure difference between the predictions of simulations 1 and 3, scaled by P_{abs} of simulation 3 130

6.21 Dimensionless streamwise velocity profile in throat, simulation 3 (red), simulation 5 (blue) 132

6.22 Streamwise velocity profile at nozzle and beyond, y=-W, y=-1.5W, y=-2W, simulation 3 (solid lines), simulation 5 (dashed lines) 132

6.23 Cross-stream velocity profile of exiting fluid, simulation 5 134

6.24 Absolute pressure as a function of streamwise distance from nozzle at x=0, simulation 5 (insert shows pre-nozzle data) 135

6.25 Absolute pressure as a function of cross-stream distance, midpoint of virtual impaction zone, simulation 5 135

6.26 Midplane contours of turbulent kinetic energy (m^2/s^2), simulation 5 138

6.27 Midplane contours of absolute pressure (Pa), simulation 5 139

6.28 Midplane contours of temperature (°K), simulation 5 140

6.29 Temperature as a function of streamwise distance from nozzle at x=0, simulation 5 141

6.30 Temperature as a function of cross-stream distance, midpoint of virtual impaction zone, simulation 5 141

6.31 Contours of the norm of velocity difference between the predictions of simulations 4 and 5 143

6.32 Contours of absolute pressure difference between the predictions of simulations 4 and 5, scaled by P_{abs} of simulation 4 144

6.33 Contours of temperature difference between the predictions of simulations 4 and 5, scaled by T of simulation 4 144

6.34 Contours of turbulent kinetic energy difference between the predictions of simulations 4 and 5, scaled by K of simulation 4 145

6.35 Fluid pathlines in simulation 1 147

6.36 Fluid pathlines in simulation 2 148

6.37 Fluid pathlines in simulation 5 149

6.38 Normalized residence time for nozzle fluid particles 150

6.39 Contours of vorticity magnitude, midplane, simulation 2 153

6.40 Contours of vorticity magnitude, midplane, simulation 5 154

6.41 Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2 (red), drag law 1 (black) – simulation 1 – mean fluid velocity tracking 158

6.42 Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2 (blue), drag law 1 (black) – simulation 2 – mean fluid velocity tracking 159

6.43 Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2, simulation 1 (red), simulation 2 (blue) – mean fluid velocity tracking 160

6.44 Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2, mean fluid velocity tracking (red), stochastic tracking (green), simulation 1 – Ƙ–ϵ 163

6.45 Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2, mean fluid velocity tracking (blue), stochastic tracking (green), simulation 2 – RSM 163

6.46 Midplane contours of turbulent kinetic energy (m^2/s^2), simulation 1 – incompressible Ƙ–ϵ 164

6.47 Midplane contours of turbulent kinetic energy (m^2/s^2), simulation 2 – incompressible RSM 165

6.48 Contours of the turbulent kinetic energy difference between the predictions of simulations 1 and 2, scaled by ½ U^2_{avg} 168

6.49 Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2, AeroTrack (red), FLUENT (black) – simulation 4 169

6.50 Collection efficiency (solid lines), and wall loss(dashed lines), drag law 2, AeroTrack (red), FLUENT (black) – simulation 5 170

6.51 Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2, AeroTrack, simulation 4 (red), simulation 5 (blue) 171

6.52 0.1 μm particle path, drag law 2, AeroTrack (blue), FLUENT (red), simulation 5 172

6.53 0.25 μm particle path, drag law 2, AeroTrack (blue), FLUENT (red), simulation 5 173

6.54 0.4 μm particle path, drag law 2, AeroTrack (blue), FLUENT (red), simulation5 174

6.55 0.1 μm particle cross-stream velocity, AeroTrack (blue), FLUENT (red), path shown partially in Figure 6.52 175

6.56 0.1 μm particle streamwise velocity, AeroTrack (blue), FLUENT (red), path shown partially in Figure 6.52 176

6.57 0.1 μm particle spanwise velocity, AeroTrack (blue), FLUENT (red), path shown partially in Figure 6.52 176

6.58 0.1 μm particle time steps, AeroTrack (blue), FLUENT (red) path shown partially in Figure 6.52 177

6.59 0.4 μm particle time steps, AeroTrack (blue), FLUENT (red) path shown partially in Figure 6.54 178

6.60 Dimensionless slip velocity as a function of time, 0.1 μm particle, drag law 2 – simulation 4 182

6.61 Dimensionless slip velocity as a function of time, 0.25 μm particle, drag law 2 – simulation 4 182

6.62 Dimensionless slip velocity as a function of time, 0.4 μm particle, drag law 2 – simulation 4 183

6.63 Dimensionless particle descent as a function of time, 0.1, 0.25, and 0.4 μm particle, respectively. drag law 2 – simulation 4 183

6.64 Fluid velocity (black) and 0.25 μm particle velocity (red), drag law 2 – simulation 4 184

6.65 Fluid velocity (black) and 0.4 μm particle velocity (red), drag law 2 – simulation 4 184

6.66 Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2 (red), drag law 3 (blue), drag law 4 (green) – simulation 4 186

6.67 Collection efficiency (solid lines), and wall loss (dashed lines), drag law 4, simulation 4 (red), simulation 5 (blue) 187

6.68 Collection efficiency (solid lines), and wall loss (dashed lines), drag law 3, simulation 4 (orange), simulation 5 (green) 187

6.69 Collection efficiency (solid lines), and wall loss (dashed lines), drag law 4 (red), drag law 3 (blue), simulation 5 189

6.70 0.1 μm particle path, simulation 2 (red), simulation 5 (green), drag law 3 190

6.71 0.25 μm particle path, simulation 2 (red), simulation 5 (green), drag law 3 191

6.72 0.3 μm particle path, simulation 2 (red), simulation 5 (green), drag law 3 192

6.73 Collection efficiency (solid lines), and wall loss (dashed lines), drag law 3, simulation 2 (blue), simulation 5 (red) 194

6.74 Normalized residence time for discrete nozzle particles, drag law 3, simulation 2 (blue), simulation 5 (red) 195

6.75 0.4 μm particle paths from nozzle, simulation 2, drag law 3 197

6.76 0.4 μm particle paths from nozzle, simulation 5, drag law 3 197

6.77 Collection efficiency (solid line), and wall losses (dashed line) vs. $St_{50}^{1/2}$, drag law 4, simulation 5 199

6.78 Collection efficiency (solid lines), and wall losses (dashed lines), Experiment [3] (black), drag law 4 (red), drag law 3 (blue), simulation 5 201

6.79 Effect of Re on the collection efficiency of a slit virtual impactor from Ding and Koutrakis [49] 202

6.80 Effect of Q_m/Q_T ($\equiv$r) on the particle losses in a slit virtual impactor at Q_T = 50 LPM from Ding and Koutrakis [49] 202

List of Symbols

Re: Flow Reynolds number at accelerating nozzle

U_i: Instantaneous fluid velocity component in the i-th direction

$\overline{U_i}$: Reynolds-averaged fluid velocity component in the i-th direction

u: Fluctuating fluid velocity component in the i-th direction

U_p: Particle velocity

D_h: Hydraulic diameter

ν: Kinematic viscosity of fluid

W: Width of accelerating nozzle

W_1: Width of collection nozzle

S: Distance from accelerating nozzle to collection nozzle

T: Length of throat. In equations, T is temperature

θ: Inclination angle of cone inlet

Q_m: Minor flow rate at STP

Q_T: Total flow rate at STP

τ_p: Particle relaxation time

ρ_p: Particle density

d_p: Particle diameter

$d_{p\,50}$: Particle diameter at 50% collection efficiency

C_c: Cunningham slip correction factor

μ: Molecular viscosity of fluid

ρ: Density of fluid

St: Stokes number

$St^{1/2}{}_{50}$: Stokes number at 50% collection efficiency

τ_e: Fluid eddy turnover time

Φ_p: Volume fraction of particles

Ƙ: Turbulent kinetic energy

ϵ: Energy dissipation rate

L: Length of device in the spanwise direction

P: Instantaneous pressure

$\bar{P}$: Reynolds-averaged pressure

p: Fluctuating pressure

τ: Viscous stress tensor

S: Rate of strain tensor

δ: Unit tensor

E: Sensible enthalpy

k_{eff}: Effective thermal conductivity

τ_{eff}: Effective stress tensor

Y_M: Turbulent Mach number

v_{sound}: Speed of sound at STP

μ_t: Turbulent viscosity

ϕ: Transported scalar quantity

Re_p: Relative or particle Reynolds number

Ma_r: Relative Mach number

$|\vec{U}-\vec{U}_p|$: Slip Velocity

C_D: Drag Coefficient

g: Gravitational acceleration

λ: Mean free path 3

Kn: Knudsen number

α: Normally distributed random number

ϕ(p,q): Euclidean distance squared between points p and q

W_k: Inverse distance weight function

ω_k: Inverse distance weight function

R_w, R_q: Radii of influence in scattered interpolation

N_w, N_q, N_b, N: Number of nodes in scattered interpolation

Q_k: Quadratic polynomial

R: Residual of interpolation

Chapter 1

Introduction

1.1 Problem Statement

Our goal is to employ Computational Fluid Dynamics (CFD) simulations to design efficient samplers for collecting and concentrating biological aerosols. The concept of using computers to design and study such systems is appealing because numerical solutions serve as a reliable substitute for often fastidious and expensive experiments, not to mention the fact that they offer the ability to subject those systems to a wide range of conditions normally unattainable in the laboratory. The problem lends itself to the broad area of multiphase flow which remains a challenging problem for scientists and engineers despite the notable achievements in understanding and characterizing single phase flows [1]. Particulate flows are of interest due to their ample abundance in nature, their appearance in several industrial applications and human health activities. This thesis focuses on the subcategory of gas-solid flows as encountered in an aerosol “virtual impactor” sampling device (Figure 1.1). A virtual impactor is used to separate particles from the atmosphere based on their inertia or aerodynamic size. The device also serves as a concentrator for particles with initially low concentrations. The collection of particles in high concentrations, and with relatively the same physical properties aids real time measurement of aerosol composition in areas such as toxicological studies, and biological threat assessment situations. The need for mobility and versatility in such situations prompted the integration of virtual impactors into “Personal Aerosol Samplers” [2], and consequently renewed interest in this classical engineering problem.

The underlying framework for designing efficient aerosol samplers encompasses a multitude of problems concurrently related to physics and engineering. The prediction of the carrier fluid flow, especially in the presence of turbulence, introduces difficulties that are still being studied and analyzed within the fluid mechanics community. Furthermore, the solution of the particle equation of motion and its correct representation of the dominant physical phenomena remains an active area of research for many scientists. Lastly, the computational aspects of modeling such an intricate dynamic system in an accurate and efficient way also manifest some challenges.

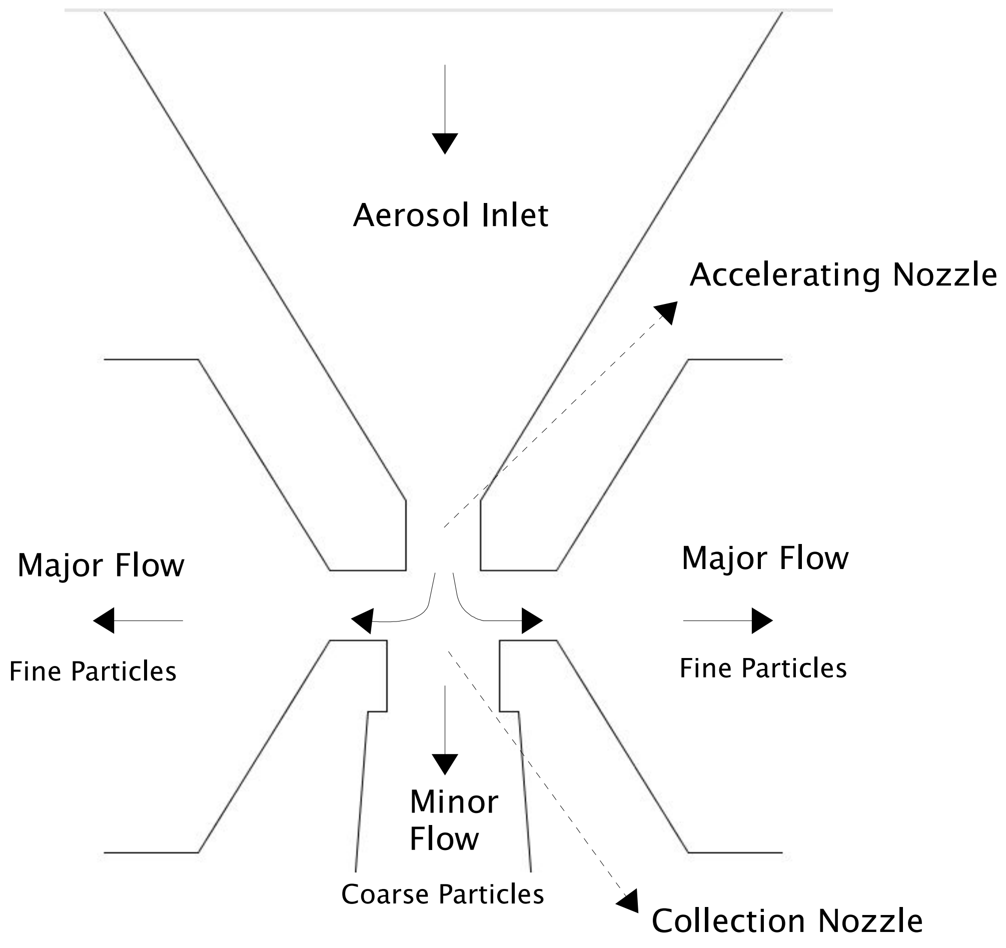

Figure 1.1: Overview of a virtual impactor

1.2 Overview of Virtual Impactors

The schematic shown in Figure 1.1 is a two dimensional cross-section of the fully three dimensional virtual impactor under study. The top inlet cone creates a passage for the particle-laden air to a rectangularly shaped throat section which in turn converges the flow into the accelerating nozzle. The receiving end of the impactor constitutes the collection nozzle which is slightly larger in width than the accelerating nozzle. Larger particles cross the fluid streamlines into the collection probe, while smaller particles move with the majority of fluid as it is forced to exit into the side passages (vacuumed). Typically, only a small fraction (10% – 20%) of the inlet flow is allowed to enter into the collection region normally referred to as the minor flow, whereas the side effluent is referred to as the major flow. In reality, the particles exiting the minor flow are collected on a filter, or in a liquid solution for experimental evaluation [3], or are passed through a cyclone for further separation. Virtual Impactors became a sound replacement for solid surface inertial impactors due to several advantages [4]; they eliminate particle bounce and re-entrainment, reduce interstage wall losses, and prevent large particle breakup due to impaction. Additionally, they allow the collected aerosol to remain suspended rather than deposited with more control over its final concentration [5]. The performance of a virtual impactor is characterized by a collection efficiency and wall losses curve. The efficiency is defined as the fraction of particles of a given size that end up in the minor flow. For an ideal impactor, this is a sharp step function, however due to the inevitable contamination of the collected aerosol with the relatively smaller particles, the curve takes on an "***S***-shape" (Figure 1.2) [6]. The efficiency curve reveals the 50% cutpoint diameter indicating that half the particles with the cutpoint diameter is collected while the other half is forsaken to the major flow. Wall losses are generally undesirable in virtual impactors, and parametric studies reported in the literature have been done to minimize them. The majority of the losses are observed on the inner surfaces of the collection nozzle.

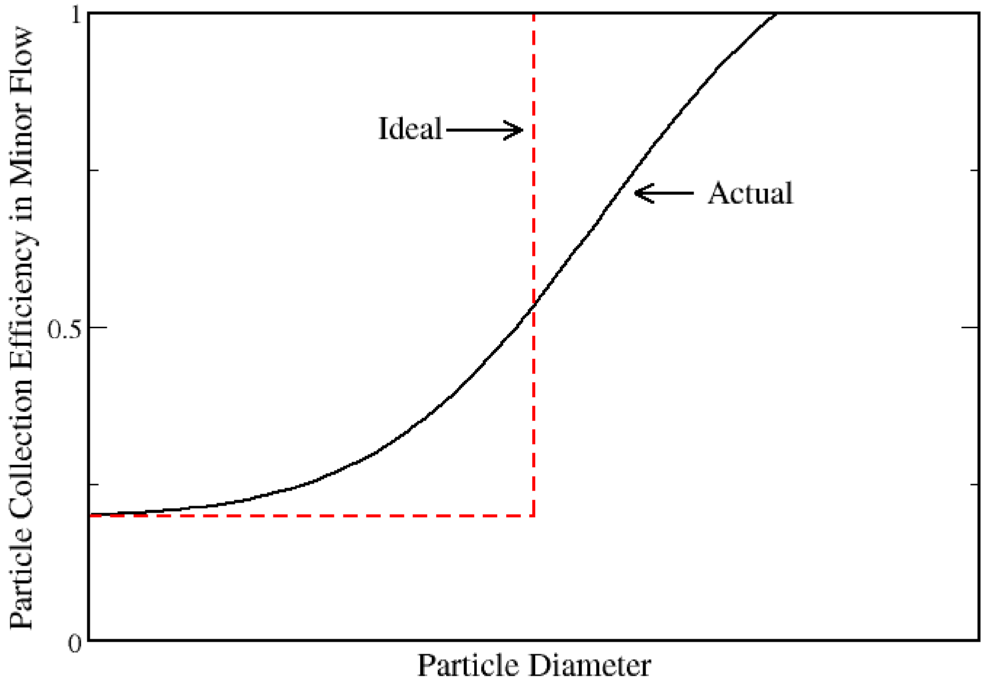

Figure 1.2: Actual and ideal collection efficiency for a virtual impactor

1.3 Contributions

This thesis makes the following key contributions:

- The simulation of a contentious virtual impactor at high Reynolds number conditions.
- Investigation of the differences in the solutions of the widely used

isotropic Ƙ-ϵ turbulence model, and the more robust non-isotropic Reynolds stress model.

- The study of the effect of turbulent particle dispersion on the impactor's efficiency and wall losses curve.
- Development of AeroTrack; an efficient and versatile particle tracking computer program.
- Application of a mathematically commendable multi-variate interpolation scheme, and the demonstration of its superior accuracy over commonplace methods.
- Analysis of the impact of the drag coefficient formulation on the particle motion.
- Demonstration of the potential of CFD/particle tracking simulations to design aerosol samplers, and optimize their performance.

1.4 Organization of the Thesis

Chapter 2 provides background on virtual impactor studies, both experimental and numerical. It helps set the stage for the important features of a well designed virtual impactor, and also shows the lack of initiative when it comes to modeling such a system under turbulence conditions. Chapter 3 describes the simulation components put together for the conduction of this research. Chapter 4 presents the fundamental theoretical equations and models used to arrive at a fluid flow solution.

Moreover, the numerical schemes employed are discussed, and a detailed portrayal of the algorithms developed for the particle tracking code is adduced. Chapter 5 deals with the numerical analysis of the computational methods. Chapter 6 discusses the simulation results obtained, and depicts the evaluation of the device performance. Finally, Chapter 7 concludes with a summary of the work presented and outlines future work.

Chapter 2

Background and Related Work

2.1 Experimental Investigation of Virtual Impactors

The vitality of virtual impactors as effective means for sampling aerosols had been recognized for quite some time. For this reason, many experimental studies have been dedicated to evaluating and optimizing their performance. A brief review of the main findings and the optimal set of parameters is presented here. Chen and Yeh [14] appraise a well-designed virtual impactor as one with a sharp separation curve, little wall losses, and minimal fine particle contamination. To achieve such objectives they conducted experiments with varying geometrical dimensions and flow parameters such as the nozzle Reynolds number, Stokes number, and the minor-to-total flow ratio, Q_m/Q_T. The Reynolds number for a rectangular or slit nozzle is defined as:

$$Re=\frac{UD_h}{\nu}=\frac{U(2W)}{\nu} \tag{2.1.1}$$

where U is the average fluid velocity at the nozzle, D_h is the hydraulic diameter, ν is the kinematic viscosity of the fluid, and W is the slit width. The collection efficiency of impactors is conventionally plotted as a function of the Stokes number, which is defined as the ratio of the particle relaxation time, $\tau_p = \frac{\rho_p d_p^2 C_c}{18\mu}$ to the fluid time scale at the nozzle:

$$St = \frac{\tau_p}{\left(\frac{1/2\,W}{U}\right)} = \frac{\rho_p d_p^2 C_c U}{9\mu W} \qquad (2.1.2)$$

where ρ_p is the particle density, d_p is the particle diameter, C_c is the Cunningham slip correction factor, and μ is the fluid viscosity. It can be seen that the square root of the Stokes number acts as a dimensionless particle diameter, thus to achieve low 50% cutpoints, high velocities and extremely narrow nozzles are compulsory [3]. Chen et al. [15] conclude that for a fixed set of geometrical and dynamical configurations, variations in the Reynolds number between 1000 and 8000 do not gravely influence the 50% cut-off Stokes number ($\sqrt{St_{50}}$ ≈ 0.67, for Q_m/Q_T ~ 11%). The minor-to-total flow ratio, on the other hand, has severe consequences on the cutpoint as well as on the wall losses. As Q_m/Q_T is increased $\sqrt{St_{50}}$ is decreased. This can be explained by the fact that higher ratios allow more small particles to pass through to the collection nozzle. Chen et al., arrived at a least-squares polynomial fit over a range of experimental data using two prototype impactors (Figure 2.1). The effect of increasing Q_m/Q_T on the wall losses is also favorable in a sense that they are decreased. A typical wall losses curve exhibits a peak around the 50% cutpoint, and this peak is usually reduced for higher Reynolds numbers [16].

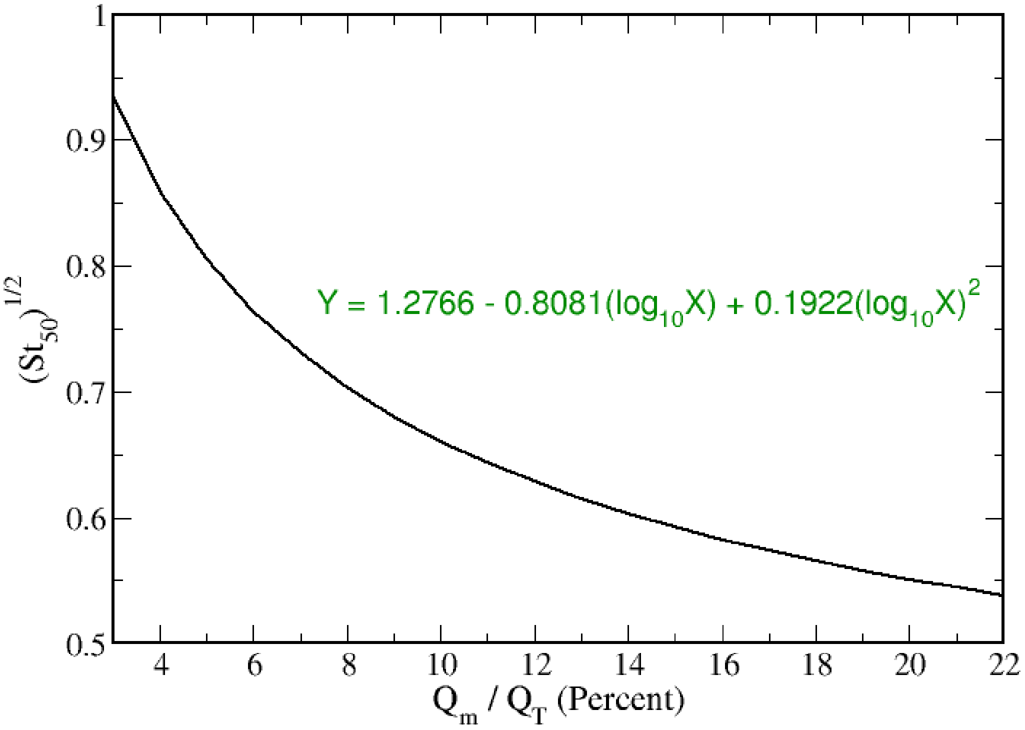

Figure 2.1: Semi-empirical dependence of $St_{50}^{1/2}$ on Q_m/Q_T from [15]

Geometric considerations whose impact on the performance of the virtual impactor was studied are the throat length (T), the inlet angle of inclination (θ), the dimensionless nozzle to probe distance (S/W) and the dimensionless probe diameter (W_1/W). The value of inlet inclination angle remained fixed at 30° in all the experiments conducted by Chen et al., who report slight variations in the 50% cutpoint when their data is compared to another study in the literature with a value of 45°. This can

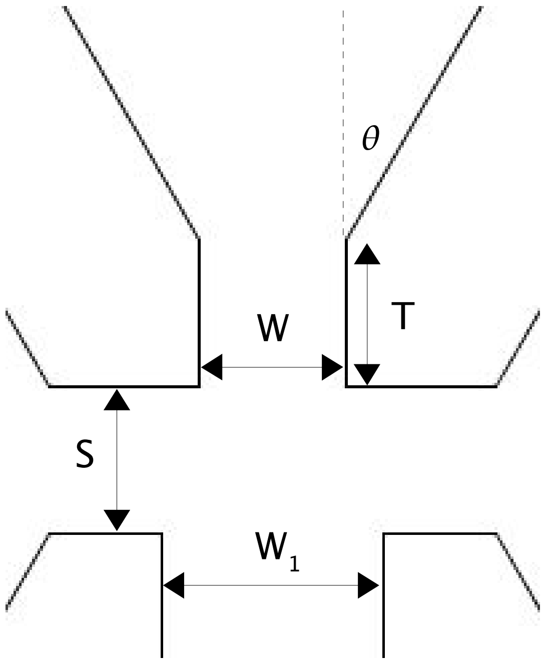

Figure 2.2: Influential nozzle design parameters

be attributed to a more focused jet where the particles are driven closer towards the center of the separation zone. The throat length effect was investigated with a virtual impactor with no throat which resulted in a broader slope for the efficiency curve. Since steep slopes are normally desirable, impactors with finite throat lengths were also tested, however increases in the ratio T/W beyond unity did not result in any improvements. Much of the pioneering work on virtual impactors was done with a ratio S/W close to unity. In the same study [15], experiments with three different ratios (S/W = 0.53, 1.03, 1.59) revealed similar characteristics for the efficiency curve with a slim advantage to the smallest ratio. Similarly, the ratio W_1/W was varied between 1 and 1.5, and the measured separation efficiency is identical for small values of the Stokes number, but develops a slightly steeper slope for the unity ratio at high Stokes numbers.

The challenge of designing a virtual impactor suitable for sampling particles generated by industrial and combustion processes, which are normally in the fine spectrum of particle diameters (0.2-0.8 μm), and at high sampling flow rates was addressed by Sioutas in his PhD dissertation [3]. The virtual impactor in that study consisted of a rectangular jet, with a nozzle width W = 0.33 mm, and a total sampling flow rate of 225 LPM. The reported $\sqrt{St_{50}}$ at 20% flow separation is 0.45, which reflects a 0.12 μm cutpoint. To the knowledge of the author, no other work has been done to validate or reproduce the Sioutas findings under those particular conditions. For this reason, we aim at studying the characteristics of such an impactor in this work.

2.2 Theoretical and Numerical Studies of Virtual Impactors

Using numerical analysis techniques, Marple [6] obtained flow field information inside a virtual impactor by solving the Navier-Stokes equations for an axis-symmetric and a two-dimensional jet. He then integrated the particle equation of motion governed by Stokes drag with the Cunningham slip correction factor. The study was instrumental in providing theoretical insight into the effect of the flow and physical design parameters. However, judged by today's computational standards it can be viewed as preliminary especially when it comes to considering high Reynolds number flows. The approach was later revised in a subsequent paper by Rader and Marple [17] where they applied the technique to study solid surface impactors. In that paper, the refinement focused on two main areas, (i) the discretization gird, and (ii) the drag coefficient. The flow regime, however, was far from turbulent. The premise behind grid refinement is that the numerical solution inherently carries some errors which can be minimized by reducing the grid spacing, and, in fact, they observed significantly different results on the finer grids which yielded efficiency curves with steeper slopes. In addition, the drag coefficient formulation was re-written to account for instances where the particle Reynolds number exceeds unity; in other words, violates the Stokes drag regime (termed “ultra-Stokesian”). The drag coefficient is of primary concern to this research as well, and will be discussed in detail in a later chapter. In a more recent study [18], Asgharian and Godo employed a commercial finite-element fluid dynamics code to obtain the flow velocity in a two-dimensional “improved” virtual impactor. An improved virtual impactor is one with a clean air core in the center of the inlet to the impaction zone. Masuda et al. [19] reported low fine particle contamination in the minor flow by the introduction of a clean air core, and from their experimental data proposed a semi-empirical equation for the prediction of the collection efficiency. Chen and Yeh [14], conducted experiments on such an impactor and arrived at the same conclusions as the original study. In the numerical study, Asgharian and Godo re-construct the geometry proposed by Chen and Yeh, and solve for the incompressible steady-state flow field at a Reynolds number of 4000 using the standard k-epsilon turbulence model. The computational mesh consisted of approximately 14,000 nodes which is relatively modest compared to what is used in this research. The trajectories of solid particles were obtained by solving the particle equation of motion governed by drag using a fourth-order Runge-Kutta integration scheme. The authors resorted to linear interpolation to obtain the fluid velocity at a position that does not coincide with a mesh point. As we shall discuss later in this thesis, such a methodology introduces additional numerical errors that can deteriorate the overall accuracy. Furthermore, the effect of particle dispersion due to turbulence was not addressed in their paper.

2.3 Approaches to Two Phase Turbulent Flows

From a more fundamental perspective, the numerical prediction of turbulent fluid particle flows has been the subject of several distinguished articles [7,8,9] and reviews [10,11,12]. In this section we aim at presenting the two mainstream approaches of Lagrangian and Eulerian schools of thought, with emphasis on the former since it is the ideology adopted in this research. Crowe [11] classifies fluid particle flows as being either dilute or dense. In a dilute flow, the surface and body forces dictate the motion of the particle, whereas in a dense flow, particle-particle collisions play a prominent role. The philosophy behind this classification stems from the concept of coupling, which generally accounts for the dominant physical interactions in the flow. Two key scaling parameters prescribe the coupling phenomenon. The Stokes number, which is the ratio of particle response time τ_p to a characteristic time in the fluid itself τ_e, such as large eddy turnover time. If the particle response time is much smaller than the fluid time scale then the particle will most likely align itself with the fluid motion. On the other hand, if τ_p is much greater than τ_e then the particle is less likely to respond to fluid motions whose time scale is reflected by τ_e. The other scaling parameter is the volume fraction of the particles Φ_p, defined as the ratio of the total particle volume to the volume occupied by the fluid. According to Elghobashi [10], insight into the coupling problem is gained by mapping a certain set of parameters to one of the regions in Figure 1.3. In the one-way coupling regime, the particles have negligible effect on the carrier phase turbulence. In the two-way coupling regime, particle concentrations are high enough to induce momentum transfer to the fluid thus altering its turbulence properties. As the volume fraction increases beyond the limits of dilute suspensions and into the realm of dense suspensions, the coupling is termed four-way implicating the effect of particle-particle collisions.

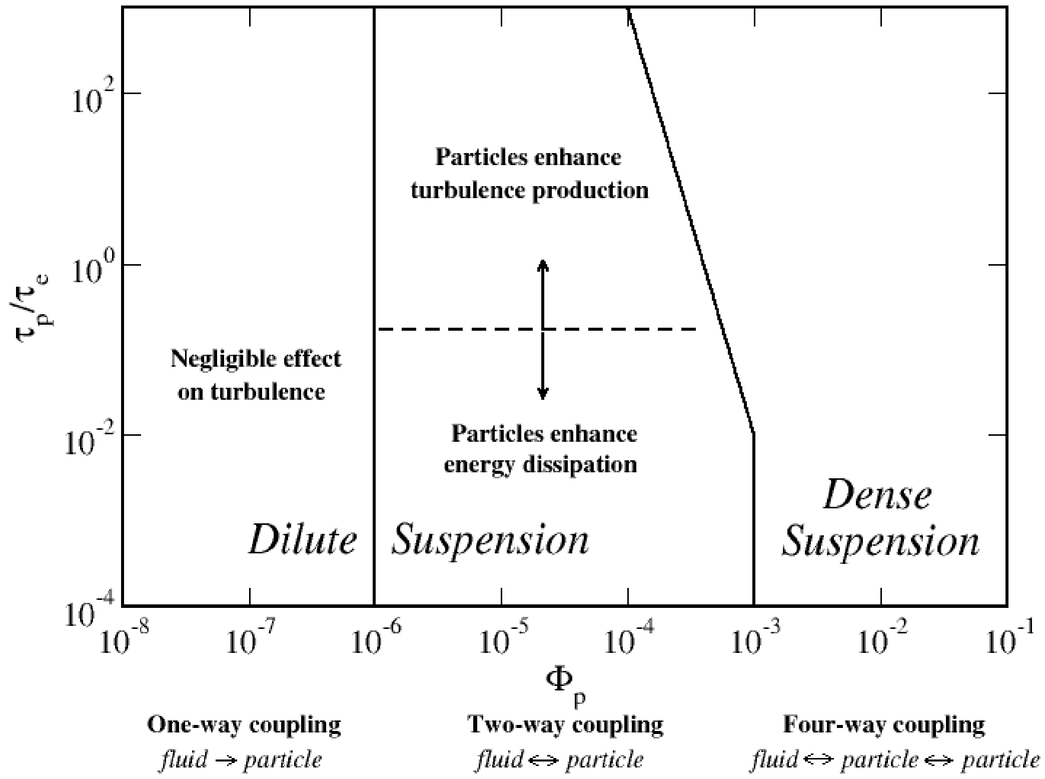

Figure 2.3: Map of regimes of interaction between particles and turbulence from [10]

The Lagrangian approach computes the trajectories of computational particles released into the flow field. Updates to the particles' position and velocity are acquired by integrating the particle equation of motion with respect to time. Historically the equation has been named the

Basset-Boussinesq-Oseen (BBO) equation after the people who worked on it. BBO arrived at the equation by solving the Navier-Stokes equations without the advective acceleration terms for the unsteady rectilinear motion of a sphere in a stagnant incompressible creeping fluid [13]. Several subsequent authors have studied and analyzed the individual terms in the equation and proposed modifications for situations not considered in the original BBO equation. The form and limitation of the equation as related to this research will be discussed in a later chapter.

The alternative Eulerian or two-fluid approach regards the dispersed phase as a continuum whose motion is prescribed by transport equations similar in mathematical form to those of the carrier fluid. Those transport equations are associated with a cloud of particles present in a unit volume as opposed to the single particle Lagrangian equation. The Eulerian approach proved to be useful in simple dense flows since it has built into it quantities that describe particle-particle collisions and the effect of neighboring particles. Unfortunately, it suffers from the same drawbacks encountered in solving single phase flows, namely the resort to constitutive and empirical models. For aerosol flows with one-way coupling, the Lagrangian approach is more suitable since it provides information specific to each individual particle and allows for the simulation of poly-dispersed suspensions, as is the case in this work.

Much of the theoretical work on two-phase flows relies on the Direct

Numerical Simulation (DNS) of the three-dimensional, time-dependent Navier-Stokes equations to predict the fluid phase motion. This technique caters detailed instantaneous information about all the scales of turbulence, and is a valuable asset in setting the directions for numerical models that aim at predicting particulate flows. However, due to its harsh computational requirements in terms of grid spacing and time stepping, it is impractical for most engineering applications (computational effort proportional to Re^3). The workaround to this problem is to use turbulence closure models which are solved in conjunction with the Reynolds Averaged Navier Stokes (RANS) equations to provide the mean quantities of the flow field. The most widely cited model is the standard Turbulent Kinetic Energy (Ҡ-ϵ) model, which incorporates into its methodology transport differential equations for the turbulent kinetic energy and dissipation rates. Particle dispersion can then be calculated via stochastic methods that rely on the random eddy representation of the underlying fluid.

Chapter 3

Simulation Components

3.1 Compressibility

Originally the flow was assumed incompressible, however the average velocity of the fluid at the nozzle is close to the sonic speed (0.61 Mach), moreover, the discrete phase results, as will be shown in Chapter 6, revealed wall losses that were beyond what was expected, so it became obvious that compressibility effects may introduce changes in the flow which in turn affect the particulate flow, and hence the efficiency curve. Therefore, simulations were conducted by additionally solving the energy equation along with the ideal gas law to determine the temperature and density variations in the flow.

3.2 Device Geometry

As mentioned in Chapter 2 the optimal set of dimensions for the virtual impactor has been established through both experiments and theory. In the simulation presented here, we utilize the recommended aspect ratios for the nozzle geometry as they were reported in the high sampling rate experiment done by Sioutas [3,51]. The length scales in the remaining parts of the device are partially based on the design of Asgharian and Godo [18], and are depicted in Figure 3.1.

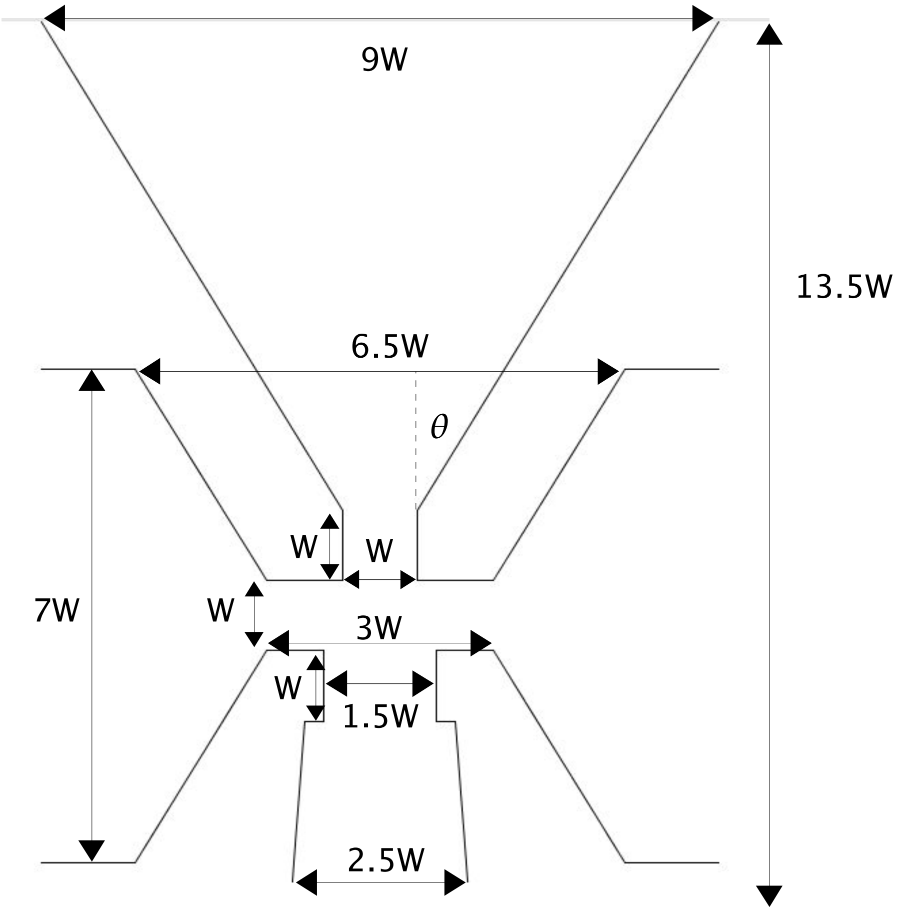

Figure 3.1: Dimensions of simulated virtual impactor (XY plane)

The width of the acceleration nozzle is identical to that of Sioutas (W=0.33mm). The inlet angle of inclination was not reported in his experiment so we used the canonical value of 30°. The depth of the device into the plane of the paper is also taken from Sioutas to be L = 5.6 cm. The simulations were performed in 3D despite the ratio L/W which is much greater than one, because it is physically unrealistic to study the effect of turbulence in two dimensions only.

3.3 Computational Meshes

The physical domain is constructed using GAMBIT, which is a subset of the FLUENT™ CFD package [20]. GAMBIT is a pre-processor for the solver capable of geometric modeling and mesh generation. The grid is generated by first meshing a two-dimensional plane of the device, and then constructing hexahedral cells in the third dimension via a sweep operation (Figure 3.2), hence the entire domain is divided into six-faced elements or control volumes. The grid is considered structured since it is formed by a repeated geometric topology, namely the hexahedron. Structured girds are known to provide accurate calculations of spatial derivatives, and are most often flow aligned thus yielding greater accuracy. Two meshes with varying degrees of resolution were constructed. The coarse mesh containing 926,250 cells (965,736 nodes) was used for the incompressible and compressible flow simulations, whereas the finer mesh with 2,021,600 cells (2,116,797 nodes) was only used for the compressible flow simulation. The mesh statistics are summarized in Table 3.1. Two-dimensional cross-sections of the meshes in the (XY) plane are shown in Figures 3.3 and 3.4. In the coarse mesh the width of the nozzle is discretized by 50 equally spaced nodes, whereas in the finer mesh 100 nodes are used. The resolution of the grid in the nozzle section remained constant away from the wall in order to maintain the accuracy all the way to the bulk region.

	Min Face Area (m^2)	Max Face Area (m^2)	Min Cell Volume (m^3)	Max Cell Volume (m^3)	No. Cells in XY Plane
Coarse	3.80E-011	1.20E-007	4.60E-014	6.00E-012	18525
Finer	1.10E-011	1.10E-007	2.20E-014	2.70E-012	72200

Table 3.1: Computational mesh statistics

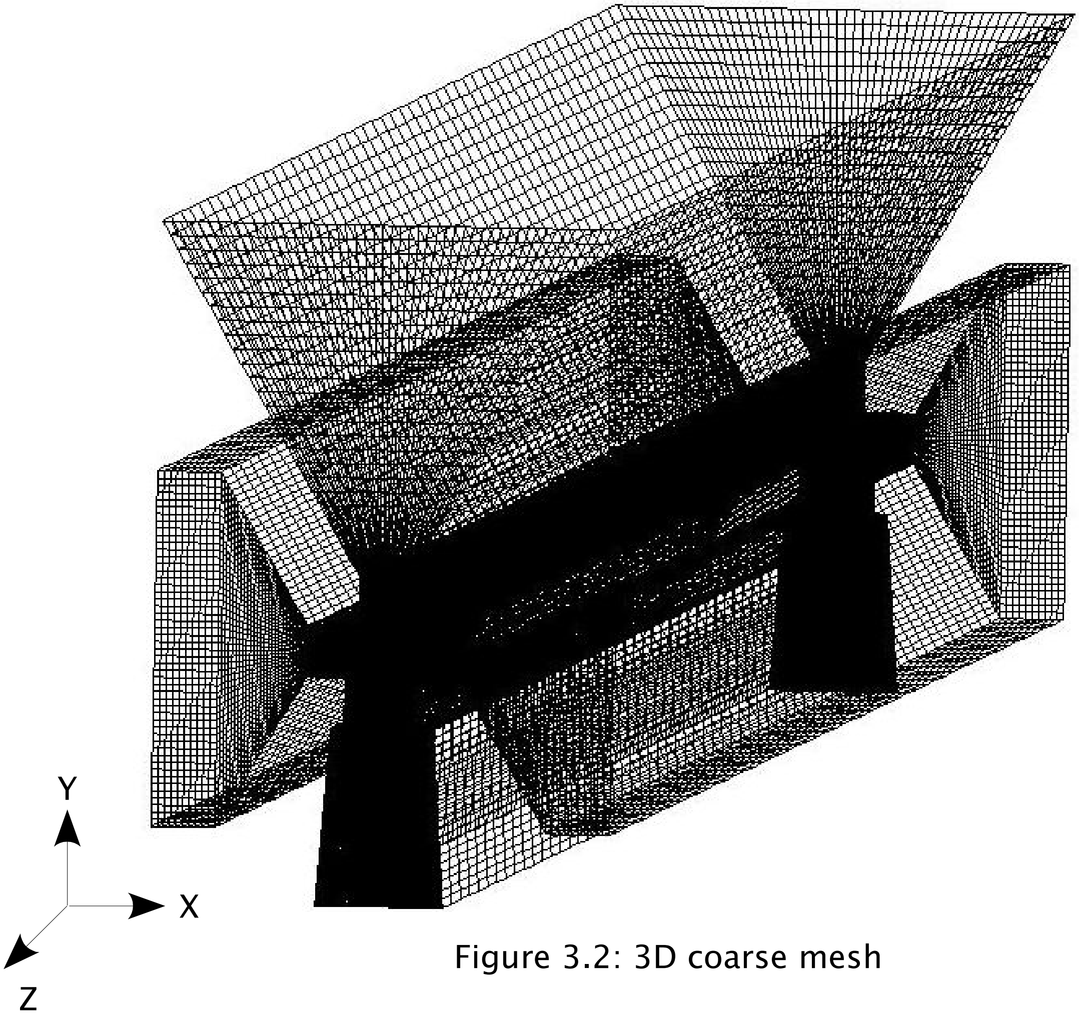

Figure 3.2: 3D coarse mesh

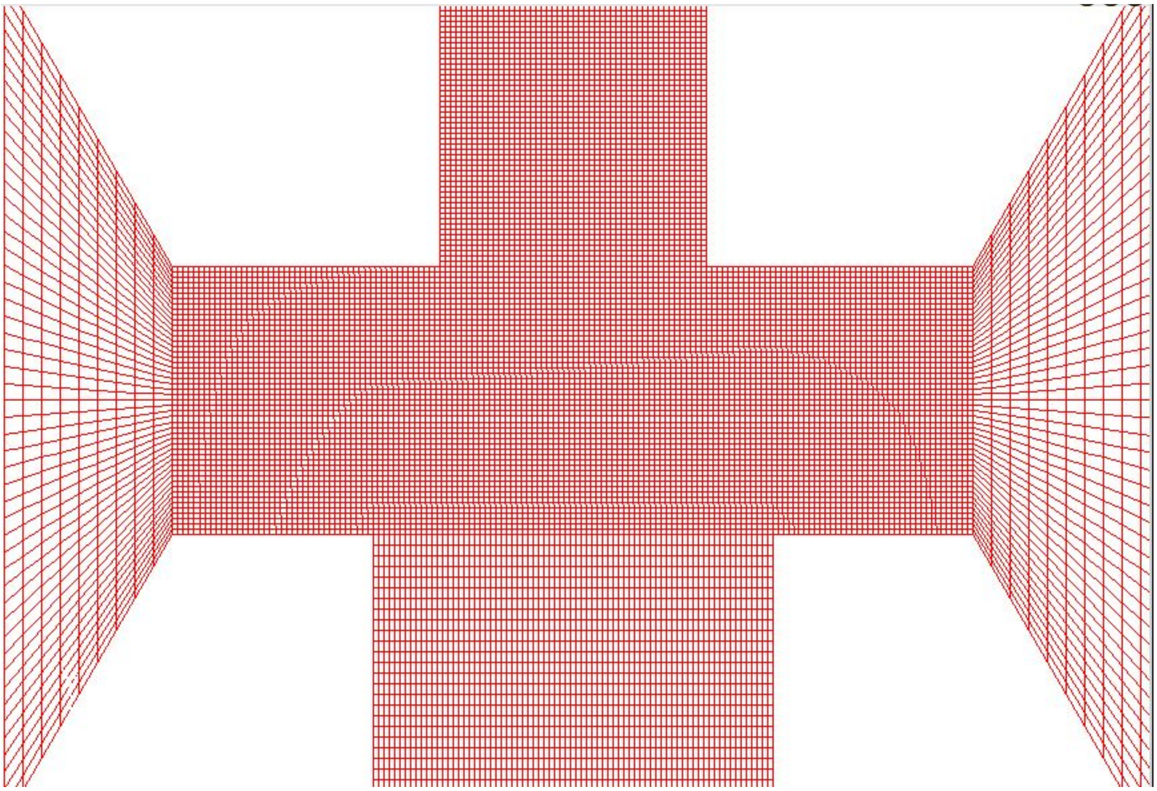

Figure 3.3: Nozzle cross-section (coarse mesh: Δ=W/50)

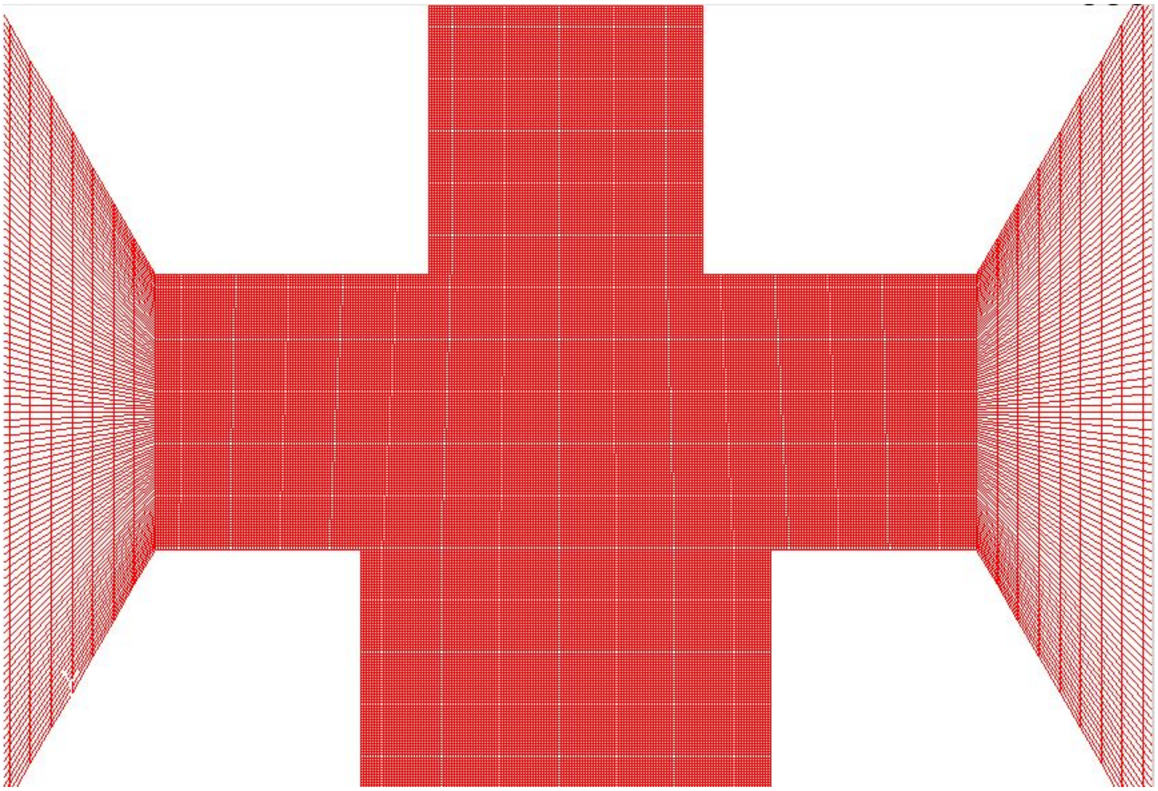

Figure 3.4: Nozzle cross-section (finer mesh: Δ=W/100)

3.4 Boundary and Operating Conditions

An inlet boundary condition is specified at the top entrance of the device. To mimic the experimental setup of Sioutas where the flow rate was set to Q_T = 225 liters per minute, we specify a uniform mass flow rate, m' = ρQ_T = 4.59375e-03 Kg/s at STP, which is applicable for both incompressible and compressible simulations (air density, ρ = 1.225 Kg/m^3). The flow Reynolds number under these conditions is in the turbulent regime (Re $\simeq$ 9169, for μ = 1.7894E-05 Kg/m.s). The turbulence intensity is thus estimated from an empirical correlation derived for fully turbulent pipe flows, and is given by: $\boldsymbol{I} = u/u_{avg} = 0.16Re^{-1/8} \simeq 5\%$, where u is the fluctuating velocity component.

The conditions at the inlet are considered ambient, namely the temperature is set to T = 300°K, and the pressure to atmospheric (gauge pressure: $P_G = 0$). At the exit locations of the device, a "pressure-outlet" boundary condition is used. The minor flow rate Q_m is calculated using the mass flow rate reported by the simulation, and the density of air at STP. To achieve the desired 20% Q_m/Q_T ratio, the minor flow gauge pressure was fixed at slightly below atmospheric (P_G = -100 Pa), and the major flow gauge pressure was varied over a number of trials to obtain the desired ratio since the dependence of the flow separation on the vacuum pressure is not known a priori. To guide the trials, advantage was taken of the reported experiment's value for the pressure drop across the nozzle itself, ΔP = 30 KPa. For the incompressible simulation it was sufficient to use a value of P_G = -27.2 KPa to arrive at a minor-to-total flow ratio close to 20%. The compressible simulation, however, demanded a value of P_G = -50 KPa at the major flow outlet.

The boundary conditions at the walls of the device are taken to be no-slip for the velocity, and adiabatic for the temperature (no heat flux). Details of the calculation procedure and the use of "wall functions" will be discussed in the next chapter.

3.5 Discrete Phase Properties

Solid spherical particles were used to simulate the flow of aerosols through the virtual impactor. Their density was taken to be that of Polystyrene Latex microspheres in order to mimic the experiment (ρ_p= 1.047 g/cm^3). Eleven mono-disperse samples each with a different particle diameter (0.1-0.4 μm) are separately released from random locations in the entrance-cone of the device. The samples were released in vertical planes away from the front and back walls so that fluid disturbances in the spanwise direction are negligible. Details on the number of particles used, and the confidence limits for the Lagrangian statistics will be presented in Chapter 5. Finally, the walls of the device are treated as perfect collectors or traps. Thus, when a particle is at a distance less than or equal to its radius from a wall it is recorded as deposited and its calculation terminates.

Chapter 4

Governing Equations

4.1 Fluid Flow

According to Panton [21], there are three predominant independent dynamical laws in continuum mechanics: the continuity equation, the momentum equation, and the energy equation. The continuity equation prescribes the law of conservation of mass and is given by:

$$\partial\rho/\partial t + \nabla\bullet(\rho\mathbf{U}) = 0 \tag{4.1.1}$$

where ρ is the density, ∇ is the divergence operator, and $\mathbf{U}$ is the fluid velocity vector. The physical interpretation of the continuity equation is understood in reference to a fixed point in space as the balance between the rate of accumulation of mass per unit volume and the net outflow of mass per unit volume. For an incompressible flow, the continuity equation reduces to $\nabla\bullet\mathbf{U} = 0$.

The momentum equation for a continuum fluid is derived in a manner similar to Newton's second law of motion, and is essentially a balance between the rate of change of momentum per unit volume on one hand, and the pressure, viscous, and body forces in each direction, on the other. The differential form of the equation is given by:

$$\partial(\rho\mathbf{U})/\partial t + \nabla\bullet(\rho\mathbf{UU}) = -\nabla P + \nabla\bullet\tau + \rho\mathbf{F} \tag{4.1.2}$$

where P is the pressure, **F** is the body forces vector, and τ is the viscous stress tensor, given by Newton's law of viscosity as:

$$\tau = -2/3\mu\delta\nabla\bullet\mathbf{U} + 2\mu\mathbf{S} \tag{4.1.3}$$

where **S** is the rate-of-strain tensor, and δ is the unit tensor.

Lastly, the energy equation represents a balance between the rate of accumulation of energy (internal and kinetic) per unit volume, and its convection due to flow on one hand, and the net heat flow, and work due to surface and body forces on the other. The differential form of the equation as it is solved by FLUENT™ is as such:

$$\partial(\rho E)/\partial t + \nabla\bullet(\ (\rho E+P)\mathbf{U}\ ) = \nabla\bullet(\ k_{eff}\nabla T + \tau_{eff}\bullet\mathbf{U}\ ) \tag{4.1.4}$$

where

$E = h - P/\rho + U^2/2$; h is the sensible enthalpy, k_{eff} is the effective thermal conductivity (accounts for turbulent viscosity), and τ_{eff} is the effective stress tensor (includes the thermodynamic pressure).

For incompressible flows with a Newtonian fluid, the continuity equation along with the three momentum equations in the x, y and z direction form a closed system of four equations with four unknowns (P,U_x,U_y,U_z), and are commonly known as the Navier-Stokes equations. Ideally, one would like to solve a flow problem directly by integrating the Navier-

Stokes equations, however, as mentioned in Chapter 2, with the current computer capabilities this approach is limited to low Reynolds number flows in simple geometries. The need to predict turbulent flows in engineering problems brought about the concept of time averaging, first pursued by Reynolds in the 19th century. The theory and models of this technique as used in this research are presented in the next section.

4.1.1 Reynolds Averaged Navier Stokes Equations

RANS equations are obtained by decomposing the instantaneous flow variables such as the velocity and pressure into a mean component and a fluctuating component, as such:

$$U=\overline{U}+u \tag{4.1.5}$$

$$P=\overline{P}+p \tag{4.1.6}$$

Substituting those expressions into the general Navier-Stokes equations, neglecting the gravitational body force term, and taking their time or ensemble average gives in index notation:

$$\frac{\partial \rho}{\partial t}+\frac{\partial(\rho \overline{U_i})}{\partial x_i}=0 \tag{4.1.7}$$

$$\frac{\partial(\rho U_i)}{\partial t}+\frac{\partial(\rho U_i U_j)}{\partial x_j}=\frac{-\partial \overline{P}}{\partial x_i}+\frac{\partial}{\partial x_j}\left[\mu(\frac{\partial U_i}{\partial x_j}+\frac{\partial U_j}{\partial x_i}-\frac{2}{3}\delta_{ij}\frac{\partial U_l}{\partial x_l})\right]-\frac{\partial(\rho \overline{u_i u_j})}{\partial x_j} \tag{4.1.8}$$

where $\delta_{ij} \equiv$ Kronecker delta . The last term of the equation $\left(\rho\overline{u_i u_j}\right)$ is referred to as the Reynolds stress tensor and it introduces an additional unknown that needs to be modeled in order for the system to be fully closed. Physically, one can interpret the Reynolds stress as a supplemental stress acting on the mean field due to the turbulent velocity fluctuations analogous to the stress exerted on the fluid due to the velocity fluctuations at the molecular level [22]. This perspective establishes the foundation for most of the work that aims at postulating turbulence models that can provide closure to the RANS equations. The two models investigated in this thesis are the standard k-epsilon (Ƙ-ϵ), and the Reynolds Stress Model (RSM).

4.1.2 The Ƙ-ϵ Closure

The k-epsilon model falls within the class of two-equation turbulence models since it introduces two additional transport equations, one for the turbulent kinetic energy Ƙ, and one for the dissipation rate ϵ. The Ƙ-ϵ model relies on a constitutive law for the Reynolds stress as means of providing closure. This law known as the Boussinesq hypothesis, although not rigorously justified, builds on the analogy between the molecular momentum transport and turbulent momentum transport, and is formulated as such:

$$\rho\overline{u_i u_j}=\frac{2}{3}(\rho K+\mu_t\frac{\partial U_i}{\partial x_i})\delta_{ij}-\mu_t(\frac{\partial U_i}{\partial x_j}+\frac{\partial U_j}{\partial x_i}) \tag{4.1.9}$$

where $\mu_t > 0$ is now a turbulent or eddy viscosity. An almost automatic disadvantage of this hypothesis is its incapability of dealing with anisotropy in the flow, in other words, the fact that μ_t is a scalar quantity, forces the same constant of proportionality onto the mean rate-of-strain tensor components. A particular case where the model fails is in the basic, yet important, channel flow. The above form of the Boussinesq hypothesis erroneously predicts equal quantities for the normal components of the stress tensor, i.e. $(\overline{u_1^2}=\overline{u_2^2}=\overline{u_3^2}=\frac{2}{3}K)$. As we shall see later in Chapter 6, this will have tremendous effects on the accuracy of the stochastic models that are relying on the velocity variances to predict turbulent particle dispersion.

The turbulent kinetic energy in the constitutive model is obtained by solving the differential equation that governs its transport. Such an equation can be easily derived from the Navier-Stokes equation to yield its exact form, which can then be manipulated by inserting the constitutive law for the Reynolds stress, and modeling the pressure work and kinetic energy fluxes via a gradient-like transport law [22]. The resultant equation as solved by FLUENT™ is given below:

$$\frac{\partial(\rho \text{Ƙ})}{\partial t}+\frac{\partial(\rho \text{Ƙ}\overline{U_i})}{\partial x_i}=\frac{\partial}{\partial x_j}\left[(\mu+\frac{\mu_t}{\sigma_k})\frac{\partial \text{Ƙ}}{\partial x_j}\right]+\mu_t\frac{\partial\overline{U_i}}{\partial x_j}(\frac{\partial\overline{U_i}}{\partial x_j}+\frac{\partial\overline{U_j}}{\partial x_i})-\rho\epsilon-Y_M \tag{4.1.10}$$

where $\sigma_k = 1$ is a turbulent Prandtl number. Naturally, the first term on the right hand side is the transport of Ƙ, the second term represents the generation of turbulent kinetic energy caused by mean velocity gradients, while the third term is the dissipation rate which has its own transport equation, presented below, and the last term is only applicable for compressible flows and it represents an additional turbulence dissipation-like phenomena brought about by compressibility. Y_M is modeled through what is called a turbulent Mach number as such:

$$Y_M=2\rho\epsilon\frac{\text{Ƙ}}{v_{sound}^2} \tag{4.1.11}$$

Despite the existence of an exact transport equation for ϵ, solutions to the kinetic energy equation are most often impeded due to the difficulty in computing ϵ accurately, primarily because of the tremendous modeling that goes into the ϵ equation (seven out of eight terms require modeling). Clearly, some of the physics is captured in these models, in part because of our understanding of isotropic and homogeneous turbulent shear flows, however, an implicit leap of faith is presumed when applying such models to complicated flows. With that in mind, turbulence models have gained considerable acceptance in engineering applications, and have helped us solve a multitude of real life flow problems.

The epsilon equation as solved by FLUENT™ is given by:

(4.1.12)

$$\frac{\partial(\rho\epsilon)}{\partial t}+\frac{\partial(\rho\epsilon\overline{U}_i)}{\partial x_i}=\frac{\partial}{\partial x_j}\left[(\mu+\frac{\mu_t}{\sigma_\epsilon})\frac{\partial\epsilon}{\partial x_j}\right]+C_{1\epsilon}\frac{\epsilon}{K}\mu_t\frac{\partial\overline{U}_i}{\partial x_j}(\frac{\partial\overline{U}_i}{\partial x_j}+\frac{\partial\overline{U}_j}{\partial x_i})-C_{2\epsilon}\rho\frac{\epsilon^2}{K}$$

where $\sigma_\epsilon = 1.3$, $C_{1\epsilon} = 1.44$, and $C_{2\epsilon} = 1.92$. These constant have been calibrated from experimentally well studied turbulent shear flows, and have been found to be fairly suitable for other types of flows.

So far we have seen that the turbulent viscosity μ_t appears in both the Ƙ and ϵ equations, therefore we now present the final piece that will render the system of equations to be solved:

(4.1.13)

$$\mu_t=\rho C_\mu\frac{K^2}{\epsilon}$$

where $C_\mu = 0.09$ is a constant that guarantees consistency with the log-law of the wall.

It should be noted that the Ƙ and ϵ equations as presented above are only applicable at high Reynolds number conditions away from solid boundaries. Therefore, modifications ought to be introduced to bridge the gap between the near-wall regions and the fully turbulent region. This technique known as “wall-functions” will be discussed in a later section that deals with the numerical scheme employed by FLUENT™.

4.1.3 The RSM Closure

A transport equation for the Reynolds stress tensor itself can be derived by averaging and adding the Navier-Stokes equation one multiplied by the fluctuating velocity component u_i and the other by u_j. This results in the exact equation given below:

(4.1.14)

$$\frac{\partial(\rho\overline{u_i u_j})}{\partial t}+\frac{\partial(\rho\overline{U_k}\,\overline{u_i u_j})}{\partial x_k}=\underbrace{-\frac{\partial}{\partial x_k}[\rho\overline{u_i u_j u_k}+\overline{p(\delta_{kj}u_i+\delta_{ik}u_j)}]}_{i}+\frac{\partial}{\partial x_k}\underbrace{[\mu\frac{\partial(\overline{u_i u_j})}{\partial x_k}]}_{ii}$$

$$\underbrace{-\rho(\overline{u_i u_k}\frac{\partial\overline{U_j}}{\partial x_k}+\overline{u_j u_k}\frac{\partial\overline{U_i}}{\partial x_k})}_{iii}+\overline{p\underbrace{(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i})}_{iv}}-2\mu\underbrace{\overline{\frac{\partial u_i}{\partial x_k}\frac{\partial u_j}{\partial x_k}}}_{v}$$

The terms of the equation assert that the rate of change of Reynolds stress and its convection per unit volume is respectively balanced by: (i) turbulent transport, (ii) viscous diffusion, (iii) stress production, (iv) pressure strain, and (v) energy dissipation. This equation is somewhat appealing for turbulence modeling since only the terms (i), (iv) and (v) need to be modeled to grant closure to the RANS equations.

The models implemented in FLUENT™ are:

- term (i) is re-written in a gradient-like fashion, with $\sigma_k = 0.82$:

(4.1.15)

$$\frac{\partial}{\partial x_k}(\frac{\mu_t}{\sigma_k}\frac{\partial\overline{u_i u_j}}{\partial x_k})$$

- term (iv) is taken from the Quadratic Pressure-Strain Model, which emerged as superior to the more basic linear model (not discussed here) particularly in flows with streamline curvature. This observation was confirmed in this research as well, since we obtained better convergence with the quadratic model, hence the RSM results presented in this thesis are only those of the quadratic model. The term then becomes:

(4.1.16)

$$(iv)\equiv-(C_1\rho\epsilon+C_1^{'}P)b_{ij}+C_2\rho\epsilon(b_{ik}b_{kj}-\frac{1}{3}b_{mn}b_{mn}\delta_{ij})+(C_3-C_3^{'}\sqrt{b_{ij}b_{ij}})\rho k S_{ij}$$
$$+C_4\rho k(b_{ik}S_{jk}+b_{jk}S_{ik}-\frac{2}{3}b_{mn}S_{mn}\delta_{ij})+C_5\rho k(b_{ik}\Omega_{jk}+b_{jk}\Omega_{ik})$$

where

$$b_{ij}=-\left(\frac{-\rho\overline{u_i u_j}+\frac{2}{3}\rho k\delta_{ij}}{2\rho k}\right)$$ Reynolds stress anisotropy tensor (4.1.17)

$$S_{ij}=\frac{1}{2}(\frac{\partial\overline{U_i}}{\partial x_j}+\frac{\partial\overline{U_j}}{\partial x_i})$$ Mean rate-of-strain tensor (4.1.18)

$$\Omega_{ij}=\frac{1}{2}(\frac{\partial\overline{U_i}}{\partial x_j}-\frac{\partial\overline{U_j}}{\partial x_i})$$ Mean rate-of-rotation tensor (4.1.19)

and the constants: C_1 = 3.4, C_1' = 1.8, C_2 = 4.2, C_3 = 0.8, C_3' = 1.3, C_4 = 1.25, C_5 = 0.4

- finally, term (v) is called the dissipation rate tensor ϵ_{ij} and is modeled as:

(4.1.20)

$$\epsilon_{ij}=\frac{2}{3}\delta_{ij}(\rho\epsilon+Y_M)$$

with the scalar dissipation rate ϵ evaluated from its own transport equation similar to its counterpart in the k-epsilon model.

As revealed above, the Reynolds Stress model requires the solution of seven additional differential equations (six for the stresses and one for ϵ), which when combined with the four standard RANS equations makes the computational expense even more costly.

4.1.4 The Numerical Scheme

FLUENT's finite volume “segregated” double precision solver was used to arrive at a steady state solution for the virtual impactor simulations. The justification for performing steady state as opposed to transient calculations stems from the fact that in reality aerosol samplers are operated for significantly long periods of time, and the analysis of their performance is not particularly sought during the startup process. Therefore, the actual equations that are numerically integrated do not have the time dependent derivative, shown for completeness in the above equations.

The numerical algorithm solves the governing integral equations for conservation of mass, momentum, energy and turbulence quantities on a control volume basis for each individual computational cell in the domain. The method is called segregated since the equations are solved separately one at a time in a decoupled iterative procedure to achieve convergence of the mathematically coupled non-linear equations. First, the momentum equation is solved with the available values of the unknowns to produce an updated value for the velocity vector. If the new velocity vector does not satisfy the continuity equation, a “Poisson-type” equation is solved for the pressure and velocity corrections so that the continuity equation is satisfied. Equations for the energy and turbulence quantities are subsequently solved and a convergence check with a specified tolerance is made at the end of the loop. The algorithm employs an implicit scheme to linearize the discrete equations, this results in a system of linear equations that cover all the cells in the domain, thus allowing the sought after unknown variable to be solved for simultaneously at all cells. The details of the algebraic equations solver will not be discussed here since it is common practice in CFD codes, however we will discuss some of the methodologies invested in the intermediate steps presented above, particularly the discretization scheme, since it was found to have an impact on the final flow field solution.

4.1.4a Discretization

The governing differential equations presented in the previous sections can also be written in their integral form, simply by derivation from first principles. The finite volume method transforms those integral equations into discrete counterparts so that they can be solved numerically. To illustrate the role of discretization on the method, consider the steady state transport equation of a scalar quantity ϕ in a control volume V, with a diffusion coefficient Γ_ϕ , and a source term S_ϕ :

$$\oint \rho\phi\vec{U}d\vec{A}=\oint \Gamma_\phi\nabla\phi d\vec{A}+\int_V S_\phi dV \tag{4.1.21}$$

In discrete space, the integrals are replaced by summations, and the equation is discretized as such [20]:

$$\sum_f^N \rho_f\phi_f\vec{U}_f\circ\vec{A}=\sum_f^N \Gamma_\phi|\nabla\phi|_n\circ\vec{A}+S_\phi V \tag{4.1.22}$$

where ϕ_f is the convected quantity through face f, N is the number of faces per cell, $\vec{A}$ is the area vector of face f, and $|\nabla\phi|_n$ is the magnitude of the gradient normal to face f. The choice of the discretization scheme enters during the evaluation of the face value ϕ_f since its value is only stored at the cell center. Either a first order or a second order "upwind" scheme is employed for the task. The former scheme assumes that the cell center value represents an average that is valid anywhere in the cell. Consequently, values for the face fluxes are obtained from the cells upstream of the flow direction. For flows with streamline curvature this contributes more to the numerical diffusion and can compromise the overall accuracy. In the second order scheme, the ϕ_f value is computed using a truncated Taylor series expansion about the cell center as given below:

$$\phi_f = \phi + \nabla\phi \circ \Delta\vec{s} \tag{4.1.23}$$

where ϕ and $\nabla\phi$ are the value of the scalar and its gradient in the upstream cell center, and $\Delta\vec{s}$ is the displacement vector from the upstream cell center to the face centroid. Finally, the gradient is computed via the discrete form of the divergence theorem as follows:

$$\nabla\phi = \frac{1}{V}\sum_{f}^{N} \tilde{\phi}_f \vec{A} \tag{4.1.24}$$

where $\tilde{\phi}_f$ is the average of ϕ from the two cells adjacent to each face.

The procedure outlined above can be generalized and applied to the discretization of the momentum and continuity equation. As mentioned before, the equations are solved sequentially, therefore, there has to be a mechanism to insure that the pressure value obtained from the momentum equation satisfies continuity. This pressure-velocity coupling feature is achieved using the Semi-Implicit Method for Pressure -Linked Equations (SIMPLE) algorithm [20].

4.1.4b Wall Functions

Wall functions are semi-empirical relations used to estimate the mean velocity and turbulence quantities near the wall in a turbulent flow. In essence, they establish a boundary condition for the numerical solution of the closure equations at the first mesh point away from the wall.

Experiments have shown that the wall region in a bounded turbulent flow consists of three layers: a viscous sublayer dominated by viscosity, a buffer layer where both viscosity and the Reynolds shear stress dominate the dynamics, and a fully turbulent or logarithmic layer. In the viscous sublayer, the mean velocity is characterized by a linear dependence on the distance from the wall, and from its name, the logarithmic layer represents the mean velocity by a universal logarithmic law. No formal expression of the mean velocity is in existence for the interestingly arduous buffer layer. FLUENT™ relies on the value of a dimensionless wall unit y* to determine the appropriate equations to use. The definition of y* is given below:

$$y^* = \frac{\rho C_\mu^{\frac{1}{4}} K^{\frac{1}{2}} y}{\mu} \tag{4.1.25}$$

where y is the distance from a point in the flow to the wall.

Relying on viscosity predominance, the dimensionless mean velocity when y* < 11.225 is given by:

$$\bar{U}^* = y^* \tag{4.1.26}$$

and for larger values of y* (< 30 to 60), the log-law is used:

$$\bar{U}^* = \frac{1}{\kappa} \ln(E y^*) \tag{4.1.27}$$

where κ is the von Karman constant (0.42), and E an empirical constant

(9.81). The mean velocity can then be determined from U* using its definition:

$$\overline{U}^{\ *} = \frac{\overline{U} C_{\mu}^{\frac{1}{4}} K^{\frac{1}{2}}}{\tau_w / \rho} \tag{4.1.28}$$

where $\tau_w = \mu \frac{\partial \overline{U}}{\partial y}(0)$ is the wall shear stress. A similar decomposition is done for the thermal boundary layer to obtain the temperature variation near the wall in the compressible flow simulation [20]. Turbulence properties, on the other hand, are determined differently. The kinetic energy equation is integrated all the way to include the near wall mesh points, with an imposed boundary condition on its derivative in the wall normal direction $(\frac{\partial K}{\partial \vec{n}} = 0)$. Additionally, the local equilibrium hypothesis is invoked so that the generation and dissipation of K are assumed equal. The epsilon equation is not solved for the near wall cells, but rather ϵ is computed from:

$$\epsilon = \frac{C_{\mu}^{\frac{3}{4}} K^{\frac{3}{2}}}{\kappa y} \tag{4.1.29}$$

and the generation term in the K equation is replaced by: $\tau_w \frac{\partial \overline{U}}{\partial y}$.

Wall functions have become the de facto for many industrial flow problems due to their economical savings on the computational cost.

They are mostly adequate for shear flows with little or no pressure gradients, however their accuracy breaks down when the flow conditions depart too much from the ideal circumstances for which they were derived, such as low Reynolds number effects or boundary layer separations. For the sake of this research, such phenomena has been ruled out as being highly influential, thus justifying their use.

4.2 Particulate Flow

4.2.1 Particle Equation of Motion

The first formal derivation of the equation of motion for an isolated rigid sphere was postulated by Maxey and Riley [23]. The authors proposed an equation similar to the original BBO equation, but included extra terms to account for non-uniformity of the flow field, which can be written as [10]:

$$m_p \frac{dU_p}{dt} = m_p \frac{(U - U_p)}{\tau_p} + m_f \frac{D\vec{U}}{Dt} + \frac{1}{2} m_f \left(\frac{D\vec{U}}{Dt} - \frac{dU_p}{dt} \right) + 6 d_p^2 (\pi \rho \mu)^{\frac{1}{2}} \int_{tp0}^{tp} \frac{d/d\tau(\vec{U} - \vec{U}_p)}{(t-\tau)^{1/2}} d\tau + (m_p - m_f)\vec{g} + \frac{1}{2}(\pi \rho d_p^2) C_L \vec{L} |\vec{U} - \vec{U}_p|^2 \tag{4.2.1}$$

where m_p and m_f are the mass of the particle and fluid respectively, and $\vec{g}$ is the gravitational acceleration. The equation represents a balance between the forces acting on the particle. The inertia force on the left hand side is balanced by, respectively, the viscous and pressure drag, fluid pressure gradient and viscous stresses, inertia of virtual mass, the

Basset “history” force, buoyancy, and Saffman's lift force (C_L is the lift coefficient, and $\vec{L}$ is the direction cosine vector). Two critical assumptions went into the derivation of the Maxey and Riley equation: (1) the particle diameter is small compared to the Kolmogorov length scale, and (2) the particle or relative Reynolds number Re_p is small compared to 1, where $Re_p = \frac{d_p|\vec{U}-\vec{U}_p|}{\nu}$. The first condition is mostly true in sub-micron particle-laden flows, however as we shall see in the next chapter, it is quite often the case that the particle Reynolds number exceeds one.

For aerosol flows, where the density of the dispersed phase is significantly larger than that of the fluid phase ($\rho_p/\rho \approx 10^3$), the equation takes on a simpler form [1,12,13,24,25,26,33]:

$$\frac{dU_p}{dt} = C_D \frac{Re_p}{24} \frac{(U-U_p)}{\tau_p} + \vec{g} \tag{4.2.2}$$

where C_D is the drag coefficient. The effect of gravity can be further neglected considering the sufficiently small size of the particles ($\tau_p \approx 10^{-7}$).

McLaughlin [27] studied the deposition of dense aerosol particles in a turbulent channel flow. He confirmed the insignificance of the Basset force. In addition, the Saffman force was found to have virtually no effect outside the viscous sublayer but contributed to the inertial deposition and accumulation inside the viscous layer. Although, such phenomena is intriguing to investigate in the virtual impactor design, we are currently limited by the RANS solution to the fluid motion to be able to accurately resolve the viscous sublayer in order to study such an effect. For the above reasons, we focus our attention on the particle equation of motion governed predominantly by the drag force:

$$\frac{d\vec{U}_p}{dt}=C_D\frac{Re_p}{24}\frac{(\vec{U}-\vec{U}_p)}{\tau_p} \tag{4.2.3}$$

4.2.2 The Drag Coefficient

Since the drag force dominates the dynamics of the aerosol particle motion inside the virtual impactor, it is crucial that an appropriate formulation for the drag coefficient is used to reliably predict the particle trajectory. The particles are expected not to follow the Stokes drag regime for two main reasons: (i) Their entrapment in a high speed curved flow structure as they exit the acceleration nozzle, and (ii) their sub-micron size which makes them susceptible to ***slip***. The first condition was confirmed by monitoring the particle Reynolds number for different size particles during their flight, and will be discussed further in the next chapter. The second condition can be understood by looking at the Knudesn number Kn = λ/d_p, where λ is the mean free path of air. Crowe classifies the different particle-fluid flow regimes based on values of the Knudsen and relative Reynolds and Mach numbers. The relative Mach number is defined as the ratio of the magnitude of the slip velocity to the speed of sound:

$$Ma_r = \frac{|\vec{U} - \vec{U}_p|}{v_{sound}}. \tag{4.2.4}$$

Continuum	Kn < 10^{-3}	$Ma_r < 0.01 Re_p^{1/2}$
Slip Flow	10^{-3} < Kn < 0.25	$0.01 Re_p^{1/2} < Ma_r < 0.1 Re_p^{1/2}$
Transitional Flow	0.25 < Kn < 10	$0.1 Re_p^{1/2} < Ma_r < 3 Re_p$
Free Molecule Flow	Kn >10	$Ma_r > 3 Re_p$

Table 4.1: Classification of flow regimes from [1]

The mean free path may be calculated knowing the temperature (°K) and pressure (KPa) [28]:

$$\lambda = 2.15 \mu \frac{T^{1/2}}{P} \quad (\mathrm{cm}) \tag{4.2.5}$$

Evidently, for the particle diameter range (0.1-0.4 μm), the Knudsen number takes on values between 0.65 and 0.16, assuming air at STP. Clearly, those limits are prone to slightly change in different regions in the flow, for instance at the high speed nozzle where the pressure is below atmospheric, and the temperature is lower than ambient, however, those variations will not alter the underlying flow regimes, which are

"Slip" and "Transitional", noting that the relative Mach number, as will be shown later, is always less than $3Re_p$. Unfortunately, the flow in the transition regime is poorly understood, and traditionally the practice is to use the same correction to the Stokes drag coefficient, commonly known as the Cunningham correction factor C_c, to account for such non-continuum effects. Therefore, the slip C_D becomes:

$$C_D = \frac{C_{D,\mathrm{Stokes}}}{C_c} = \frac{24/Re_p}{C_c} \tag{4.2.6}$$

where C_c is given by:

$$C_c = 1 + Kn[2.514 + 0.8\exp(-0.55/Kn)] \tag{4.2.7}$$

A useful form for the slip correction that shows the dependence on pressure was given by Hinds [29] as:

$$C_c = 1 + \frac{2}{Pd_p}[6.32 + 2.01\exp(-0.1095\,Pd_p)] \tag{4.2.8}$$

where P is the absolute pressure in cm Hg, and d_p is the particle diameter in μm.

Compressibility effects become significant for $Ma_r > 0.2$, and we found that for some of the bigger particles, Ma_r does in fact exceed this limit in the virtual impaction region of the device. Therefore, to account for compressibility effects we employ a semiempirical expression proposed originally by Crowe [30], and later validated experimentally in a microballistic range [31]. The expression is valid for relative Reynolds numbers less than 100 and Mach numbers less than 2, which makes it suitable for use in the virtual impactor, since the observed values for those two quantities conformed to that range. The equation for the "compressible" C_D is then given by [1]:

$$C_D = 2 + (C_{Do} - 2)\exp(-3.07\sqrt{\gamma}\, g(Re_p) Ma_r / Re_p) + \frac{h(Ma_r)}{\sqrt{\gamma}\, Ma_r}\exp\left(\frac{-Re_p}{2\, Ma_r}\right) \tag{4.2.9}$$

where $\gamma = 1.4$ is the ratio of specific heats of the gas, $g(Re_p)$ and $h(Ma_r)$ are the following functions:

$$g(Re_p) = \frac{1 + Re_p(12.278 + 0.548 Re_p)}{1 + 11.278 Re_p} \quad , \quad h(Ma_r) = \frac{5.6}{1 + Ma_r} + 1.7\sqrt{\frac{T_p}{T_f}}$$

where T_p and T_f are the particle and fluid temperatures, respectively (taken to be equal hereinafter). C_{Do} is the standard drag coefficient for incompressible flow which can take several forms. For incompressible flow, the most widely used correlation is that of Schiller and Nauman, which fits the standard drag curve with less than 5% error, and is valid for $Re_p \leqslant 800$:

$$C_D = \frac{24}{Re_p}(1 + 0.15 Re_p^{0.687}) \tag{4.2.10}$$

FLUENT™ employs a more general correlation applicable to non-spherical particles but closely matches the above formula when used for spherical particles, that is given by:

$$C_D=\frac{24}{Re_p}(1+0.1862\,Re_p^{0.6529})+\frac{0.4373\,Re_p}{7185.4+Re_p} \tag{4.2.11}$$

For the flow conditions studied in this thesis, it was found that the particle Reynolds number does not gravely exceed unity, hence it would suffice to use the theoretical result of Oseen, who extended Stokes analysis and arrived at a drag coefficient valid for $Re_p \leqslant 5$, which is given by:

$$C_D=\frac{24}{Re_p}(1+\frac{3}{16}Re_p) \tag{4.2.12}$$

The compressible flow results presented in Chapter 6 utilize this Oseen formulation to substitute for C_{Do} in Equation (4.2.9). The inadequacy of the Stokes drag coefficient for large Re_p values can be seen in the following figures. Figures 4.1 and 4.2 show the dependence of the drag coefficient on the relative Mach number. The top axis also shows the relative Reynolds number which can be computed for a known particle size.

First, it is clear that the Stokes regime underpredicts the drag coefficient beyond $Re_p = 0.1$ for an incompressible flow when compared to the Oseen curve. Second, the effect of slip is pronounced as a significant reduction in the drag coefficient. Also, the compressible curve remarkably collapses onto the Stokes-Cunningham curve at least for higher values of the relative Mach number. This does not remain true for a bigger particle as shown in Figure 4.2. As can be seen, for the same magnitude of the Mach number, a bigger particle will have a higher magnitude for the Reynolds number, thus a lower drag coefficient regardless of the formulation used. Moreover, at high Mach numbers, a bigger particle will experience a higher drag from the compressible formulation than from the Stokes-Cunningham correction. The effect of the drag coefficient correlation on the particle trajectories and collection efficiency curve will be discussed in the next chapter.

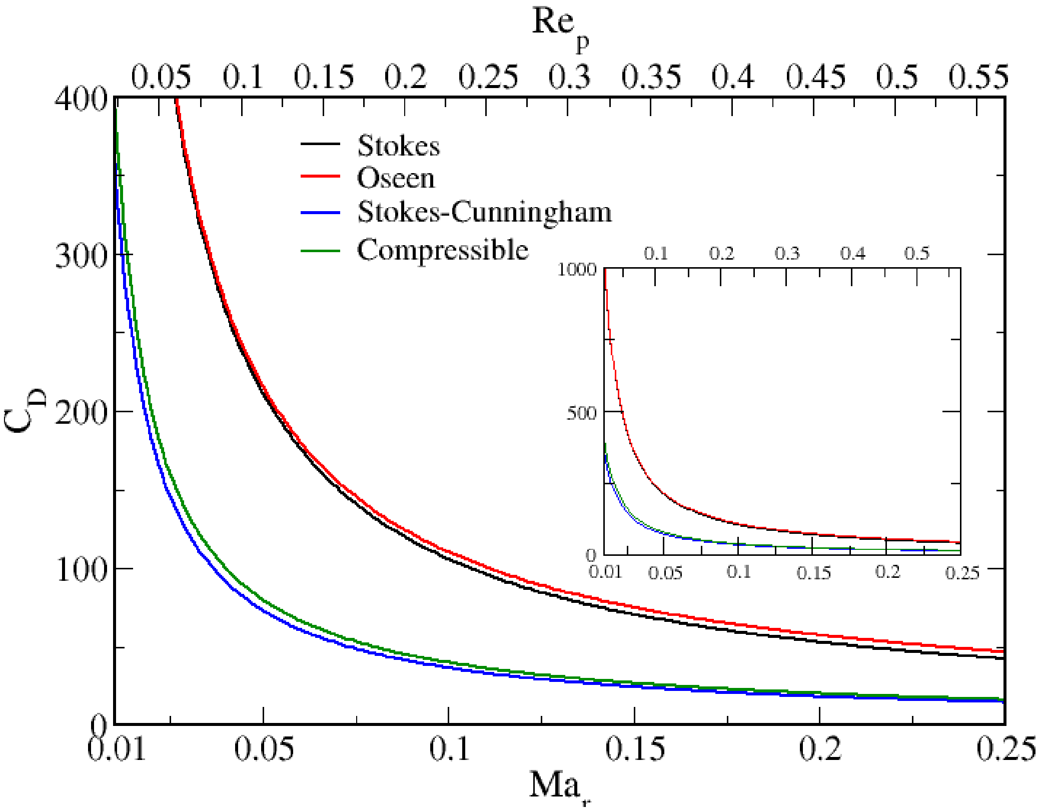

Figure 4.1: Drag coefficients for a 0.1 μm particle, Stokes = 24/Re$_p$, Oseen = eqn. 4.2.12, Stokes-Cunningham = eqn. 4.2.6, Compressible = eqn. 4.2.9 (insert shows entire y-axis scale)

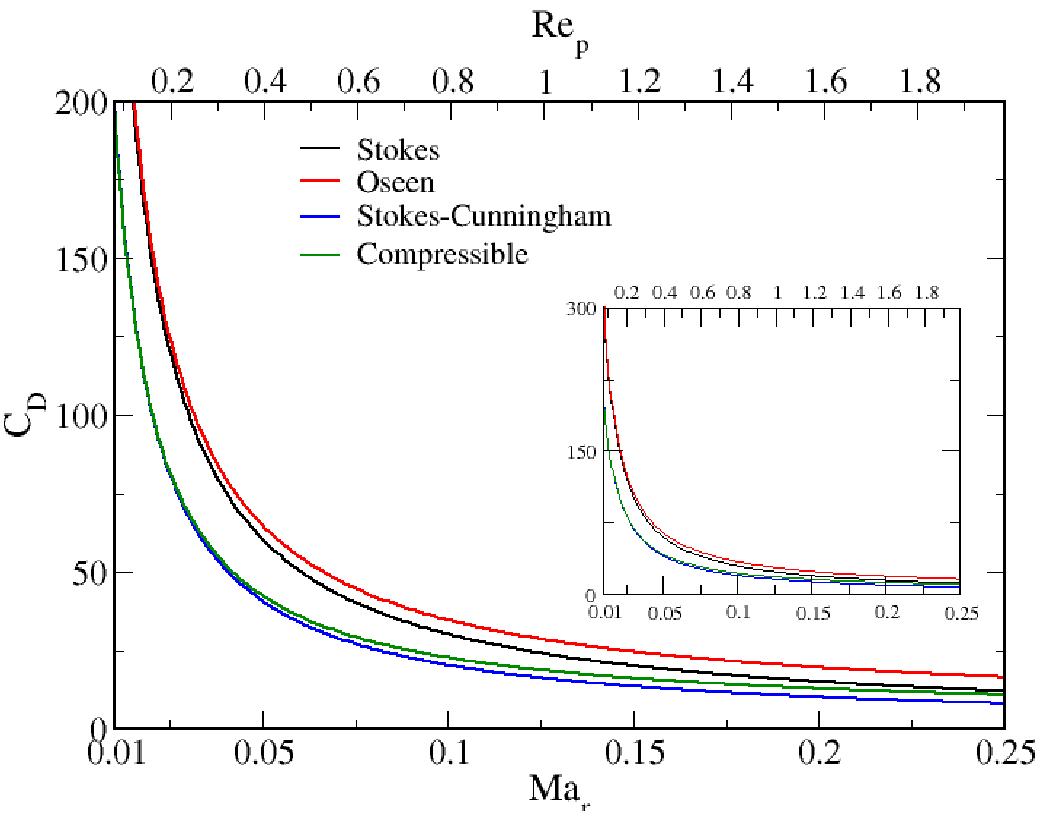

Figure 4.2: Drag coefficients for a 0.35 μm particle, Stokes = 24/Re$_p$, Oseen = eqn. 4.2.12, Stokes-Cunningham = eqn. 4.2.6, Compressible = eqn. 4.2.9 (insert shows entire y-axis scale)

4.2.3 Random Walk Model: Stochastic Tracking

Inherent to the RANS solution is the decomposition of the velocity vector into a mean and fluctuating component. As shown in the beginning of the chapter, the mean velocity is directly predicted. A turbulence closure model, however, is inevitable to arrive at second moment correlations. Undoubtedly, the more robust the model, the more accurate those moments are. Equation (4.2.3) requires the fluid velocity vector $\vec{U}$ in order to integrate and solve for the particle velocity $\vec{U}_p$. One can neglect the fluctuating part and simply set $\vec{U}=\vec{\bar{U}}$, which is referred to as mean tracking [18,26]. The alternative, of course, is to include the fluctuating component, $\vec{U}=\vec{\bar{U}}+\vec{u}$, and evaluate it based on the local turbulence intensity of the fluid. The first such model engaged in conjunction with the K-ϵ equations is the "eddy-lifetime" model [32], and is incidentally the model implemented in FLUENT^{TM}. The particle is made to interact with a turbulent eddy over a time interval which is the minimum of a typical eddy lifetime, τ_e, and a characteristic particle residence time in the eddy, τ_r. The eddy lifetime is taken to be a random energy dissipation time scale:

$$\tau_e=-C_L\frac{K}{\epsilon}\log(r) \tag{4.2.13}$$

where r is uniformly distributed in (0,1), and C_L is the constant 0.15 or 0.3 for either the K-ϵ or Reynolds stress model, respectively. The particle transit time through the eddy is further estimated using the eddy dissipation length scale L_e:

$$\tau_r=-\tau_p\ln(1-\frac{L_e}{\tau_p|\vec{U}-\vec{U}_p|}) \tag{4.2.14}$$

where $L_e = C_\mu^{3/4} K^{3/2}/\epsilon$. Needless to say, when a solution to the above equation is not feasible, the interaction time is taken to be τ_e.

The fluctuating velocity of the eddy, on the other hand, is assumed to satisfy a Gaussian distribution such that:

$$u_i = \alpha \sqrt{\overline{u_i^2}} \tag{4.2.15}$$

where α is a normally distributed random number, and i=1, 2, or 3 for each velocity direction. The same value of α is used for the three fluctuating components. As alluded to in a previous section, the K-ϵ model yields equal quantities for the root-mean-square velocities,

$\sqrt{\overline{u_i^2}} = \sqrt{\frac{2}{3}\mathrm{K}}$, thus forcing an isotropic condition on the generated fluctuating velocities u_i, which limits the validity of the stochastic approach. The Reynolds stress model, however, solves the time averaged equations for the independent Reynolds stress tensor components, thus the stochastic method can benefit from the computed normal components of the stress tensor to generate three distinct velocity fluctuations. The methodology described above is still employed, the only advantage, however, is that the root-mean-square velocities that enter into Equation (4.2.15) will, by virtue of solving Equation (4.1.14), have distinct values, which are the outcome of solving their transport equation. The supporting argument for the stochastic approach was that if a statistically large sample of particle trajectories are computed, then their ensemble average would represent particle dispersion.

4.2.4 Numerical Solution of the Particle Equation

4.2.4a FLUENT'S Implementation

Starting from Equation (4.2.3), and substituting the expressions for the

particle relaxation time, $\tau_p = \frac{\rho_p d_p^2}{18\mu}$ and Re_p, we can write:

$$\frac{d\vec{U}_p}{dt} = \frac{3\rho|\vec{U}-\vec{U}_p|}{4\rho_p d_p} C_D(\vec{U}-\vec{U}_p) = \frac{1}{\beta}(\vec{U}-\vec{U}_p) \tag{4.2.16}$$

where $\beta = \frac{4\rho_p d_p}{3\rho|\vec{U}-\vec{U}_p|C_D}$.

The discretized form of Equation (4.2.16) is then taken to be (dropping the vector notation):

$$\frac{U_p^{n+1}-U_p^n}{\Delta t} = \frac{1}{\beta}(\tilde{U}-U_p^{n+1}) \tag{4.2.17}$$

where $\tilde{U} = U^n + \frac{\Delta t}{2} U_p^n . \nabla U^n$ is an estimate or ***linearly-interpolated*** value of the fluid velocity at the particle position, n is the time index, and U^n is the known fluid velocity at an Eulerian mesh point closest to the particle. The coefficient β is a constant evaluated at the previous time step. The discrete equation reveals that the method is a first order accurate, ***O*** (Δt), “Euler” scheme. Re-arranging Equation (4.2.17) yields:

$$(1+\frac{\Delta t}{\beta})U_p^{n+1} = U_p^n + \frac{\Delta t}{\beta}\tilde{U} \tag{4.2.18}$$

Since the right hand side of Equation (4.2.18) consists only of variables that are evaluated at time n, the method is deemed explicit, and the equation can be integrated forward in time from a known initial condition at n = 0.

4.2.4b AeroTrack: The Developed Code

A custom developed FORTRAN 90 code was written to allow for more control over the accuracy of the numerical algorithm. The basic features of the new code are summarized below:

1. Employs a second order accurate trapezoidal integration scheme.
2. Uses a multi-variate scattered point interpolation scheme with third order precision.
3. Provides more control over the choice of the drag law.
4. Flexibility to work with any exported CFD solution, and allows for "on the fly" monitoring of critical flow variables such as the slip velocity.

The details of the program are now presented.

4.2.4b.1 Trapezoidal Scheme

In order to obtain trajectory information, Equation (4.2.16) is integrated along with the particle position vector to yield the system of Ordinary Differential Equations:

$$\frac{d\vec{X}_p}{dt}=\vec{U}_p \tag{4.2.19}$$

$$\frac{d\vec{U}_p}{dt}=\frac{1}{\beta}(\vec{U}-\vec{U}_p) \tag{4.2.20}$$

with a known initial condition: $\vec{X}_p^o=\vec{X}_p(0)$, and $\vec{U}_p^o=\vec{U}_p(0)$. The initial particle velocity is always taken to be that of the fluid at $\vec{X}_p(0)$. To simplify the notation, the ODE system is written in the general mathematical form:

$$\frac{dy}{dt}=f(t,y) \tag{4.2.21}$$

where y is now the vector of unknowns, and f is the vector of functions. We denote $y_n = y(t_n)$ and n = 0, 1, 2, ... The trapezoidal scheme for integrating (4.2.21) becomes [52]:

$$\frac{y_{n+1}-y_n}{\Delta t}=\frac{1}{2}\left[f(t_n,y_n)+f(t_{n+1},y_{n+1})\right]+\boldsymbol{O}(\Delta t^3) \tag{4.2.22}$$

with a global integration error proportional to Δt^2. Re-arranging and setting equal to zero:

$$y_{n+1}-y_n-\frac{\Delta t}{2}\left[f(t_n,y_n)+f(t_{n+1},y_{n+1})\right]=0 \tag{4.2.23}$$

which results in an algebraic system of equations that is solved by Newton's method at each time step (See Appendix A).

4.2.4b.2 Multi-Variate Scattered Interpolation

4.2.4b.2-i Motivation

The issue of finding the fluid velocity at the position of a traveling particle in a numerical flow solution always arises because, in general, the particle can move anywhere in the computational domain, and will not necessarily coincide with the fixed grid points or cells. The ideal solution to this problem is to perform a coupled direct numerical simulation of the fluid and particle motion that resolves the flow around each particle by imposing the no-slip boundary condition at its surface, and including the hydrodynamic forces and torques exerted by the fluid. Clearly, this approach is far from becoming an engineering tool at this point due its magnificent computational requirement, but appears to be the only theoretical tool capable of providing insight into two-way coupling models [34]. The practical option which is widely used today, is to employ an interpolation scheme on the Eulerian velocity field that delivers a fluid velocity to the Lagrangian particle tracking algorithm.

A number of studies that deal with the uncertainties associated with interpolation schemes have been done for direct calculations. Kontomaris et al. [35] explored the accuracy of various interpolation schemes for the case of a turbulent channel flow, and made recommendations that were successfully adopted and used by other researchers [36]. According to Kontomaris, the interpolation error is, by far, the most serious error incurred during a typical particle trajectory calculation. The magnitude of the error, of course, is a function of the underlying velocity field, the grid resolution, and the interpolation method. The authors confer that linear interpolation performs the worst when compared to cubic spline or third order Lagrangian interpolation. The conclusions reached by Kontomaris et al. are certainly applicable to this research, however, it should be noted that their calculations were performed in a Spectral Simulation environment, and that their trajectory computations were limited to fluid particles. Given the nature of a RANS simulation, and the fact that employing a drag coefficient introduces empirical uncertainties, we expect the interpolation error to be one among other serious errors. Typically in a DNS, the geometry is relatively simple (a box to represent a channel or a boundary layer), so the numerical grid consists of a perfectly aligned lattice of mesh points, which makes the use of higher order polynomial interpolation schemes feasible. Wang & Squires [33], for instance, utilize a fourth-order Lagrange polynomial, whereas, Ahmed & Elghobashi [37], make use of a Hermitian cubic interpolation scheme. In complex geometries on the other hand, the computational meshes are often irregularly shaped and the grid points are usually unequally spaced or scattered, which makes the three dimensional interpolation problem more difficult [38]. Most recent engineering work that deals with the simulation of particle-laden flows in real-life equipment, for example, a cyclone [39] or a stirred tank [40], relies on linear interpolation. While such an approach may present some advantages as far as ease of implementation and computational speed, it is prone to augment the numerical error. For the virtual impactor, we found that linear interpolation increased the relative error (defined later) in interpolating the velocity field by one order of magnitude when compared to the proposed interpolation scheme.

4.2.4b.2-ii Theory

The problem of fitting a smooth surface to a scattered set of data points is encountered in many scientific applications such as environmental modeling, geographical and meteorological studies, and computer aided design (CAD). The two mainstream and most accurate interpolation methods are triangulation-based or weighted-inverse-distance methods. Triangle based methods appear to work well in interpolating data on a plane, however their extension to three or more independent variables is obstructed by the complexity of the algorithms and storage requirements. Shepard in 1968 [41] first introduced his inverse distance bi-variate function as a viable candidate for interpolating sets of scattered data with local accuracy, and showed the potential of this method for the tri-variate case. The original strategy, although successful in accomplishing its objective, suffered from a number of shortcomings, which led to a series of modifications and enhancements that resulted in the form it is used in this research. To illustrate the concepts of modified Shepard's methods, we briefly discuss the theory behind the original approach. The discussion is first given for functions in $\mathbb{R}^2$ as it will make the transition to $\mathbb{R}^3$ conceptually transparent. Consider a set of N distinct data nodes $p_i(x_i,y_i)$ where the value of the function $f(p_i)$ is precisely known, and $1 \le i \le N$. We seek an interpolant function F that fits the given nodes exactly. F is taken to be:

$$F(p)=\sum_{i=1}^{N} f(p_i)\,u_i \tag{4.2.24}$$

where u_i are cardinal functions defined analogous to the Lagrange interpolating polynomial as:

$$u_i(p)=\prod_{j=1,\,j\neq i}^{N} \frac{\phi(p,p_j)}{\phi(p_i,p_j)} \tag{4.2.25}$$

In order for $u_i(p_j)$ to satisfy the cardinality property $\{u_i(p_j)=\delta_{ij}\}$, ϕ must satisfy the condition: $\phi(p,q) = 0$ if and only if $p = q$. Therefore, ϕ is almost always chosen to be the square of the Euclidean distance between p and q, $\phi = |p-q|^2$. Putting all of this together, the final form for F becomes:

$$F(x,y)=\sum_{i=1}^{N} f(x_i,y_i) \prod_{j=1,\,j\neq i}^{N} \frac{(x-x_j)^2+(y-y_j)^2}{(x_i-x_j)^2+(y_i-y_j)^2} \tag{4.2.26}$$

A slight variations to this formulation is given by Kincaid and Cheney [42], where a new cardinal function is selected. Their interpolant function is given by:

$$F(p)=\sum_{i=1}^{N} f(p_i)\frac{v_i(p)}{v(p)} \tag{4.2.27}$$

where $v_i(p)=\prod_{j=1,j\neq i}^{N}\phi(p,p_j)$, and $v(p)=\sum_{i=1}^{N}v_i(p)$. This version of Shepard's method is more desirable because it attributes to F(p) certain characteristics of the function being interpolated. This method is the one we implemented as a reference point for the other enhanced versions; namely the Franke-Nielson [43] and the Renka [44] modifications. Franke and Nielson define the interpolant as:

$$F(x,y)=\sum_{k=1}^{N} Q_k(x,y)\frac{W_k(x,y)}{\sum_{i=1}^{N}W_i(x,y)} \tag{4.2.28}$$

where the nodal function Q_k is a bivariate quadratic polynomial that reproduces the true value of f at the nodes (x_k,y_k), and fits the data values on a set of nearby nodes in a weighted least square sense. The unnormalized weights are inverse distance functions evaluated as:

$$W_k(x,y)=\left[\frac{(R_w-d_k)_+}{R_w d_k}\right]^2 \quad \text{for } (R_w-d_k)_+ = \begin{cases} R_w-d_k & \text{if } d_k<R_w \\ 0 & \text{if } d_k\geq R_w \end{cases} \tag{4.2.29}$$

where $d_k(x,y)$ is the Euclidean distance between (x,y) and (x_k,y_k), and R_w is the radius of influence about node (x_k,y_k). The radius of influence insures that the data at (x_k,y_k) only influences interpolated values at points that lie within this radius. The authors show that not only does F(x,y) interpolate the data, but also maintains the local shape properties of the nodal functions (has consistent first partial derivatives) with quadratic precision. The nodal function Q_k is defined by:

$$Q_k(x,y)=c_{k1}(x-x_k)^2+c_{k2}(x-x_k)(y-y_k)+c_{k3}(y-y_k)^2+c_{k4}(x-x_k)+c_{k5}(y-y_k)+f_k \tag{4.2.30}$$

where the coefficients c_{k1} to c_{k5} minimize the residual

$$R^2=\sum_{i=1,i\neq k}^{N}\omega_i(x_k,y_k)[Q_k(x_i,y_i)-f_i]^2 \quad \text{for} \quad \omega_i(x,y)=\left[\frac{(R_q-d_i)_+}{R_q d_i}\right]^2 \tag{4.2.31}$$

where d_i is the distance between nodes i and k, and R_q is another radius of influence about node k, which limits the contribution to the least squares fit from nodes that lie further away. The selection of the radii R_w and R_q is critical to the level of accuracy achieved by this modified Shepard's method. Intuitively, it is desired for R_q to be greater than R_w, so as to allow the nodal function to approximate as many as the nearby nodes as possible. Rather than specifying values for the radii, Franke and Nielson propose the use of a fixed number of data points, N_q and N_w, that are anticipated to lie within the radii of influence. The relations to compute the radii are then given by:

$$R_q = \frac{D}{2}\sqrt{\frac{N_q}{N}}, \quad R_w = \frac{D}{2}\sqrt{\frac{N_w}{N}}, \tag{4.2.32}$$

where D = maximum distance between any two data points. The authors demonstrate the accuracy of the method by conducting a number of interpolation tests on representative functions, and recommend a value of $N_q = 18$ for somewhat uniformly distributed data, and also find that the ratio $N_q/N_w \approx 2$ is useful.

4.2.4b.2-iii Proposed Algorithm

Renka [44] refined the Franke-Nielson approach to increase accuracy and reduce the computational cost associated with finding the nearest-neighbor nodes. He proposed allowing the radii R_w and R_q to vary with each node k. His method also makes use of a fixed number of nodes N_w and N_q, but the framework with which the radii are computed is fundamentally different. R_w is taken to be the distance from node k to the j^{th} closest node subject to the conditions that $j > N_w$ and the j^{th} node is significantly more distant than the $(j-1)^{st}$ node. Renka fails to elaborate on this idea, or give any details about how it is implemented, so for the sake of clarity, we present the pseudocode in Figure 4.3 to depict its implementation in AeroTrack. R_q is computed in a similar fashion. Renka obtained improved accuracy over the traditional Franke-Nielson method with values of $N_q = 13$, and $N_w = 19$. Note that these values should not be compared to those of Franke and Nielson, since, despite their common labels, they are used in a completely different context.

```
Outer Loop: For $k = 1$ to $N_b$

    Inner Loop: For $j = 1$ to $N_b$
        Compute distance between node k and all other nodes j → $d_{kj}$
    End Inner Loop

    Sort $d_{kj}$ in ascending order → $sd_{kj}$

    Dispose of the distances up to $N_w$ and compute relative distances:

        Inner Loop: For $i = N_w+1$ to $N_b-1$
                        $RelD(i) = sd_{kj}(i+1) - sd_{kj}(i)$
        End Inner Loop

    Compute maximum relative distance, note its location in the array:

        $mRelD = \max (RelD); locm$

    Define: $R_w = sd_{kj} (locm+1)$

End Outer Loop
```

Figure 4.3: Algorithm for computing dynamic radii

Perhaps the algorithm for finding the dynamic radii presented above is not exactly what Renka intended. Nonetheless, given the circumstances in which we need to use the interpolation scheme, namely in a CFD/particle tracking program, we determined that such an algorithm performs well on both the accuracy and efficiency scale. This will be evident in the following chapter where we subject this method to a number of interpolation tests. Furthermore, the use of a new variable N_b

is only pertinent to AeroTrack and not to the Renka implementation. In a CFD solution, the number of data points is normally very large (equal to the number of computational cells), so it would be highly inefficient to compute the distance from a point where an interpolated value is sought to all the data points. Thus, we introduce $N_b << N$, to limit the scope of the search loop. $N_b = N_w + 2$ is the minimum value required in order for the second inner loop to execute at least once when computing R_w. In the limited number of cases we tried, the minimum value proved to produce the most accurate results. In essence, this shows that the algorithm adheres to the locality guidelines intended for the modified Shepard's method. The final step in the computation, after the dynamic radii and consequently the weights have been found, is the solution of the least squares problem. The implementation in AeroTrack relies on Singular Value Decomposition to solve the linear system of equations that result from finding the minimum of the residual R^2 (See Appendix B). This is absolutely different from the approach adopted by Renka who applies Givens rotations. A comparison between the two techniques is beyond the scope of this thesis.

A strong virtue of Renka's method is its translucent extension to the three dimensional case, which is ultimately what we seek in a 3D simulation. Renka demonstrates the usefulness of the method and gives recommendations for the new values of N_q and N_w, based on error analysis and parameter optimization mechanisms. The values for the tri-variate case are $N_w = 32$, and $N_q = 13$ or 17, with marginal improvement for either depending on the function being interpolated. For our purpose, an assessment of both values is done, and the results show that $N_q = 13$ achieves higher accuracy. For completeness, we now present the implementation of the method for three independent variables. Analogous to the two dimensional case, the interpolant is defined as:

$$F(x,y,z)=\sum_{k=1}^{N} Q_k(x,y,z)\frac{W_k(x,y,z)}{\sum_{i=1}^{N} W_i(x,y,z)} \tag{4.2.33}$$

with the same definition, as before, for the weights W_k. The nodal function, however, takes on a trivariate polynomial given by:

$$\begin{aligned} Q_k(x,y,z)={}&c_{k1}(x-x_k)^2+c_{k2}(x-x_k)(y-y_k)+c_{k3}(y-y_k)^2+c_{k4}(x-x_k)(z-z_k)\\ &+c_{k5}(y-y_k)(z-z_k)+c_{k6}(z-z_k)^2+c_{k7}(x-x_k)+c_{k8}(y-y_k)+c_{k9}(z-z_k)+f_k \end{aligned} \tag{4.2.34}$$

and the coefficients c_{k1} to c_{k9} minimize the residual

$$R^2=\sum_{i=1,i\neq k}^{N} \omega_i(x_k,y_k,z_k)[Q_k(x_i,y_i,z_i)-f_i]^2 \tag{4.2.35}$$

whose solution is given in Appendix C.

A cell based method for nearest-neighbor searching is implemented to improve the efficiency of the code, thus speeding its execution time. The method will be described for data on a two-dimensional plane, to facilitate its comprehension. The dimensions of the smallest rectangle containing the data nodes are taken as (XMIN,XMAX) x (YMIN,YMAX). The rectangle is then partitioned into an NR-by-NR uniform grid of cells, and the index of the nodes contained in each cell are stored in a ***hierarchal*** data structure NCELL. Renka uses two elementary data structures to accomplish the same goal, which introduces additional complications to the implementation, as well as the need to devise non-natural ways to perform the search. The methodology we propose is relatively simpler, and requires the same number of operations in the preprocessing phase, $\boldsymbol{O}$ (N). The choice for NR is based on cell density considerations, since the cell density (or average number of nodes per cell) is $C = N/NR^2$ (in three-dimensions, $C = N/NR^3$). The optimum value for C is taken from Bentley et al. [46] to be C = 3. The algorithm for inserting the indices of the nodes into the data structure is outlined by Renka, the only difference here is that in AeroTrack, the cells of NCELL are themselves expandable arrays that hold all the indices to the nodes contained in that cell. The following example with N = 7 and NR = 2 demonstrates the concept. Figure 4.4 shows a 2-by-2 cell data structure whose entries will hold an index value that references a scattered point.

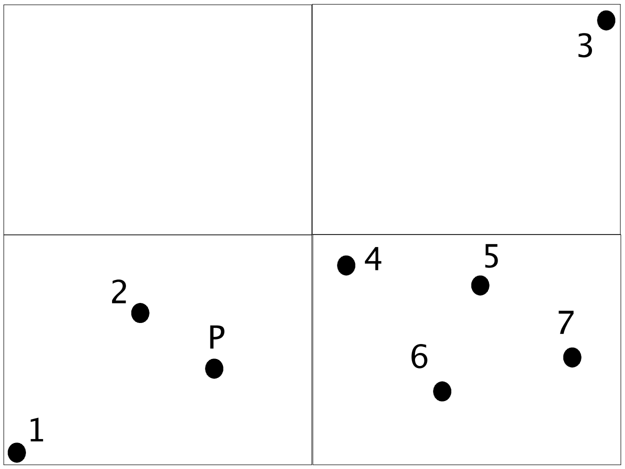

Figure 4.4: 2x2 cell structure with 7 data nodes & a query point P

The internal representation of the data structure NCELL that depicts the arrangement in Figure 4.4, starting from the top left corner, is:

NCELL(1,1) = { } ← empty array

NCELL(2,1) = {1,2}

NCELL(1,2) = {3}

NCELL(2,2) = {4,5,6,7}

Once a query for the nearest neighbors of a point P (shown above) is initiated, the location of P in NCELL is promptly computed with a cost $\boldsymbol{O}(1)$, thus immediately determining the data nodes coexisting with P in that particular cell. The search then proceeds in clockwise layers into the surrounding cells until N_b nodes are collected. In cases where the number of nearby nodes exceeds N_b, which is likely to occur in dense populations, the nodes are sorted by their respective distance from point P, and only the first N_b nodes are considered. To further illustrate the concept of overlaying the data grid with a uniform cell grid, Figure 4.4 shows a representative picture of how this is done for a given mesh used in the simulation of the virtual impactor.

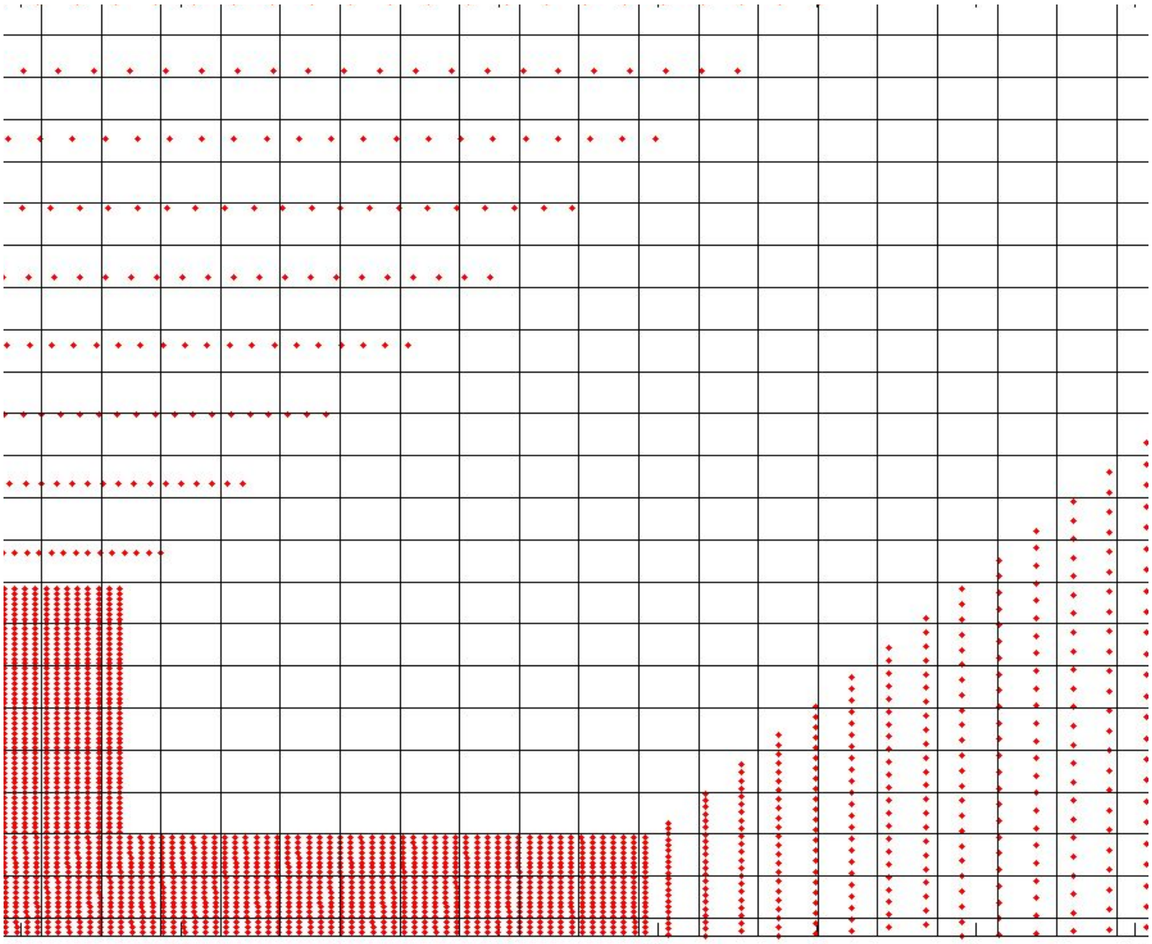

Figure 4.5: Cell grid on top of computational mesh points

The extension of this cell method to three dimensions is simply achieved by constructing a cubical data structure, NCELL(i,j,k), and performing the search in both in-plane and out-of-plane layers. For efficiency reasons, the interpolation process in AeroTrack is designed to perform the interpolation of the three components of velocity, and the value of pressure in one nearest-neighbor search operation. This is accomplished by storing the values of velocity and pressure in one matrix with four columns of data for each of the interpolated variables.

4.2.4b.3 Adaptive Drag Coefficient

The need to experiment with different formulations of the drag coefficient, and more importantly, the need to have the drag coefficient adapt to changes in the flow regime along the particle path, is another reason AeroTrack was developed. In addition to the classical formulations (incompressible, Stokes-Cunningham), we investigate the following two cases, which are not readily accessible in FLUENT™:

- The compressible formulation with a range of lower compressibility limits.
- The Stokes-Cunningham drag coefficient with a variable ***pressure dependent*** slip correction factor.

4.2.4b.4 AeroTrack Task Flow

The inner workings of the FORTRAN code and its execution of commands are summarized in the flow chart below:

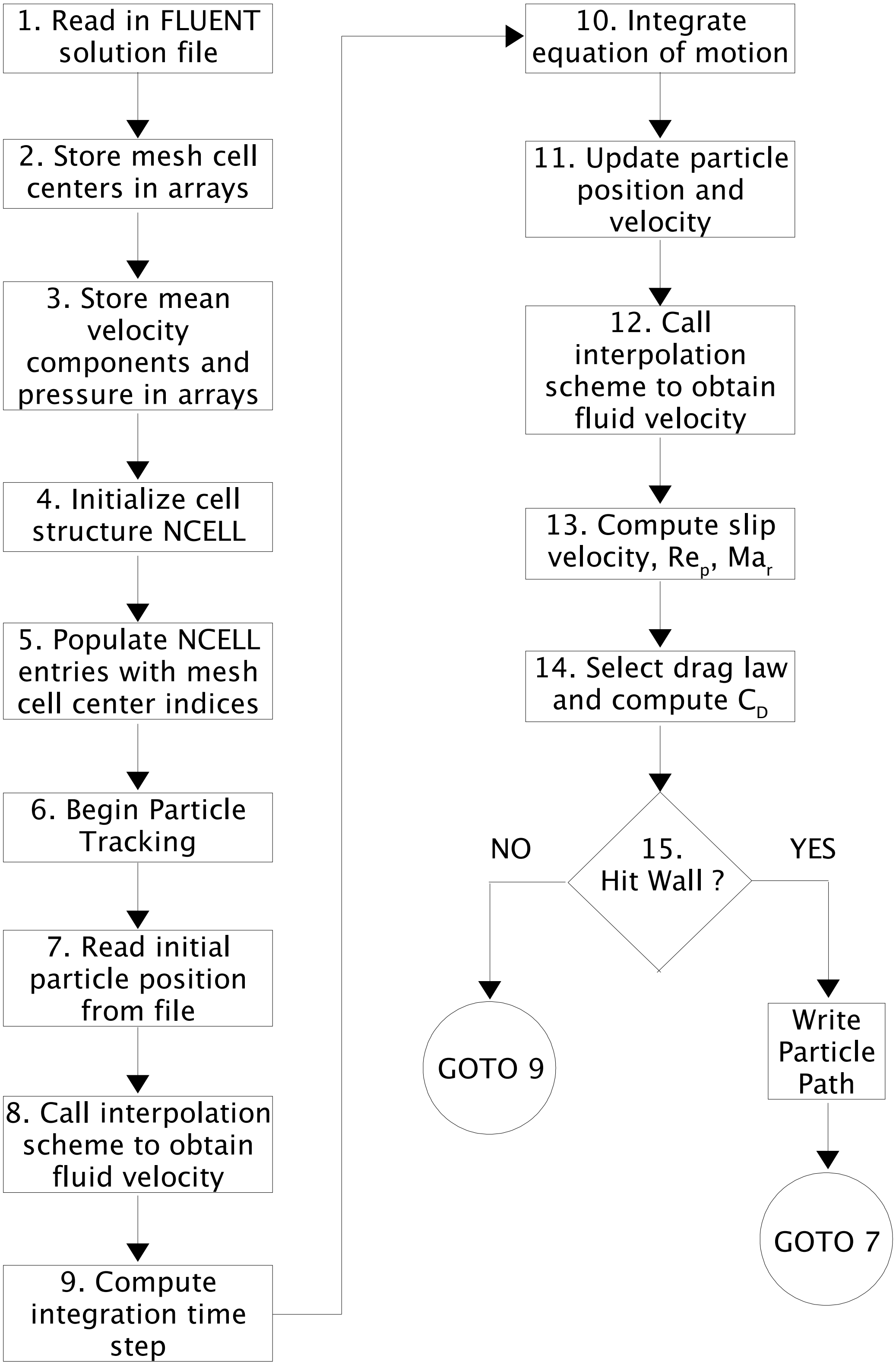
1. Read in FLUENT solution file
2. Store mesh cell centers in arrays
3. Store mean velocity components and pressure in arrays
4. Initialize cell structure NCELL
5. Populate NCELL entries with mesh cell center indices
6. Begin Particle Tracking
7. Read initial particle position from file
8. Call interpolation scheme to obtain fluid velocity
9. Compute integration time step
10. Integrate equation of motion
11. Update particle position and velocity
12. Call interpolation scheme to obtain fluid velocity
13. Compute slip velocity, Re_p, Ma_r
14. Select drag law and compute C_D
NO
15. Hit Wall ?
YES
GOTO 9
Write Particle Path
GOTO 7

Figure 4.6: Flowchart of AeroTrack execution

Chapter 5

Discussion of Numerical Approaches

This chapter covers the numerical tests and methods engaged to arrive at the simulations of the virtual impactor. The different simulations whose solution will be discussed are summarized in Table 5.1:

No.	Type	Grid	Discretization	Turbulence Model	Q_m/Q_T
1	incompressible	coarse	2nd order	κ-ϵ	19 %
2	incompressible	coarse	1st order	RSM	22 %
3	compressible	coarse	2nd order	κ-ϵ	16 %
4	compressible	fine	1st order	κ-ϵ	20 %
5	compressible	fine	2nd order	κ-ϵ	22 %

Table 5.1: Listing of virtual impactor simulations

Simulations number 1 and 2 will be compared to assess the predictions of the turbulence models, to study the effect of turbulent particle dispersion, and to assess the usefulness of stochastic tracking within the framework of the two turbulence models. Simulations number 1 and 3 will be compared to show the effect of compressibility on the fluid flow field. The mesh refinement effect will be discussed in reference to simulations 3 and 5, and finally the influence of the discretization scheme will be demonstrated in the context of the last two simulations. Section 5.1 deals with the convergence criteria for the carrier phase solution. Section 5.2 presents the numerical steps undertaken to insure the accuracy of the discrete phase results both in FLUENT™ and AeroTrack.

5.1 Convergence of the Fluid Flow Solution

The steady state solution of the fluid phase mean and turbulence quantities is obtained by an iterative process that starts from an initial guess of the solution, usually taken from the boundary conditions, and iterates on the governing equations until a solution that meets a prescribed convergence criterion is met. FLUENT™ has its own mechanism for judging convergence, but it is designed to be general to insure its functionality in a wide set of problems. For the virtual impactor simulations, we establish a specific rule to monitor convergence in addition to following the guidelines of FLUENT™. The magnitude of the mean velocity vector in the cross-sectional mid-plane of the device are stored every 500 iterations, and the absolute value of the velocity difference, $|\overline{U}_i(n)-\overline{U}_i(n-500)|$, at each computational cell is computed using the current and the previously stored velocity values. The maximum, as well as the standard deviation of the absolute error, for each velocity component are then plotted over the entire span of iterations. Furthermore, to assess the quality of the converged solution, contours of the absolute error norm are plotted for the virtual impaction region, which is the critical point in the flow that influences the separation characteristics of the device. The convergence behavior plots are presented below:

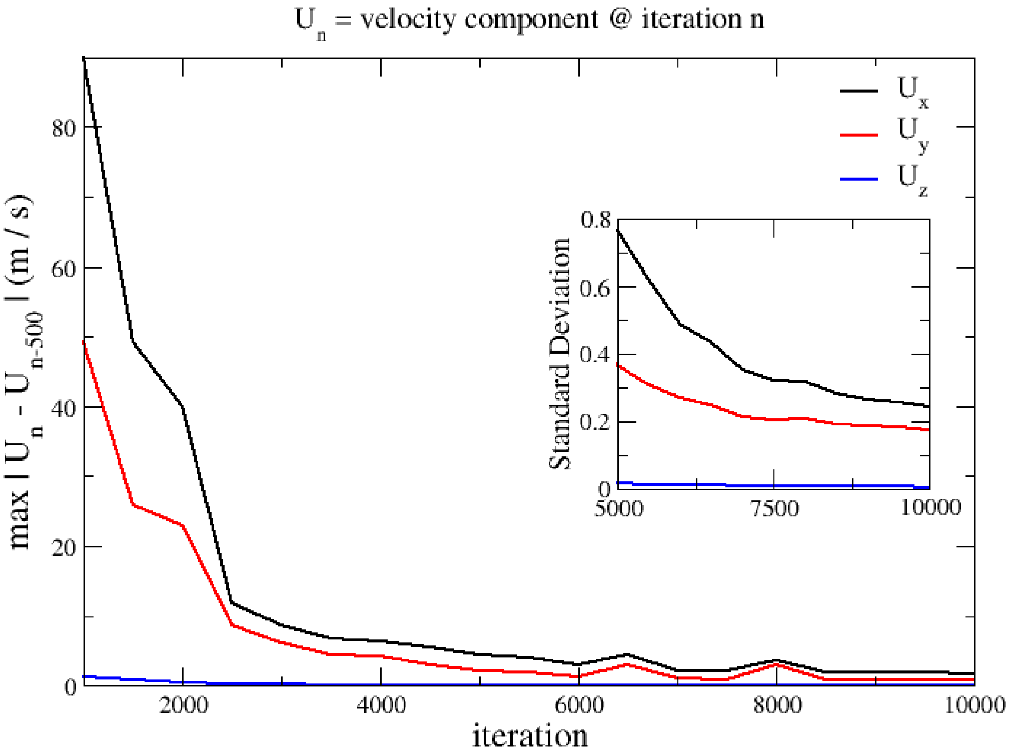

Figure 5.1: Convergence of simulation 1

The standard deviation is calculated using the average of the velocity differences from all the cells in the mid-plane. The norm of the absolute error is defined as:

$$|\bar{U}(\mathrm{last})-\bar{U}(\mathrm{last}-500)|=\sqrt{\sum_{i=1}^{3}[\bar{U}_i(\mathrm{last})-\bar{U}_i(\mathrm{last}-500)]^2} \qquad (5.1)$$

The error norm plot in Figure 5.3, utilizes the final two iteration data sets, for the mesh cells that constitute the virtual impaction region, which is the segment of the device between the accelerating and collection nozzles, as indicated in Figure 5.2. As can be seen, the mean velocity difference is zero in most of the region, except in a small area as the fluid is exiting the nozzle. Judging by the magnitude of the highest contour (1.5 m/s), this is negligible considering the high magnitude of velocity ($\approx$ 203 m/s), which makes the relative error of order 10^{-2}.

Figure 5.4 shows that for the RSM simulation the maximum error ceases to change significantly beyond 8000 iterations. In addition, the error contours are low in most of the impaction region. Apparently, the RSM solution does not have trouble converging close to the nozzles, as we saw for simulation 1, but continues to iterate to match the velocities at the side exits, as observed in Figure 5.5. A more thorough comparison between the two simulations is presented in Chapter 6.

Simulation 3 shows a similar convergence behavior to that of simulation 1, as shown in Figure 5.6, which is somewhat expected since they both use the same turbulence model on the same grid. The contour plot of Figure 5.7 also resembles that of simulation 1, in Figure 5.3, but with higher magnitudes for the velocity difference due to the increased velocities of the fluid at the nozzle expansion brought about by compressibility.

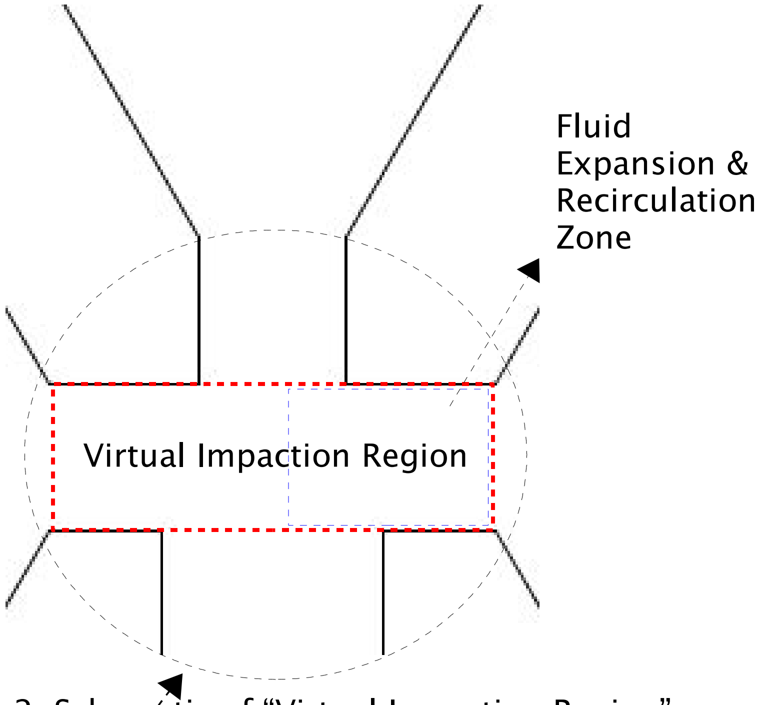

Figure 5.2: Schematic of “Virtual Impaction Region”

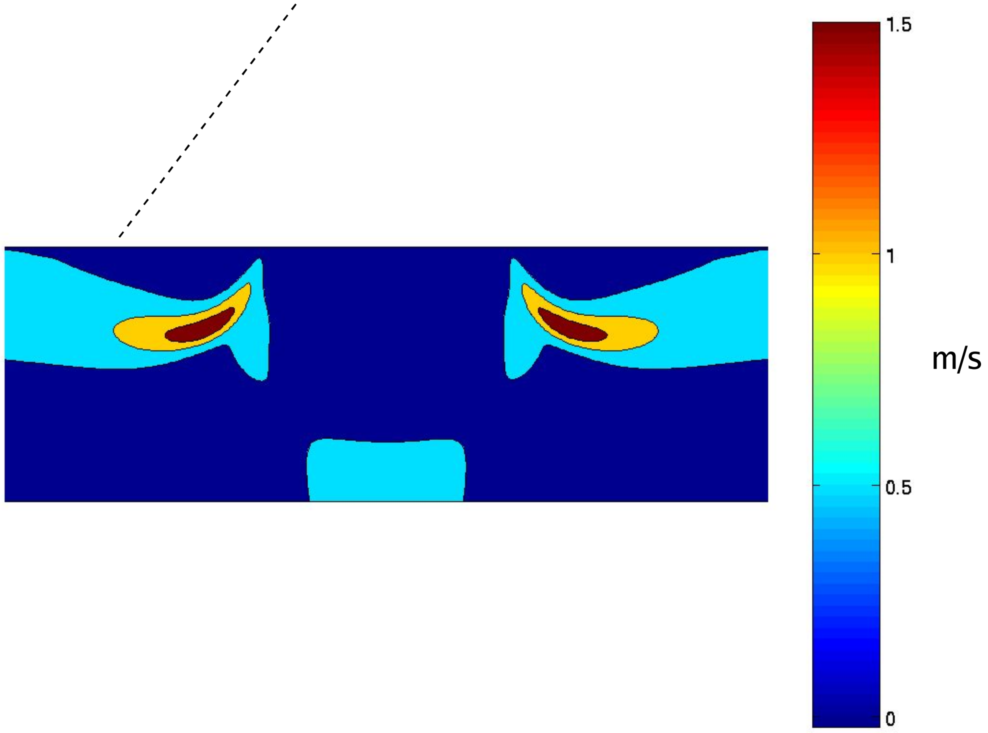

Figure 5.3: Contours of the norm of velocity difference between the final iteration sets (simulation 1)

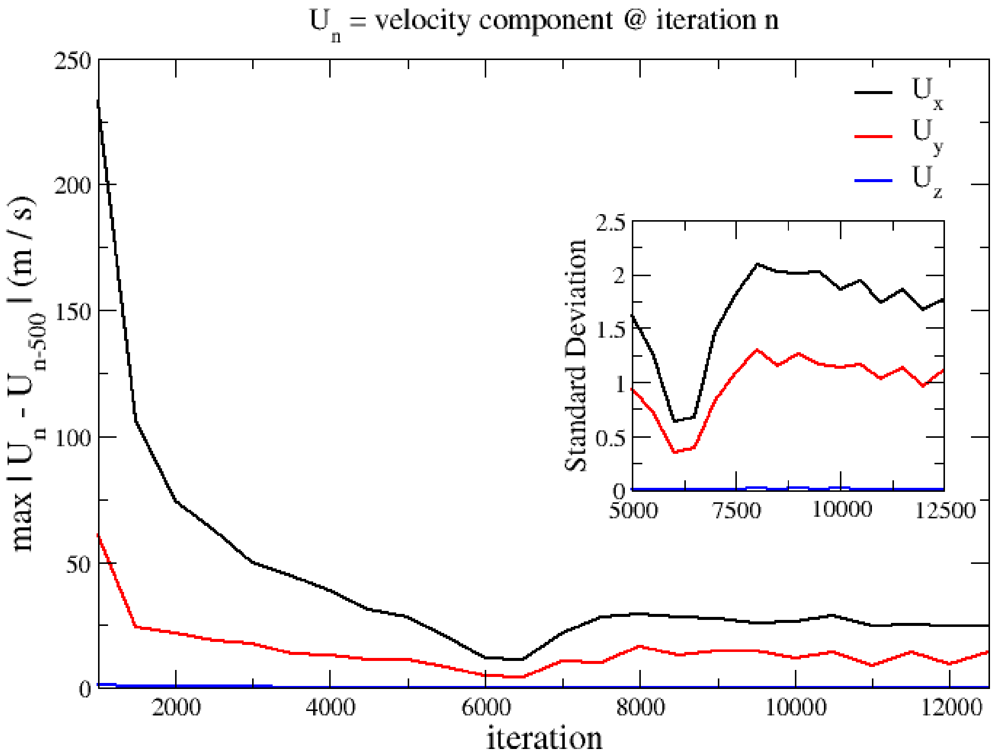

Figure 5.4: Convergence of simulation 2

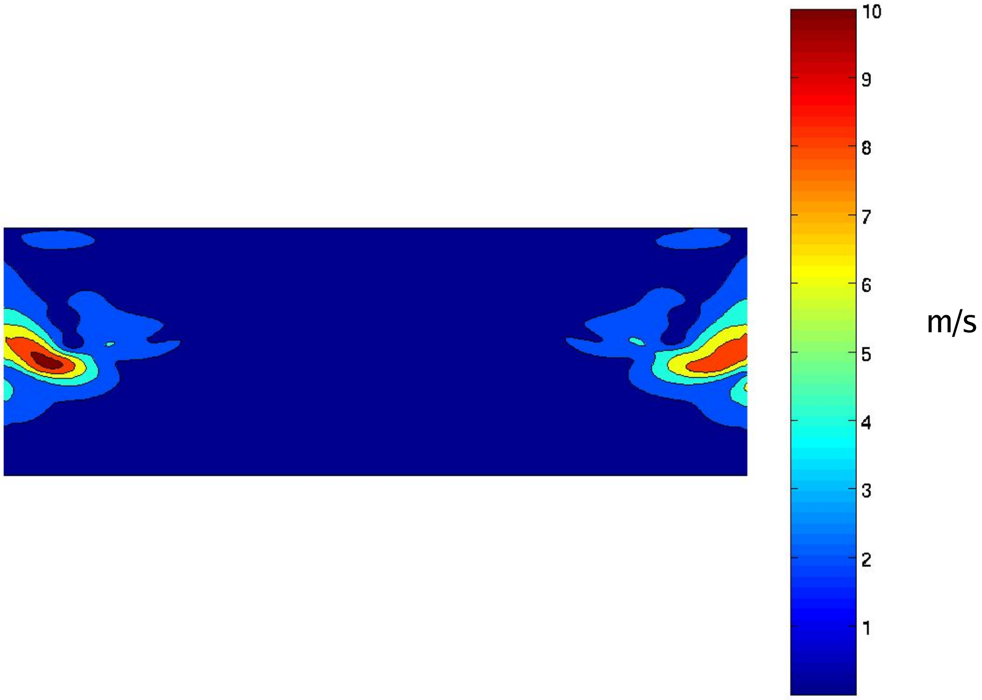

Figure 5.5: Contours of the norm of velocity difference between the final iteration sets (simulation 2)

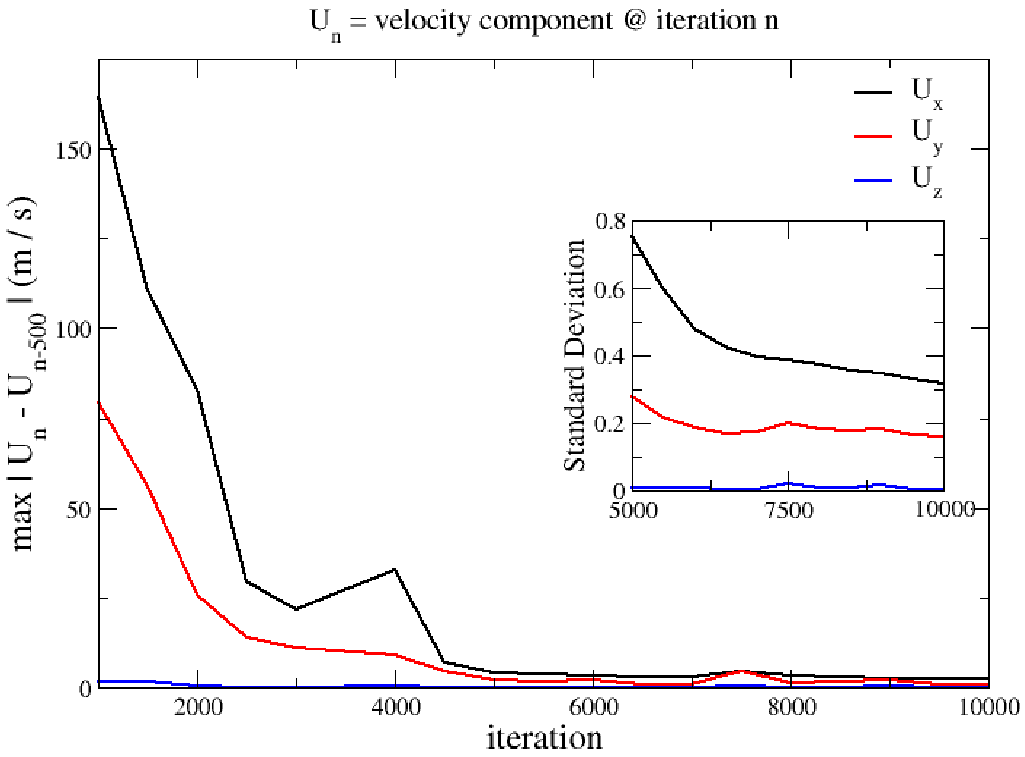

Figure 5.6: Convergence of simulation 3

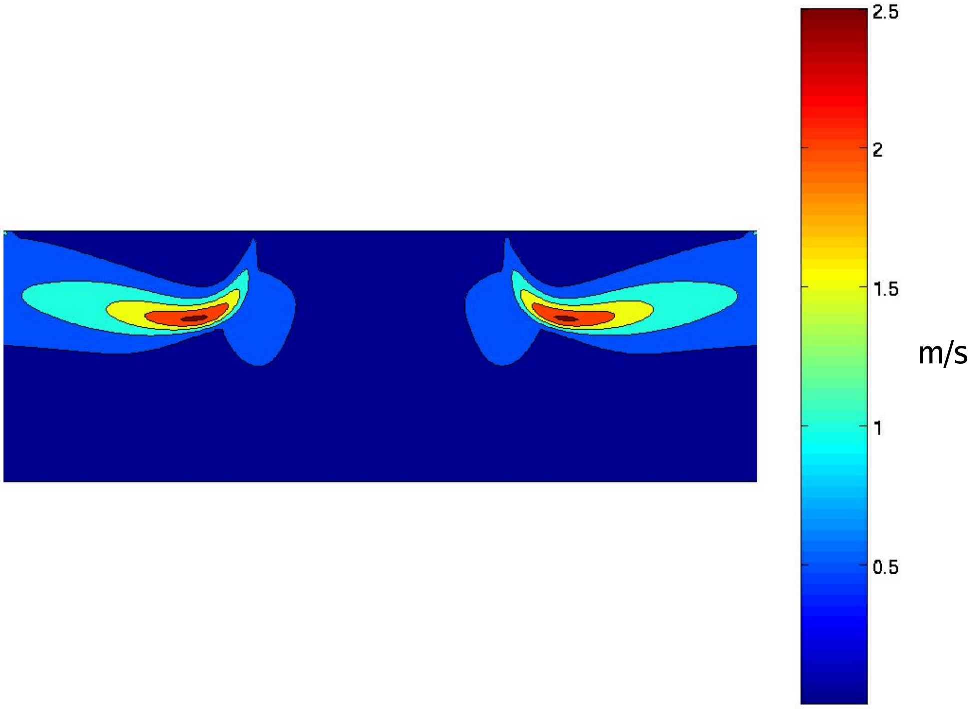

Figure 5.7: Contours of the norm of velocity difference between the final iteration sets (simulation 3)

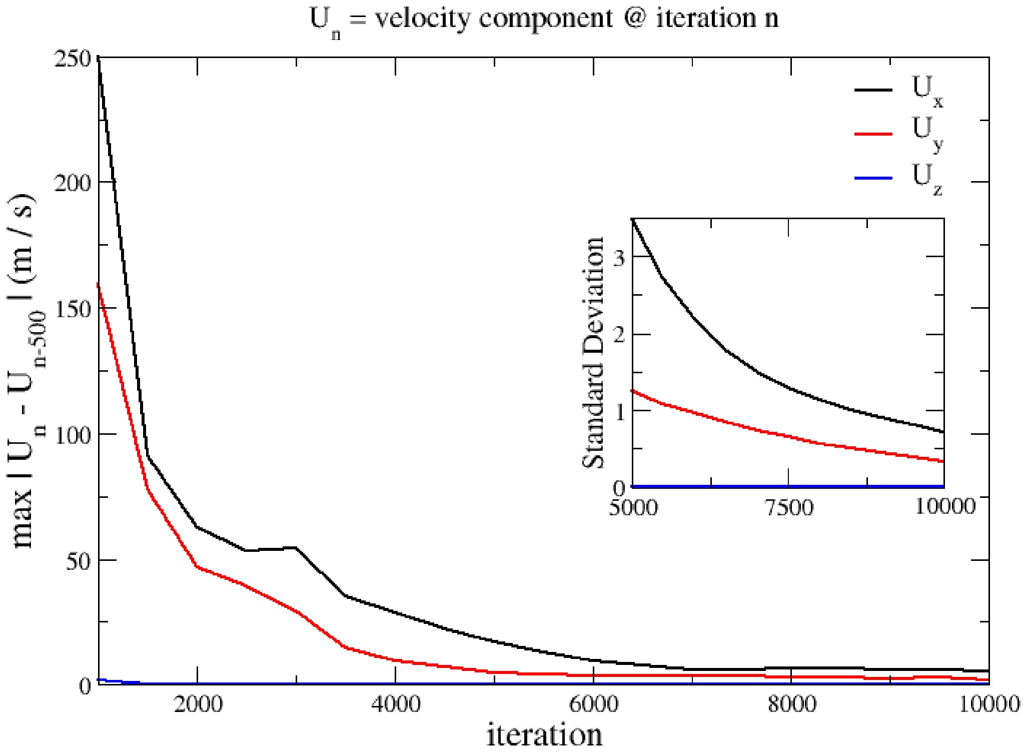

Figure 5.8: Convergence of simulation 4

The convergence plot of simulation 4 in Figure 5.8 shows a smooth decay in the absolute error norm as the number of iterations increases. The standard deviation values, however, are not as low as their counterparts in simulation 3, which can be attributed to more significant changes in the mean velocity between the cells in the virtual impaction region and the ones outside. Interestingly enough, the k-epsilon solution on the finer gird, depicts convergence difficulties that are more pronounced at the exits than at the nozzles as shown in Figure 5.9. This can be attributed to the ability of the finer gird to better resolve the high gradients near the nozzles, but appears quite lacking in the high shear region close to the wall, which may be caused by a number of reasons, wall functions being one of them. Despite this observation, the highest contour value is still within reasonable limits.

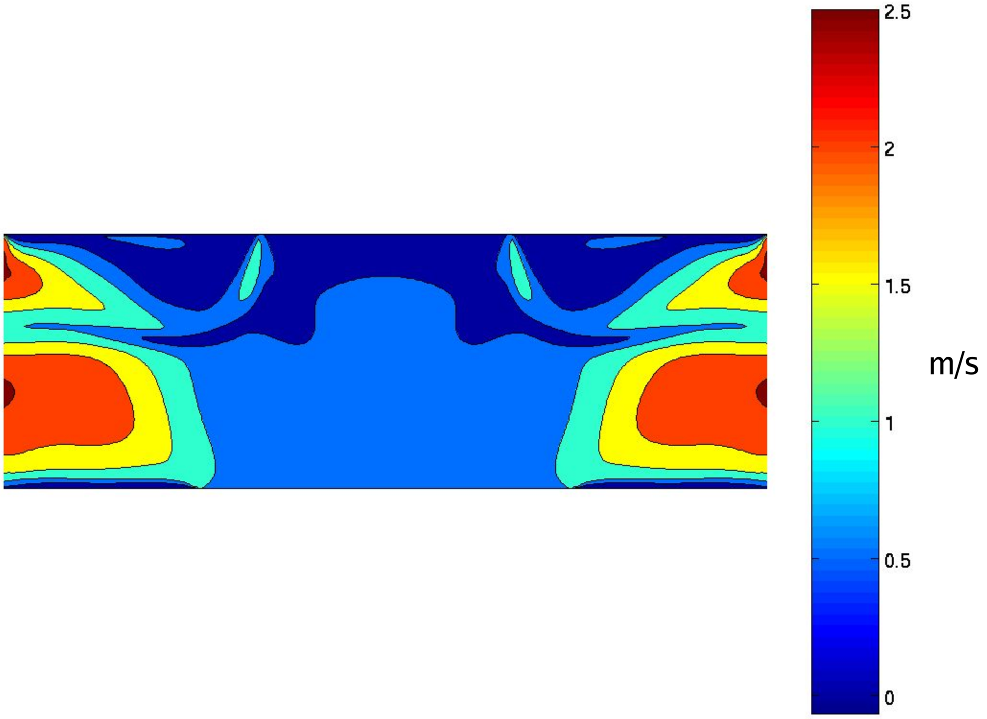

Figure 5.9: Contours of the norm of velocity difference between the final iteration sets (simulation 4)

Finally, simulation 5, as shown in Figure 5.10, achieves standard deviation values that are lower than those of simulation 4, and exhibits a similar contour plot in the virtual impaction region, shown in Figure 5.11. The scale of the error, however, is reduced by half, clearly due to the more accurate second order discretization scheme.

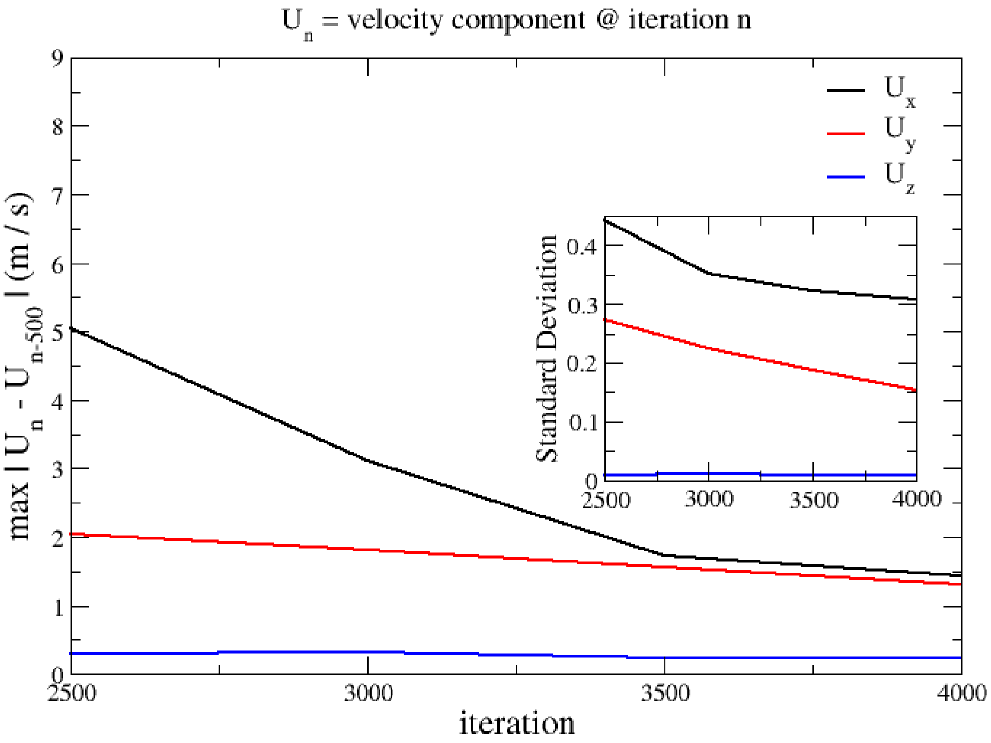

Figure 5.10: Convergence of simulation 5

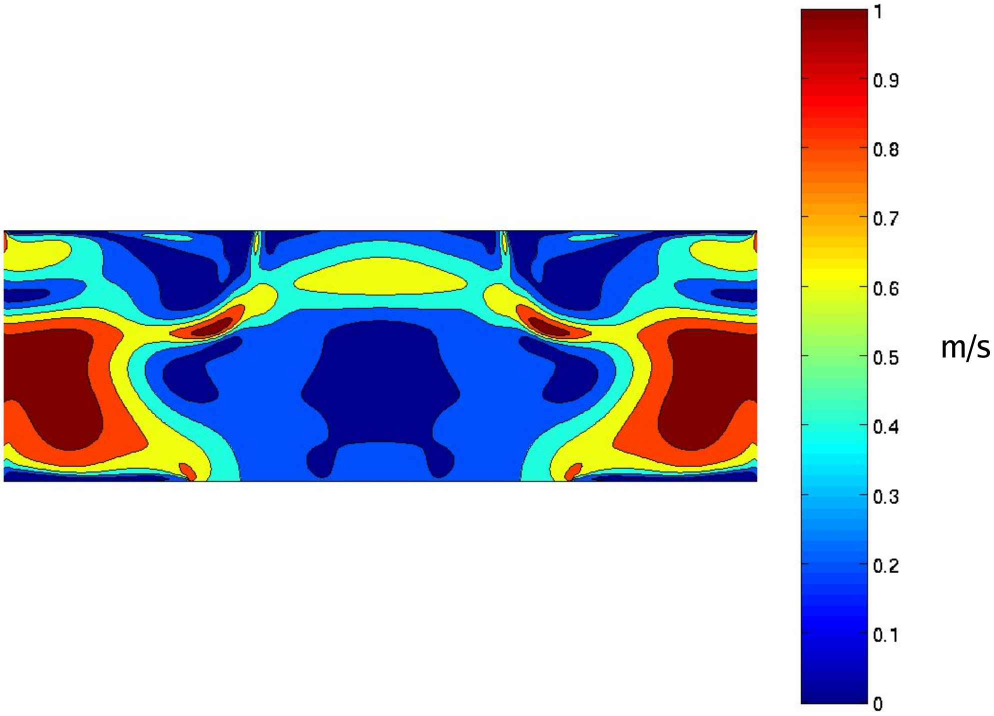

Figure 5.11: Contours of the norm of velocity difference between the final iteration sets (simulation 5)

In summary, the decrease in the maximum error with iteration number is a common attribute to all the simulations which is certainly the first frontier in achieving convergence. Second, the convergence plots show that the maximum error remains relatively constant for at least 2000 iterations. Third, the contour plots resemble reasonable limits for the absolute error, because the relative error is bounded (<< 1).

The standard deviation plots in Figures 5.1, 5.4, 5.6, 5.8, and 5.10, are calculated using the mean velocity difference values listed in Table 5.2. The table below summarizes the mean velocity difference of all the mid-plane cells from the final two iteration sets, so that standard deviation values can be ascertained.

No.	<U_x(last)-U_x(last-500)>	<U_y(last)-U_y(last-500)>	<U_z(last)-U_z(last-500)>
1	0.148	0.175	0.003
2	0.673	0.522	0.004
3	0.160	0.161	0.002
4	0.440	0.430	0.001
5	0.208	0.196	0.002

Table 5.2: Mean velocity difference between “converged” and “pre-converged” solutions

5.2 Accuracy of the Particle Tracking Algorithms

In this section, three aspects that affect the quality of the particle tracking calculations are discussed, and a description of the methods employed to guarantee or improve the accuracy is conveyed. The numerical subjects examined are: time stepping, the number of particles to use, and interpolation.

5.2.1 Time Stepping

The numerical solution of the particle equation of motion requires a value for the time integration step, Δt. The choice of Δt dictates the upper bound of the numerical error, ***O*** (Δt) or ***O*** (Δt^2), depending on the numerical scheme whether it is 1st or 2nd order respectively. To avoid setting either a "too small" or a "too large" fixed time step, the methodology for computing Δt relies on a fixed length scale ΔL. The ***dynamic*** time step is then computed as:

$$\Delta t = \frac{\Delta L}{\left|\vec{U}_p + \vec{U}\right|}. \tag{5.2}$$

The choice for fixing ΔL is usually simple knowing how much do we want the particle to travel in one time step. For the discrete phase results presented in Section 6.2, a value of ΔL = 1.0e^{-06} (m) is used for all the particle tracking calculations done regardless of the grid resolution. The justification for this value is based on the minimum computational cell dimension. As shown in Table 3.1 the minimum cell volume is of the order 1.0e^{-14}, thus its cubic root is of the order 1.0e^{-05}. A more strict constraint is the size of the 2D discretization at the nozzle: for the fine mesh, Δ = W/100 = 0.00033/100 = 3.3e^{-06} (m). Therefore, we require the particle to cross a two-dimensional cell in at least 3 time steps.

To establish the independence of the particle tracking results from the length step value, a number of calculations are performed in AeroTrack with ΔL values of 1.0e^{-05}, 1.0e^{-06}, and 1.0e^{-07} (m). Figures 5.12-5.14 show the trajectory of a 0.25 μm particle released from the nozzle throat. From the first figure, it appears that there is very little discrepancy between the paths followed using any of the ΔL values, however, the difference is amplified in the the next two figures that reveal the inadequacy of the largest ΔL value, and the duplicate outcome of the intermediate and smallest values.

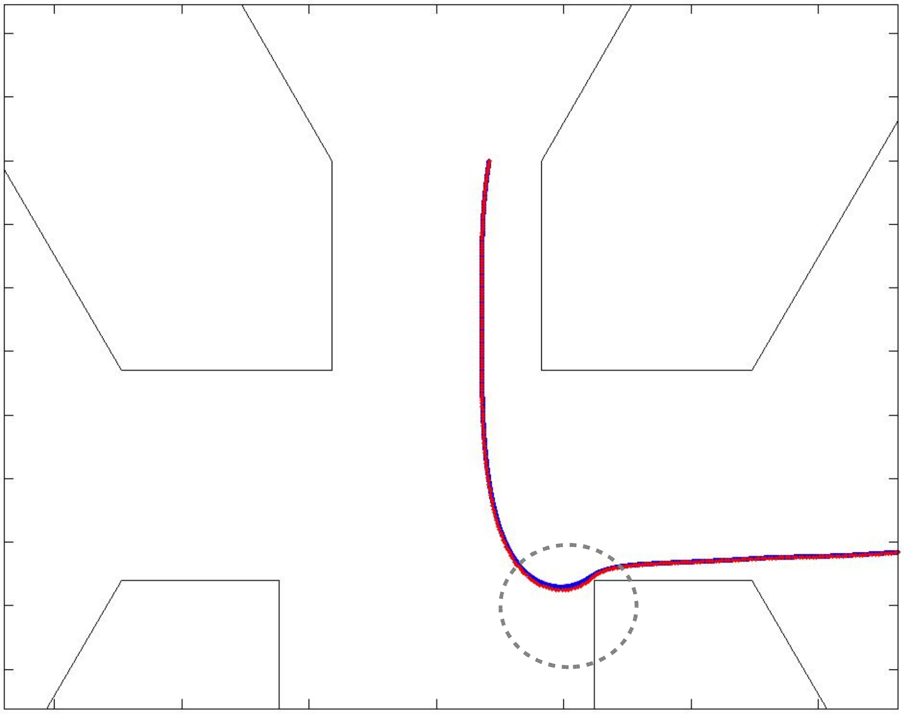

Figure 5.12: Particle paths using ΔL=1.0e^{-05} (red), 1.0e^{-06} (blue), and 1.0e^{-07} m (magenta) {simulation 4}

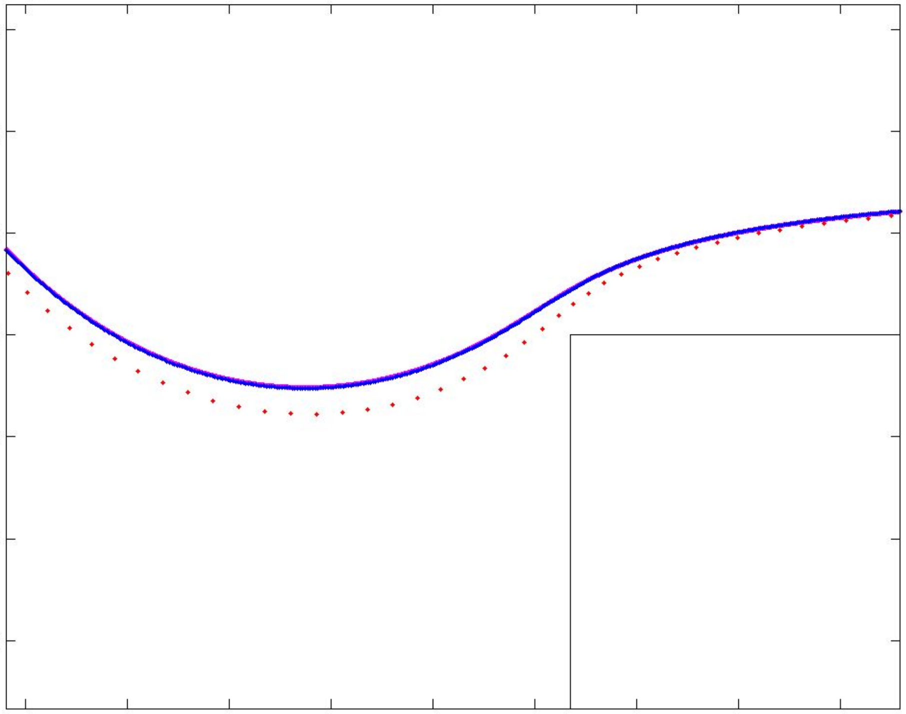

Figure 5.13: Particle paths using ΔL=1.0e^{-05} (red), 1.0e^{-06} (blue), and 1.0e^{-07} m (magenta) – zoom level 1 – {simulation 4}

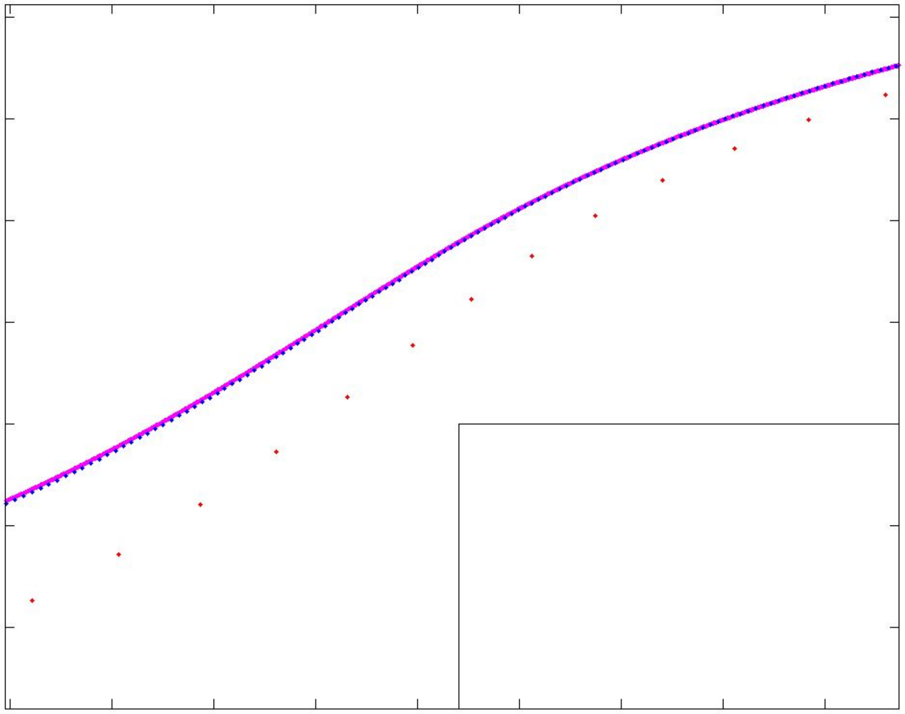

Figure 5.14: Particle paths using ΔL=1.0e^{-05} (red), 1.0e^{-06} (blue), and 1.0e^{-07} m (magenta) – zoom level 2 – {simulation 4}

The same conclusion is reached by looking at Figure 5.15, which shows the path of a heavier particle (0.3 μm), released from the same location and approaching the collection nozzle wall. Clearly, the largest ΔL value misses the impact location, whereas the other two values hit the same target with sufficiently small steps in the near wall region.

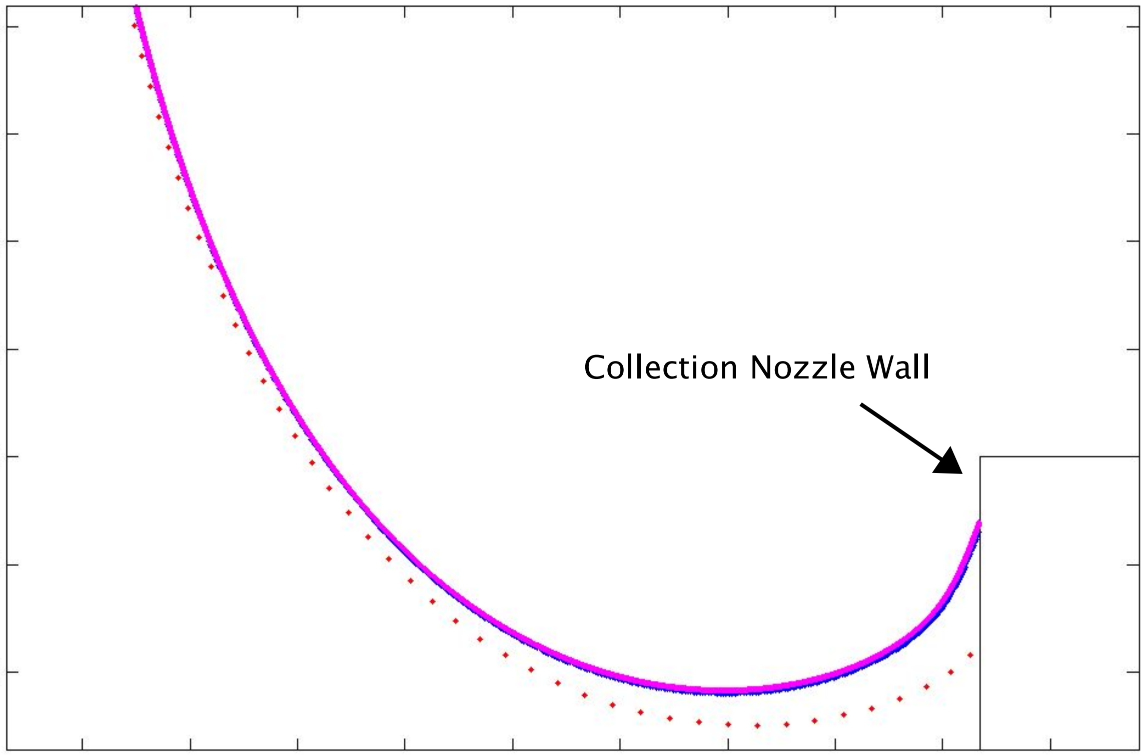

Figure 5.15: Particle approaching a wall using $\Delta L=1.0e^{-05}$ (red), $1.0e^{-06}$ (blue), and $1.0e^{-07}$ m (magenta) – zoom level 1 – {simulation 4}

5.2.2 Number of Particles

While the process of calculating the particle paths using the mean velocity of the carrier fluid (mean tracking) is purely deterministic, the Random Walk Model, described in Section 4.2.3, relies on the generation of random numbers, both for calculating the eddy lifetime, and for yielding fluctuating velocity components. Therefore, to establish sufficient accuracy in the stochastic approach, a significantly large sample of particles must be used. For mean tracking purposes, we perform a calculation in FLUENT™ for each of the eleven particle size samples using N_p = 5043 particles per sample. The calculation is then repeated using 10013 particles, and the efficiency and wall losses curves are compared. The collection efficiency is defined as:

Eff = (# of particles in minor flow)/(# of particles in minor + major flow)

and the wall losses are computed as:

Loss = (# of particles trapped)/(total # of particles)

Furthermore, to insure the one-way coupling mode, the volume fraction of the 10013 particles is calculated for the sample with the largest particle diameter (0.4 μm) to be:

Φ_p = volume of particles/(Area of cone x Δz) ≈ 8.0e^{-09}

where Δz is the depth of the computational cells from which the particles are released, and the area is that of the entrance cone section of the device. Figure 5.16 shows the efficiency and losses curves obtained from tracking the mentioned samples in simulation 1 using the Stokes-Cunningham drag law. Note that FLUENT^{TM} requires a constant value for the slip correction factor to be entered as input, so atmospheric pressure was used to specify C_c. The plot is only shown to support the argument that a sample with 5043 particles is sufficient for mean tracking, and not necessarily to convey the performance of the device, which will be elaborated upon in Chapter 6.

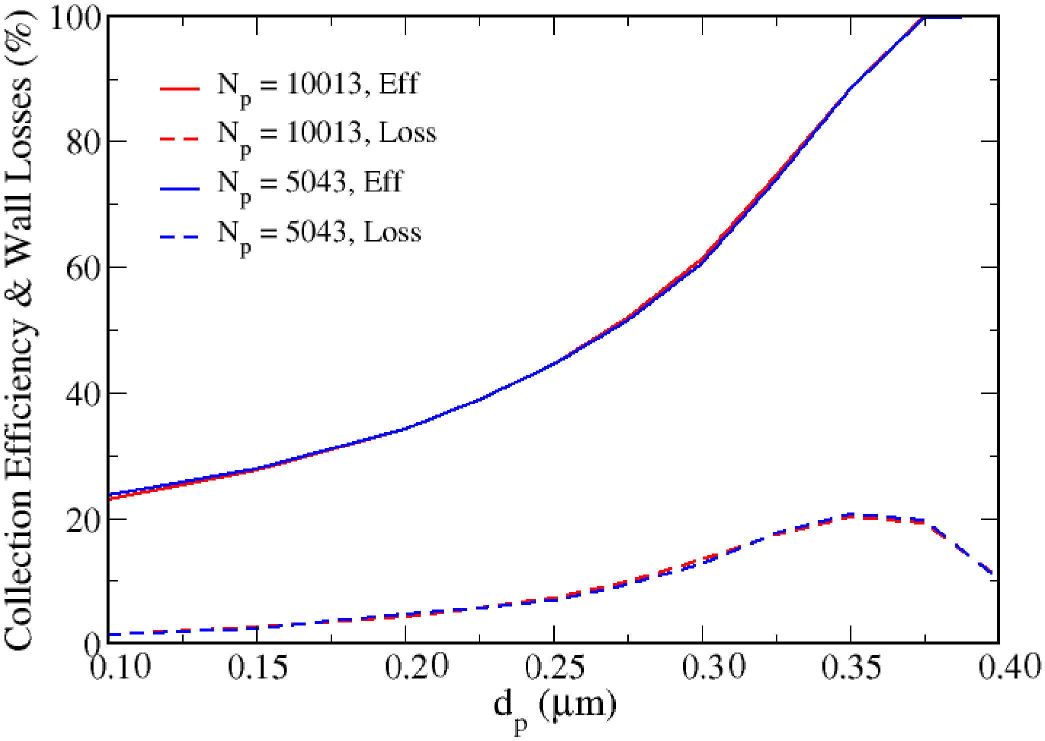

Figure 5.16: Efficiency and wall losses curve for two sample sizes, simulation 1 FLUENT™ mean tracking with Stokes-Cunningham

The question of how many particles for the stochastic tracking is addressed from a statistical standpoint. To establish confidence limits, or standard deviation figures, we conduct trajectory calculations on 0.1 μm particles for three sample sizes, N_p = 5043, 10,013, and 20,063. Each calculation is then repeated ten times, and the mean and standard deviation of the efficiency and wall losses are recorded. The variability of the results from each run are shown in Figures 5.17-5.18.

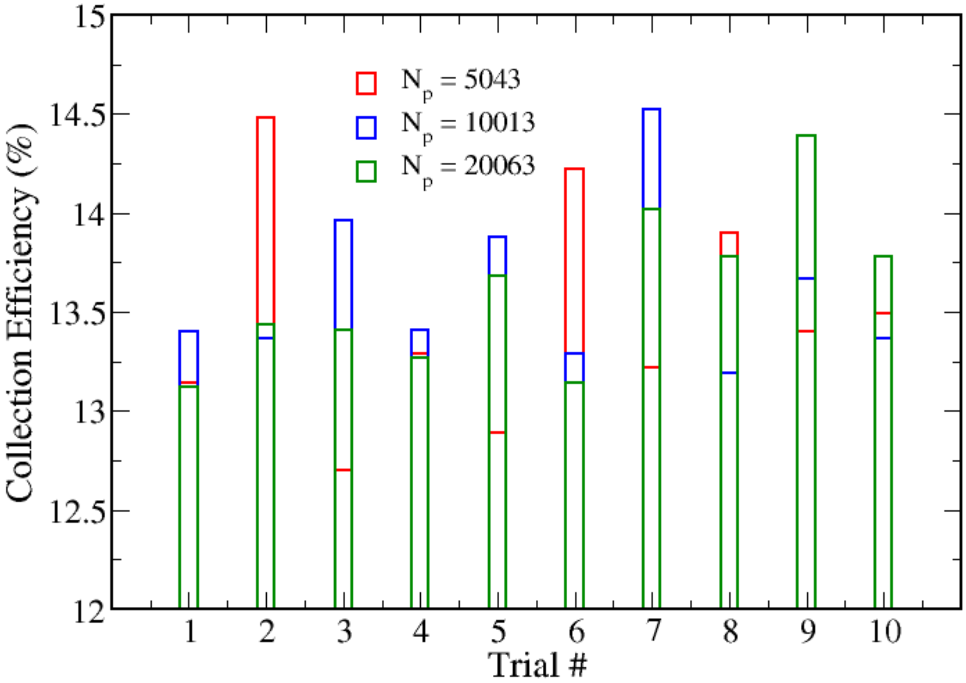

Figure 5.17: Collection efficiency of 0.1 μm particles for three different sample sizes, simulation 1 FLUENT™ stochastic tracking with Stokes-Cunningham drag law

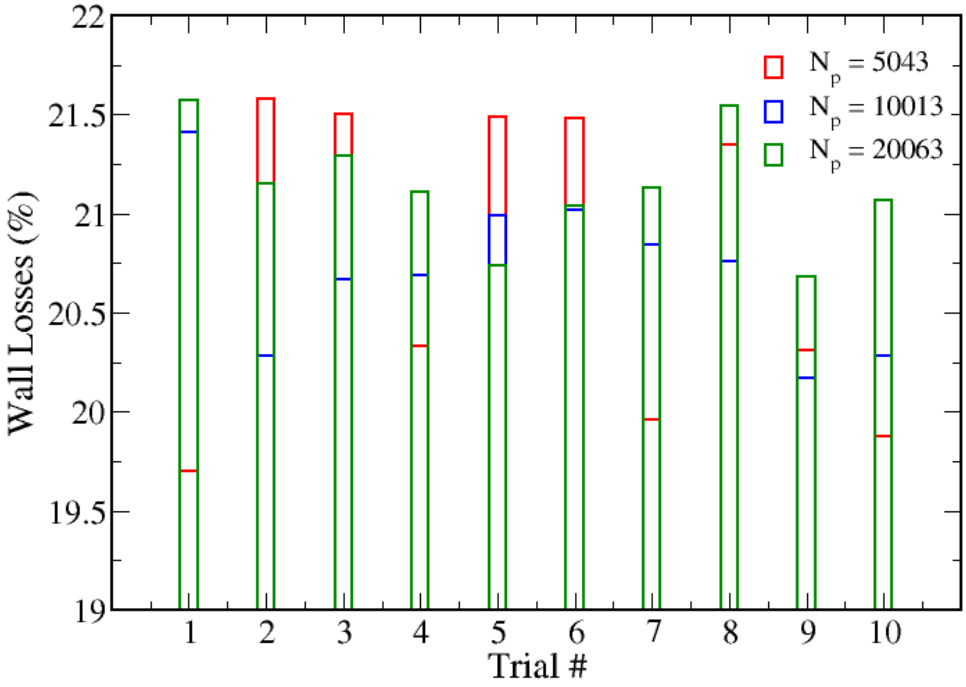

Figure 5.18: Wall losses for 0.1 μm particles for three different sample sizes, simulation 1 FLUENT™ stochastic tracking with Stokes-Cunningham drag law

The mean and standard deviation of the data from the ten stochastic runs are summarized in Table 5.3. The mean of the collection efficiency and wall losses is not altered significantly by the change in the number of particles, however the standard deviation undoubtedly benefits from the increased value. The difference, however, between the intermediate and the large sample is minimal, therefore, we choose to conduct the stochastic tracking using a sample size of N_p = 20,063.

	N_p = 5043	N_p = 10,013	N_p = 20,063
Mean Efficiency	13.47 %	13.61 %	13.60 %
STD Efficiency	0.57	0.41	0.41
Mean Loss	20.76 %	20.71 %	21.13 %
STD Loss	0.79	0.39	0.29

Table 5.3: Mean and standard deviation of collection efficiency and wall losses for 0.1 μm particles using three different sample sizes, simulation 1 FLUENT™ stochastic tracking with Stokes-Cunningham drag law

The mean values obtained from these calculations do not necessarily correspond to the actual or physical values. Clearly, the mean collection efficiency is flawed based on the simple physical argument that particles with inertia ought to collect in the minor flow more than the massless fluid elements. In other words, the particulate efficiency must be greater than or roughly equal to Q_m/Q_T. The cause of this fallacy, of course, is due to the isotropic nature of the k-epsilon turbulence model, which will be commented on further in Chapter 6. Nevertheless, for pure statistical/numerical reasons, the technique can be relied upon to reveal the mandated sample size. The choice of N_p = 20,063 was further validated in simulation 2, by generating a curve similar to Figure 5.16, with a smaller sample size of 10013 particles, and the results are indifferent.

5.2.3 Interpolation Tests

The purpose of this section is to demonstrate the validity and the improved accuracy of the multi-variate scattered point interpolation scheme, presented in Chapter 4, over the "widely used" linear interpolation method. The implemented code is subjected to a series of interpolation tests outlined below:

a. Test functions of two variables studied by Renka and Brown [47]

b. Test functions of three variables used by Renka [44]

c. Sinusoidal three dimensional velocity field examined by Kontomaris [35]

d. Actual numerical data from the virtual impactor simulations

5.2.3a Bi-Variate Functions

Following the approach devised by Renka, a 33-by-33 rectangular grid of evaluation nodes is constructed on the unit square. A set of 100 interpolation nodes with randomly assigned locations in the square is then generated. Figure 5.19 depicts the described arrangement. Shepard's method, and its two modified versions of Franke-Nielson and Renka, were programmed into MATLAB ® so as to compare their performance to MATLAB's built-in interpolation routines of linear and cubic interpolation. The following test functions listed in [47] are used:

$$\begin{aligned} F_1(x,y) &= 0.75\exp(-((9x-2)^2+(9y-2)^2)/4) \\ &+\ 0.75\exp(-(9x+1)^2/49-(9y+1)/10) \\ &+\ 0.50\exp(-((9x-7)^2+(9y-3)^2)/4) \\ &-\ 0.20\exp(-(9x-4)^2-(9y-7)^2) \end{aligned} \tag{5.3}$$

$$F_2(x,y)=\frac{\tanh(9y-9x)+1}{9} \tag{5.4}$$

$$F_3(x,y)=\frac{1.25+\cos(5.4y)}{6+6(3x-1)^2} \tag{5.5}$$

$$\begin{aligned} F_8(x,y) &= \exp\left(\frac{-(5-10x)^2}{2}\right)+0.75\exp\left(\frac{-(5-10y)^2}{2}\right) \\ &+\ 0.75\exp\left(\frac{-(5-10x)^2}{2}\right)\exp\left(\frac{-(5-10y)^2}{2}\right) \end{aligned} \tag{5.6}$$

These particular functions were chosen from a list of ten functions used in Renka's accuracy tests paper, simply because they somewhat resemble a velocity field (See Figures 5.20-5.23).

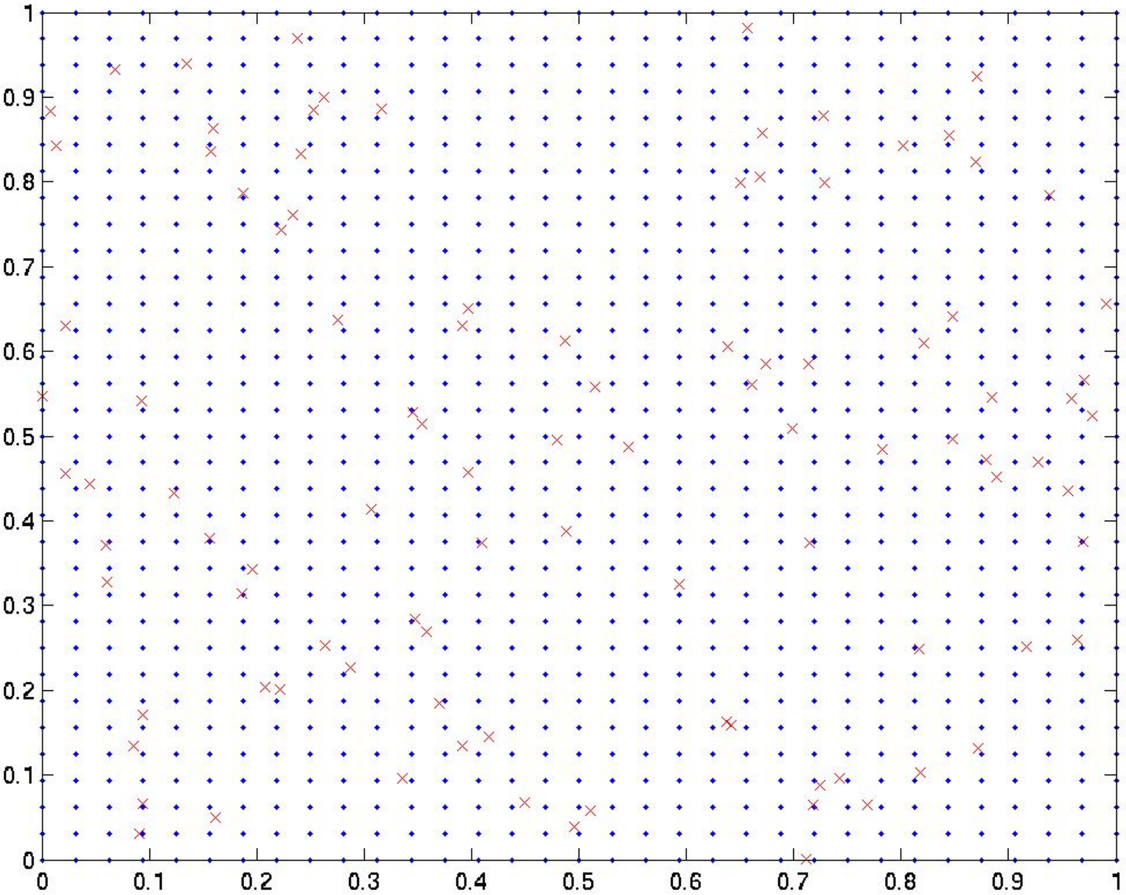

Figure 5.19: 33x33 uniform grid (blue) with a set of 100 random nodes (red)

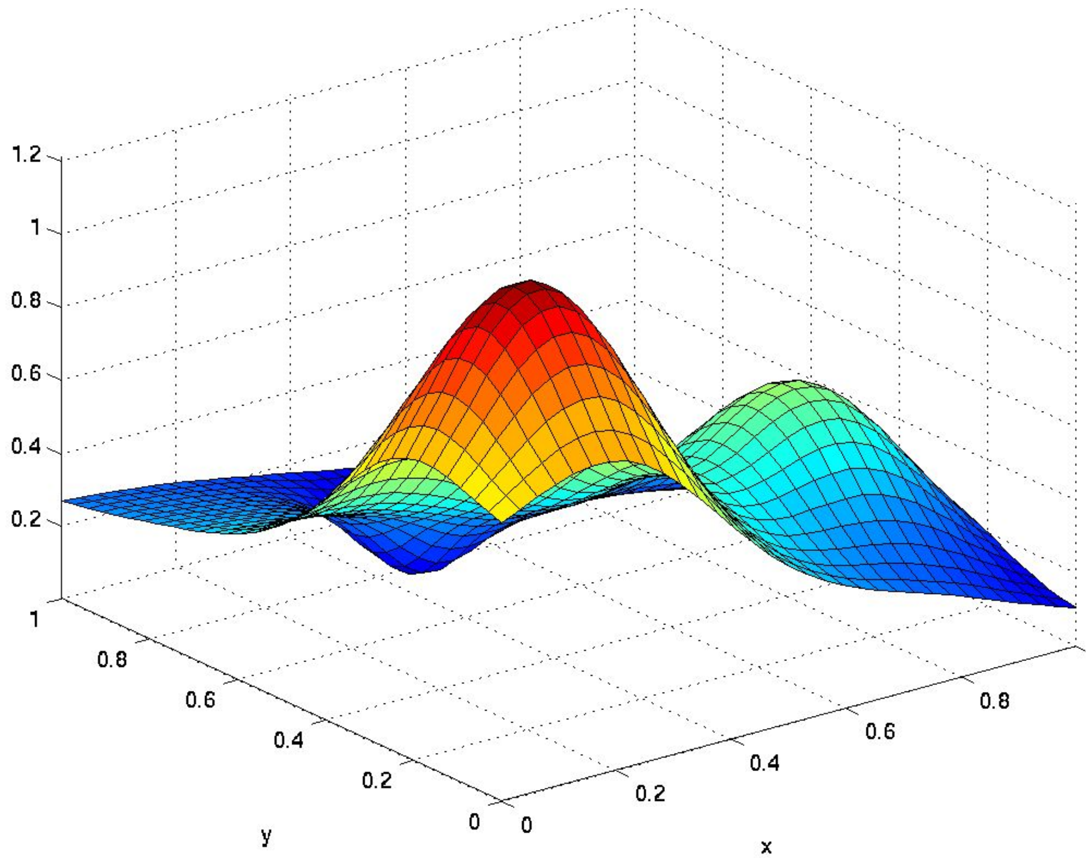

Figure 5.20: Bivariate test function F_1; Equation (5.3)

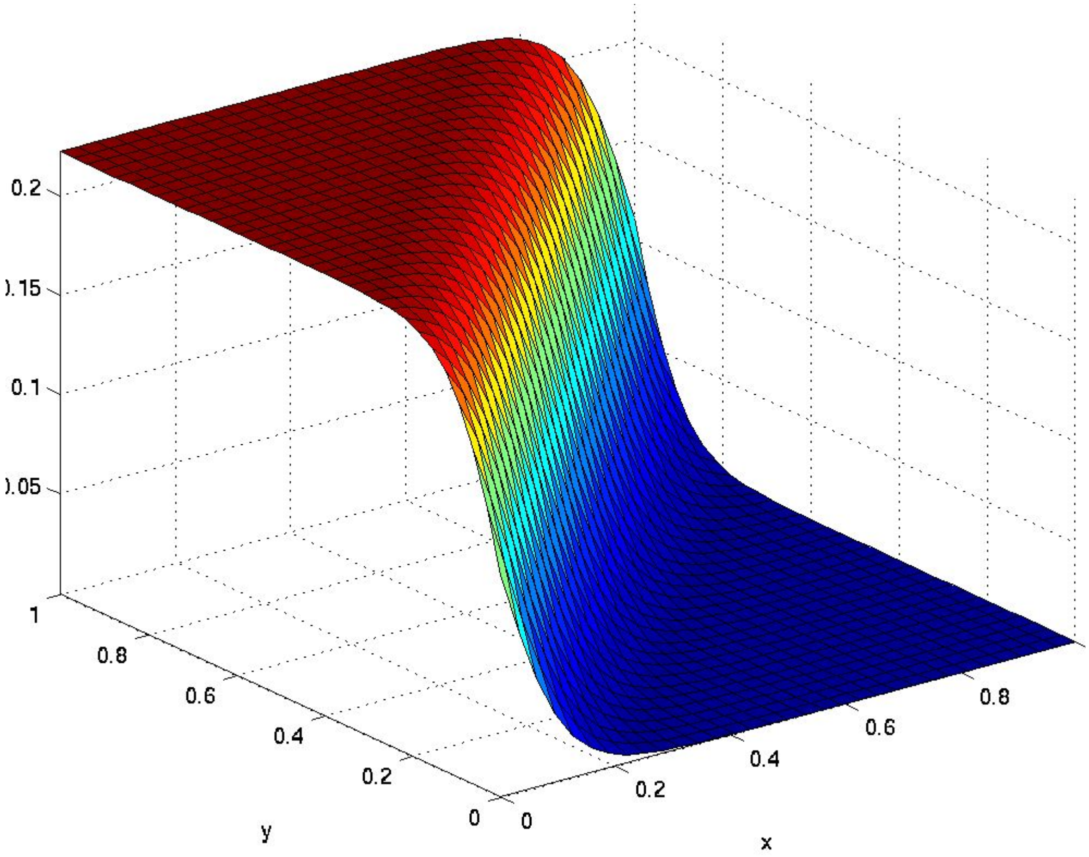

Figure 5.21: Bivariate test function F_2; Equation (5.4)

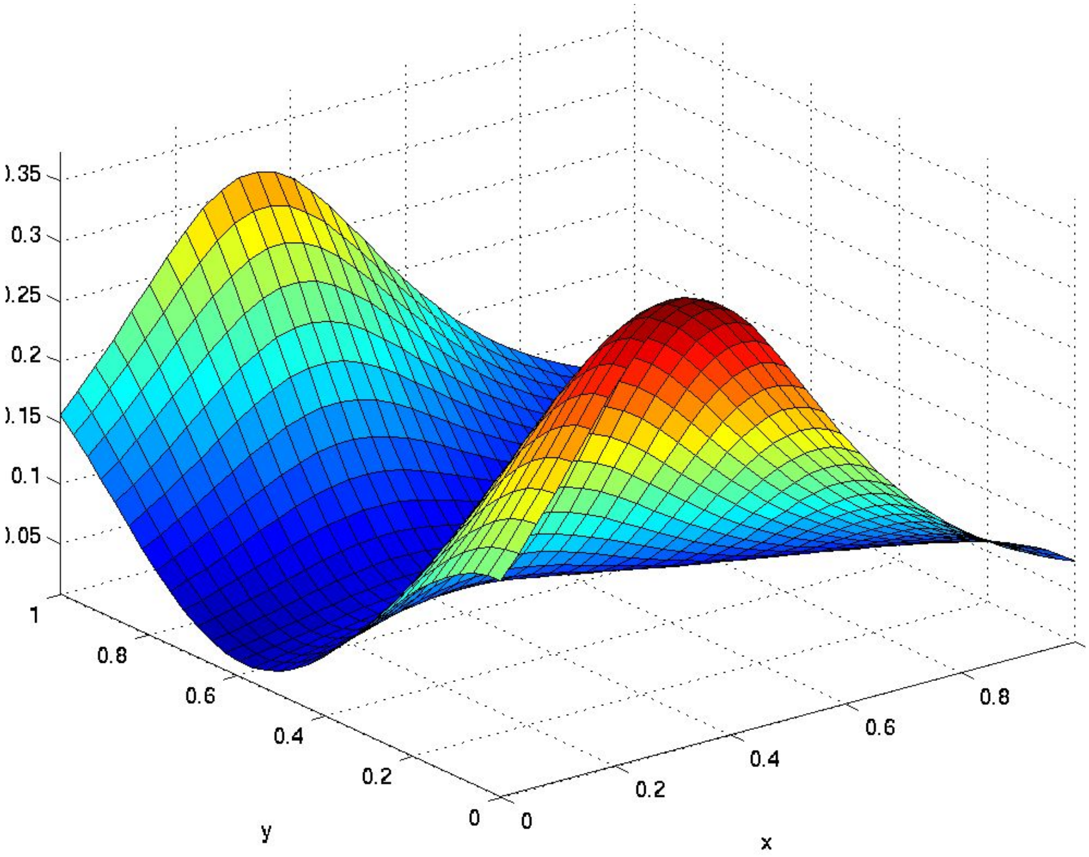

Figure 5.22: Bivariate test function F_3; Equation (5.5)

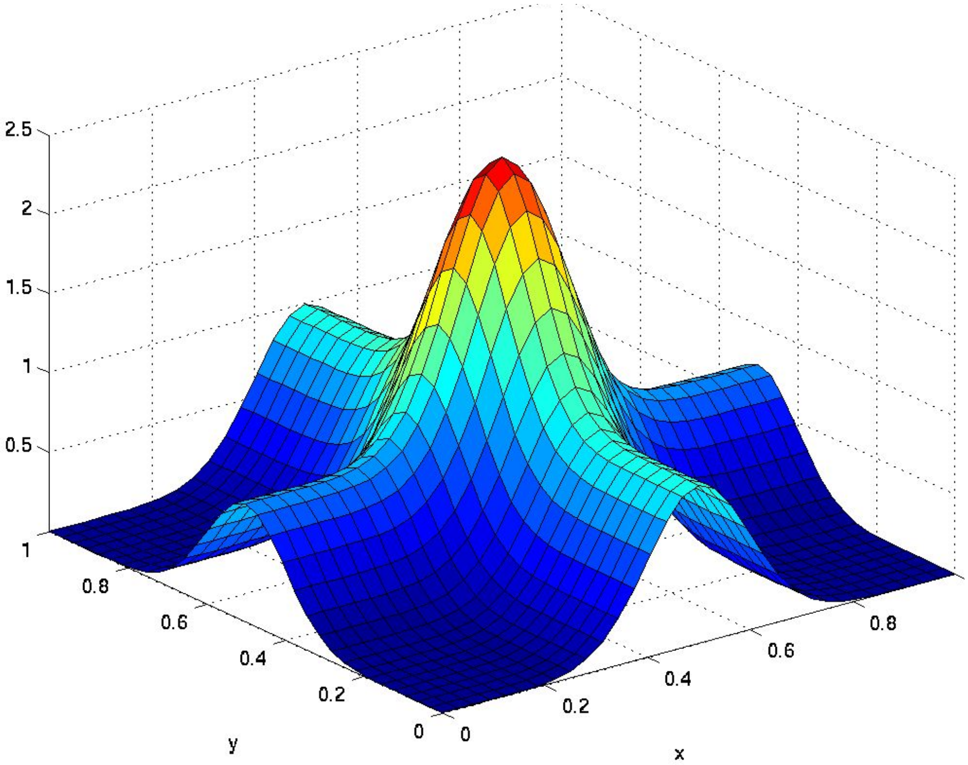

Figure 5.23: Bivariate test function F_8; Equation (5.6)

The results of the interpolation tests on the above bivariate functions with the 100 random nodes set are summarized in Table 5.4. The default parameters of N_q and N_w, given in Chapter 4, were utilized for the modified Shepard's methods. No attempt has been made to optimize the parameters at this stage, since we are only after "proof of concept" results. Table 5.4 shows the maximum, mean, and the root-mean-square of the interpolation errors for each of the functions. It is evident that linear interpolation is subordinate to either of Franke-Nielson or Renka. The results also indicate the comparable accuracy between those methods and cubic interpolation which is only valid on Cartesian grids. It is also interesting to see that the Renka approach is not always superior to that of Franke and Nielson. This, certainly, does not mean that the Renka claims are unfounded, since algorithmic differences in our implementation can be the reason.

	MAX	MEAN	RMS
Function 1: Eqn 5.3			
Linear	0.00692	0.00068	0.01250
Cubic	0.00016	0.00004	0.00051
Shepard	0.01603	0.00291	0.04252
Franke-Nielson	0.00032	0.00007	0.00095
Renka	0.00029	0.00006	0.00092
Function 2: Eqn 5.4			
Linear	0.00157	0.00031	0.00551
Cubic	0.00014	0.00002	0.00034
Shepard	0.00478	0.00065	0.01336
Franke-Nielson	0.00031	0.00004	0.00070
Renka	0.00043	0.00005	0.00097
Function 3: Eqn 5.5			
Linear	0.00095	0.00019	0.00264
Cubic	0.00002	0.00000	0.00007
Shepard	0.00496	0.00095	0.01365
Franke-Nielson	0.00005	0.00001	0.00015
Renka	0.00005	0.00001	0.00014
Function 8: Eqn 5.6			
Linear	0.03119	0.00384	0.06402
Cubic	0.00183	0.00027	0.00422
Shepard	0.03557	0.00642	0.09523
Franke-Nielson	0.00279	0.00040	0.00624
Renka	0.00302	0.00044	0.00679

Table 5.4: Interpolation errors for bivariate test functions

Lastly, to establish confidence in our implemented versions of modified Shepard's method, an interpolation test is conducted on Franke's node set used by Renka to evaluate the accuracy of his “QSHEP2D” scheme [47]. Franke's 100 node set, depicted in Figure 5.24, includes some points outside the unit square so as to include the effect of extrapolation. The results of this comparative test in terms of rms error values are shown in Table 5.5, which shows favorable similarities in accuracy between the different approaches. Despite the slight advantage of the Franke-Nielson method (on this particular node set, and for these particular functions) over Renka's method, we only choose to implement Renka's method for the three dimensional case, since it was proven to work in the literature.

	QSHEP2D*	Franke-Nielson	Renka
Function 1: Eqn 5.3	0.001029	0.003907	0.004860
Function 2: Eqn 5.4	0.001599	0.005750	0.005052
Function 3: Eqn 5.5	0.000308	0.000769	0.001012
Function 8: Eqn 5.6	0.006639	0.005984	0.007490

Table 5.5: Error Norms for modified Shepard Algorithms on Franke's node set. *Reported in [47].

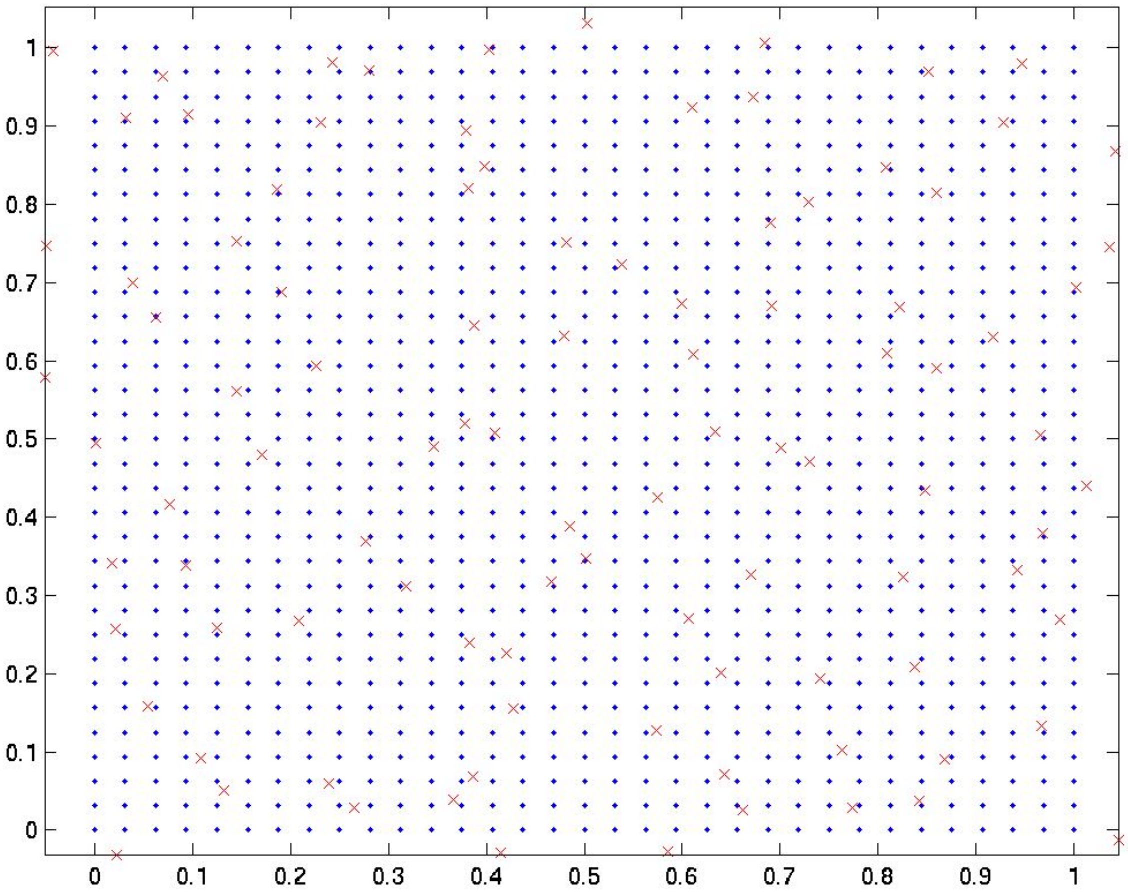

Figure 5.24: 33x33 uniform grid (blue) with Franke's 100 nodes (red)

5.2.3b Tri-Variate Functions

For the trivariate case, a 20-by-20-by-20 uniform grid of evaluation points is used in the unit cube, along with 216 randomly assigned interpolation nodes. The test functions taken from Renka [44] are:

$$\begin{aligned} F_1(x,y,z) = {} & 0.75\exp(-((9x-2)^2+(9y-2)^2+(9z-2)^2)/4) \\ & + 0.75\exp(-(9x+1)^2/49-(9y+1)/10-(9z+1)/10) \\ & + 0.50\exp(-((9x-7)^2+(9y-3)^2+(9z-5)^2)/4) \\ & - 0.20\exp(-(9x-4)^2-(9y-7)^2-(9z-5)^2) \end{aligned} \tag{5.7}$$

$$F_2(x,y,z) = \frac{\tanh(9z-9x-9y)+1}{9} \tag{5.8}$$

$$F_3(x,y,z) = \frac{[1.25+\cos(5.4y)]\cos(6z)}{6+6(3x-1)^2} \tag{5.9}$$

The interpolation errors are shown in Table 5.6 for the linear, cubic, and Renka methods. The parameters N_q =13, and N_w = 32 are used for the latter. As expected, the linear interpolation errors are an order of magnitude more than those achieved by Renka's method. Moreover, it is comforting to see that the mean errors from cubic and Renka interpolation are comparable.

The sensitivity of the interpolation error to the grid resolution is investigated. A number of evaluation grids with resolutions varying from 22x22x22 to 88x88x88 are used, and the interpolation is carried out on the same 216 random nodes. Figure 5.25 is a log-log plot showing the reduction in the rms error as the number of meshpoints is increased.

	MAX	MEAN	RMS
Function 1: Eqn 5.7			
Linear	0.01880	0.00103	0.03232
Cubic	0.00148	0.00014	0.00418
Renka	0.00154	0.00014	0.00444
Function 2: Eqn 5.8			
Linear	0.00567	0.00074	0.02302
Cubic	0.00123	0.00009	0.00319
Renka	0.00210	0.00020	0.00687
Function 3: Eqn 5.9			
Linear	0.00764	0.00086	0.02146
Cubic	0.00029	0.00004	0.00087
Renka	0.00101	0.00007	0.00217

Table 5.6: Interpolation errors for trivariate test functions

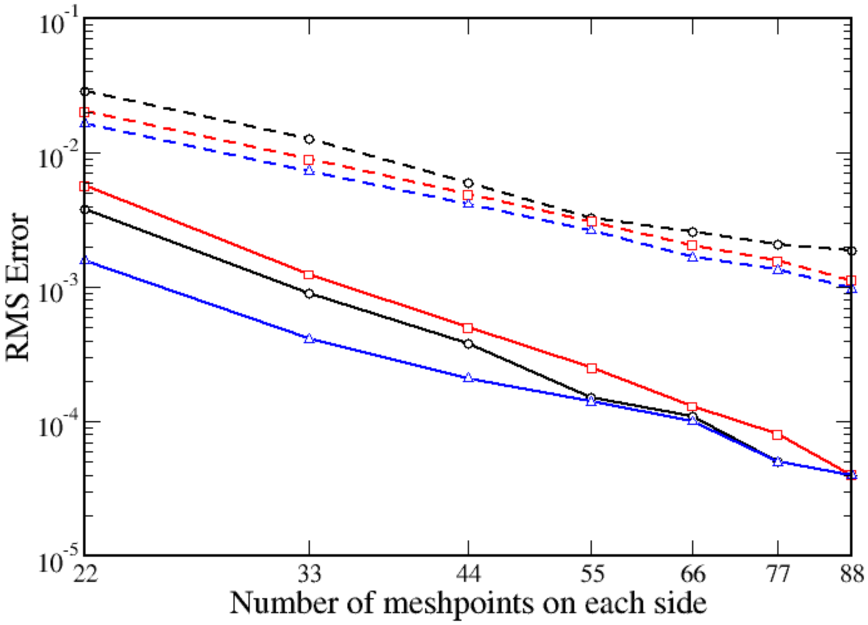

Figure 5.25: Interpolation error versus resolution for the interpolation of trivariate functions F_1 (black), F_2 (red), & F_3 (blue). Renka (solid lines), linear (dashed lines)

5.2.3c Prescribed Velocity Field

The performance of the scattered-point interpolation scheme is further assessed by computing the error incurred by interpolating a random set of 500 nodes in an analytically known velocity field that qualitatively resembles a turbulent flow field. Following Kontomaris [35], the velocity field is:

$$U(x,y,z)=\sin(\boldsymbol{k}x+\boldsymbol{k}y+\boldsymbol{k}z) \tag{5.10}$$

where $\boldsymbol{k}$ is the component of the wavevector $\vec{\mathbf{k}}=(\boldsymbol{k},\boldsymbol{k},\boldsymbol{k})$. The evaluation domain is a cubic box of side length L = 2π, gridded with a uniform mesh spacing h = L/64 in each direction. The scale of motion is obviously dictated by the value of the wavenumber $\boldsymbol{k}$. Other researchers have observed that at low wavenumber components (large scales), the interpolation is carried out with more accuracy than at high wavenumbers. Kontomaris plots the root-mean-square of the interpolation error versus R, where R is the ratio of the scale of motion to the grid spacing. R is related to the wavenumber as such:

$$R=\frac{\lambda_m}{h}=\frac{2\pi}{|\vec{\mathbf{k}}|h} \tag{5.11}$$

The code is put to the test again, and this time we compare its performance to linear, cubic, and cubic spline interpolation, which are all contained in MATLAB. Figure 5.26 demonstrates the reduction in error as the scale of motion is increased regardless of the interpolation scheme. The rate at which this reduction takes place, however, is severely influenced by the interpolation scheme. Clearly, linear interpolation is the least accurate. The plot is instrumental in showing the relative accuracy of the Renka approach in comparison to linear and cubic interpolation. The conclusions that the method achieves third order accuracy, and that the error decreases with increased resolution at a rate proportional to that of cubic interpolation can thus be made.

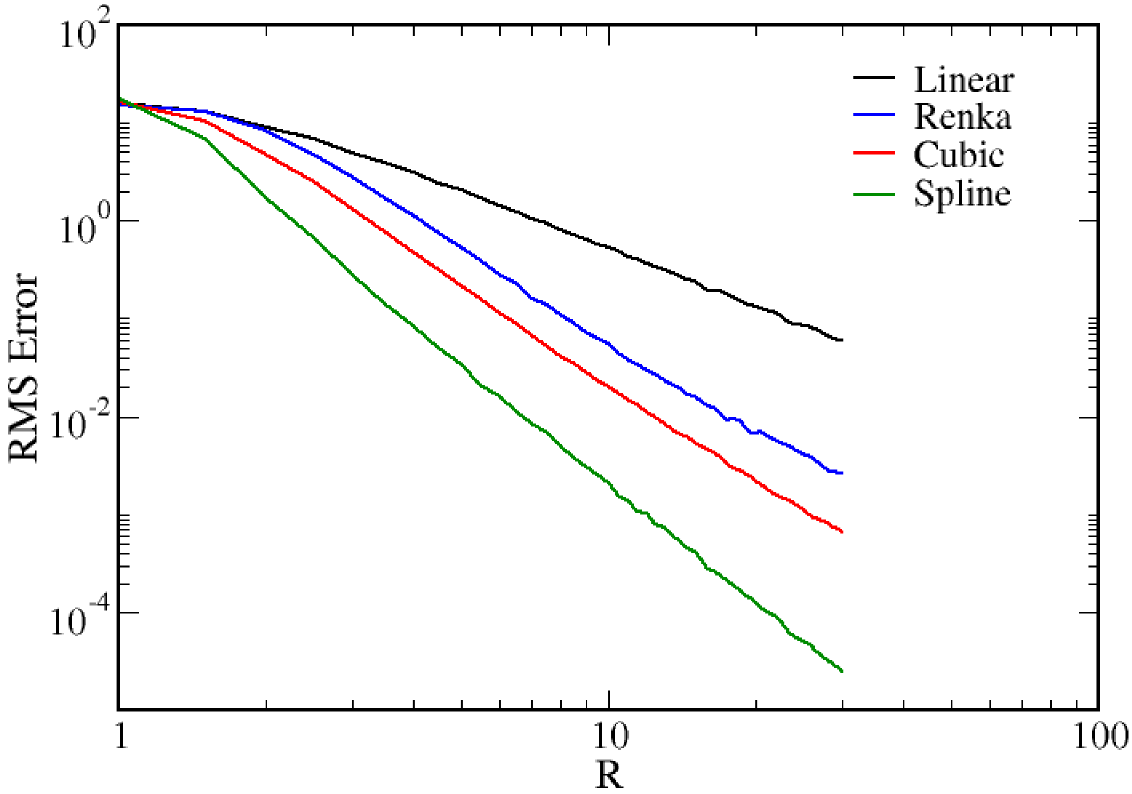

Figure 5.26: Interpolation error versus resolution for the interpolation of sinusoidal velocity fields (Eqn 5.10) using: linear, Renka, cubic, & spline interpolation

Lastly, a brief remark about the computational intensity of the above interpolation methods is deserved. For the interpolation tests performed on the sinusoidal velocity fields, the time taken by each method is monitored over the span of the resolution scale. The average time is then computed, and the performance ratios are calculated. It is found that linear interpolation requires the shortest time to compute, so its average time is used as a reference for the other methods. The performance ratios are:

$\text{Time}_{\text{Cubic}} / \text{Time}_{\text{Liner}} \approx 11$

$\text{Time}_{\text{Renka}} / \text{Time}_{\text{Liner}} \approx 3077$

$\text{Time}_{\text{Spline}} / \text{Time}_{\text{Liner}} \approx 7692$

Evidently, the execution speed of linear interpolation is significantly faster than all other interpolation methods, however that is only possible at the expense of accuracy. Furthermore, it is encouraging to see that the algorithm for scattered interpolation (Renka) executes much faster than MATLAB's native spline interpolation, which is superbly accurate, but, as with cubic interpolation, is only limited to Cartesian grids.

5.2.3d Virtual Impactor Simulation Data

To assess the effectiveness of Renka's interpolation scheme on a truly scattered computational grid, interpolation tests are conducted on numerical values from simulation 4 of the virtual impactor. The tests are also used to study the effect of the parameters, N_b, N_q, and N_w. The values of the velocity vector in a midplane cross-section of the device are stored, and a random sample of the stored (known) values is taken to be the set of interpolation nodes. The method is then invoked to generate an interpolated value, which can then be compared to the simulation value to estimate the error. The gradient of the velocity vector is also extracted from the FLUENT™ solution to aid in performing the linear interpolation. Figure 5.27 depicts the overall grid, and the location of the interpolation nodes, which are spread in different regions of the plane. Tables 5.7 and 5.8 summarize the absolute and relative errors, respectively. The absolute error is simply the difference between the known and interpolated value for each velocity component, whereas the relative error is the absolute error scaled by the known value.

	MAX	MEAN	RMS
Renka			
U_1	1.11431	0.01487	1.16472
U_2	0.14405	0.00876	0.34413
U_3	0.00005	0.00000	0.00007
Linear			
U_1	1.87153	0.08743	3.43974
U_2	0.91095	0.04244	1.38628
U_3	0.00014	0.00001	0.00041

Table 5.7: Absolute interpolation errors for numerical data, 185 random cells, simulation 4

The outcome of the test confirms that the linear scheme augments the interpolation error by an order of magnitude. This can be seen from the mean values of the U_1 and U_2 velocity components in Table 5.8. Clearly, it is essential to know the magnitude of the error for each velocity component because such data, for instance, contributes to the calculation of the particle Reynolds number, which in turn determines the proper drag coefficient that affects the particle trajectory. The results are also beneficial in determining the locations where the maximum interpolation errors occur, which were found to be in regions of sparse nodes at exit domains, and near boundaries.

	MAX	MEAN	RMS
Renka			
U_1	0.00452	0.00049	0.01515
U_2	0.01420	0.00062	0.02546
U_3	0.04042	0.00105	0.05568
Linear			
U_1	0.22279	0.00586	0.30767
U_2	0.08227	0.00304	0.11874
U_3	0.34183	0.00667	0.42560

Table 5.8: Relative interpolation errors for numerical data, 185 random cells, simulation 4

The choice for the parameters of Renka's method that influence its performance is investigated within the context of the above interpolation test. Recall that N_b determines the upper bound on the number of nearby nodes to include in the scattered interpolation, and N_q is the number of nodes used in the least squares fit for the nodal functions. Table 5.9 compares the rms of the relative error for different parameters.

Renka (N_b,N_q,N_w)	(34,13,32)	(34,17,32)	(40,13,32)	(46,13,32)
U_1	0.01515	0.01691	0.01795	0.02000
U_2	0.02546	0.03310	0.03133	0.03861
U_3	0.05568	0.07792	0.09006	0.16747

Table 5.9: RMS of relative interpolation error for different Renka parameters, 185 random cells, simulation 4

The errors consistently increase from left to right, for all the velocity components. Thus, it is preferable to use the values: N_b = 34, N_q =13 and N_w = 32.

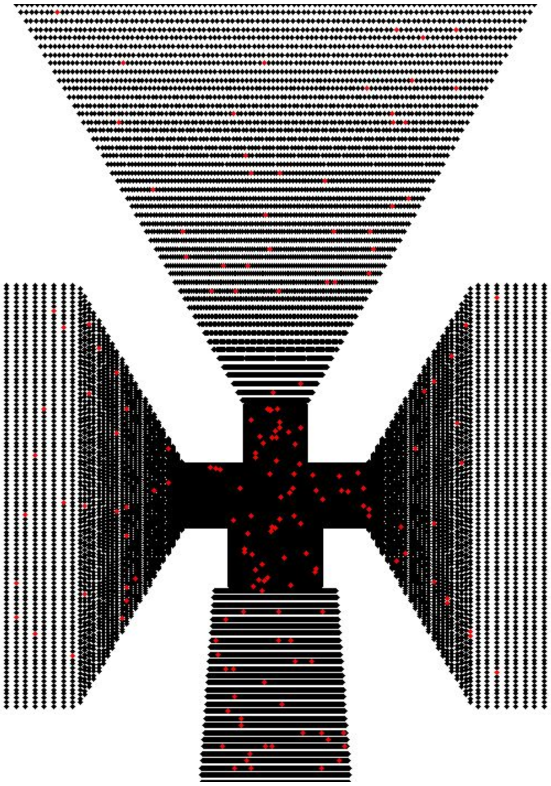

Figure 5.27: Random set of interpolation nodes in a virtual impactor grid

5.3 Summary

The analysis of the simulations convergence in the beginning of the chapter revealed the relative computational capacity of each of the turbulence models employed in terms of number of iterations, and error norms. We have seen that the predictions of the κ-ϵ model within the context of simulations 1,3,4 & 5 exhibits the most numerical error in concentrated regions close to the accelerating nozzle in the virtual impaction zone (Figures: 5.3, 5.7, 5.9, and 5.11). On the other hand, the Reynolds stress model exhibits the highest numerical error close to the side exits of the virtual impaction zone (Figure 5.5). Moreover, we were able to show that the discretization error can be reduced by resorting to a second order discretization scheme (Figures 5.9 and 5.11).

In the second part of the chapter, we established the necessary components needed to conduct accurate particle tracking calculations. The accuracy of trajectory calculations is more dependent on the number of particles when stochastic tracking is used rather than tracking solely with the mean fluid velocity. Lastly, we presented the outcome of extensive testing for the multi-variate scattered point interpolation method. The tests demonstrated the improved accuracy of this method over linear interpolation. We have shown that such a method is capable of producing third order accuracy in irregular geometry grids. The only drawback, however, is its substantial computational cost.

Chapter 6

Simulation Results

The analysis and comparison of the simulations conducted for the virtual impactor are presented in this chapter. In Section 6.1, detailed properties of the fluid flow field are plotted and compared for the different simulation models. In Section 6.2, the outcome of the particle tracking runs is shown, and a brief comparison with experimental data is exhibited.

6.1 Properties of the Fluid Flow Solution

The behavior of the fluid flow inside the device is analyzed for each of the simulations listed in Table 5.1. It should be noted that any simulation by itself does not necessarily provide an accurate picture of the physical flow dynamics. For this reason, the approach to modeling the virtual impactor is refined and “tweaked” from one simulation to the next so that ultimately we can arrive at a precise and comprehensive simulation. Nevertheless, the exploitation of distinct turbulence models, different discretization schemes and computational meshes, allows us to discover the necessary components that constitute a successful simulation.

In the following sections, we compare and contrast many of the aspects that characterize the flow solution. Contour plots of velocity magnitude, turbulent kinetic energy, pressure, and temperature are shown. In addition, plots of the velocity vectors in the nozzle region are presented. Profiles of the mean velocity in different sections of the device are also considered.

6.1.1 K-ϵ vs. RSM

The discussion in this section focuses on the results of the incompressible flow taken from the numerical data of simulation 1 and 2. The velocity magnitude contours in the midplane cross-section of the device are shown in Figures 6.1 and 6.2 for the k-epsilon and the Reynolds stress models, respectively. The contours are quite similar in the entrance section of the device, however, they are quite different beyond the accelerating nozzle, especially as the fluid is exiting into the major flow. To better characterize those differences, the velocity profiles in the interesting sections are plotted next.

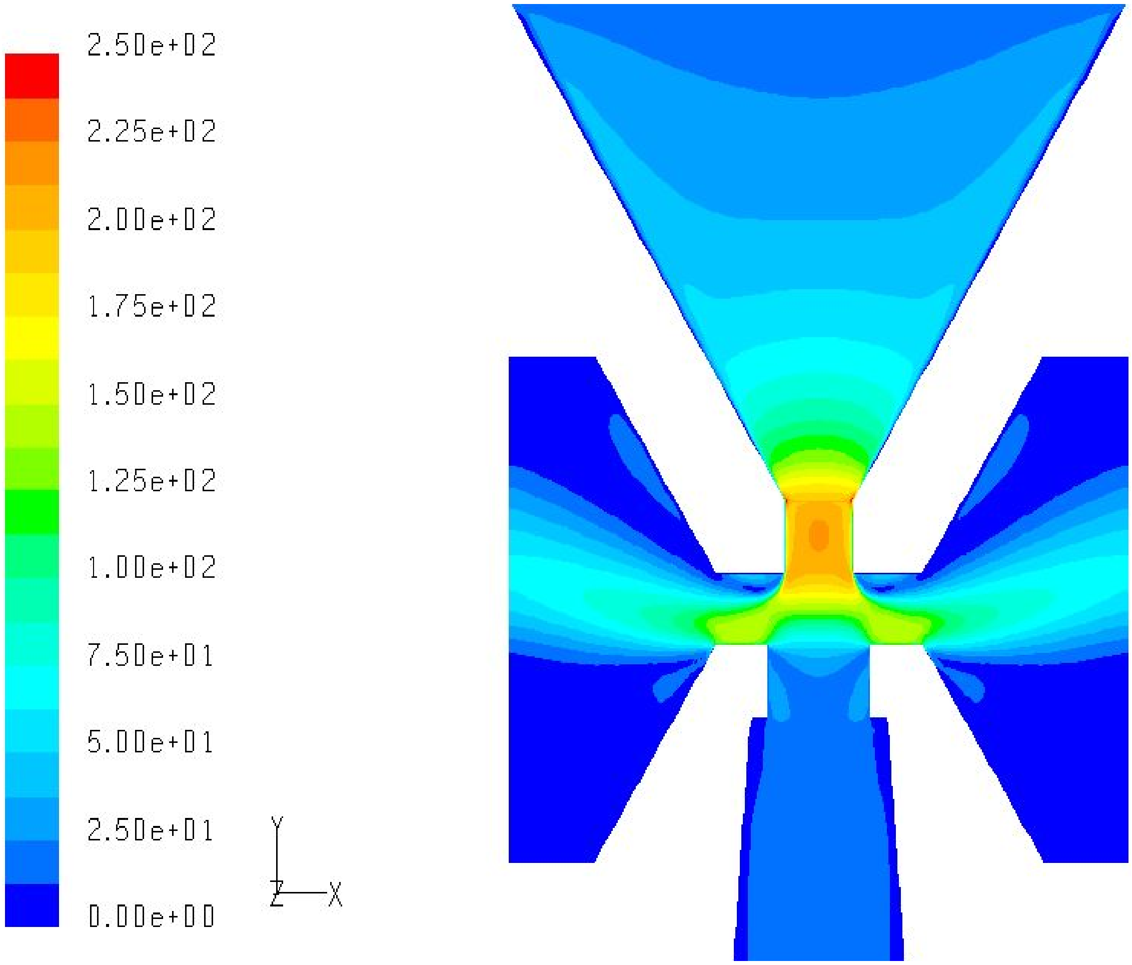

Figure 6.1: Midplane contours of velocity magnitude (m/s), simulation 1

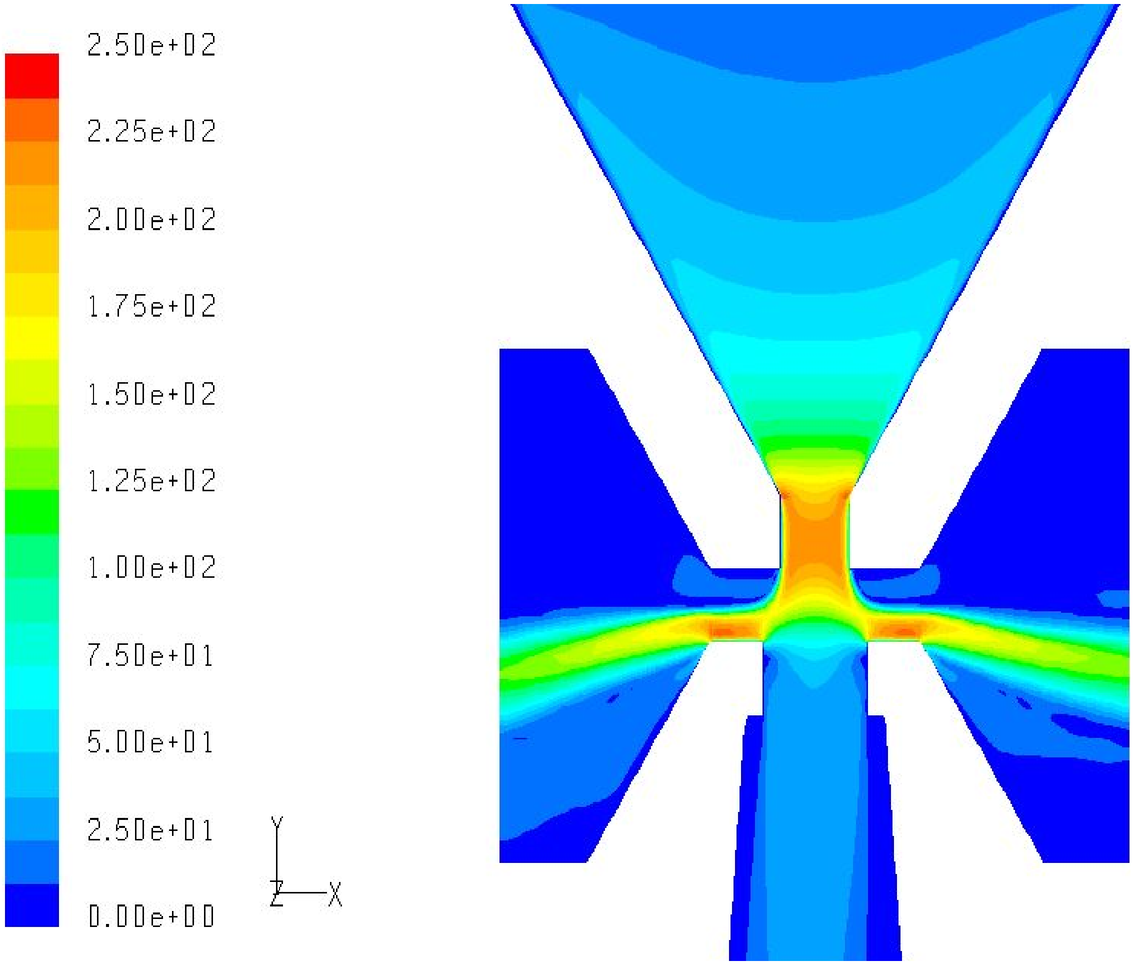

Figure 6.2: Midplane contours of velocity magnitude (m/s), simulation 2

First, the flow profile in the throat section is analyzed. Figure 6.4 shows the dimensionless streamwise velocity, $U^+_y = U_y/U_\tau$, as a function of the dimensionless wall unit, x^+, at y=-W/2, which is halfway from the entrance of the throat (y=0), as marked in Figure 6.3. The "streamwise" direction is taken in the negative Y-axis, and the "cross-stream" direction in the positive X-axis. The friction velocity is $U_\tau = \sqrt{\frac{\tau_w}{\rho}}$, and the wall unit is defined as: $x^{+} = \frac{U_{\tau} d_{x}}{\nu}$ where d_x is the distance from the throat wall.

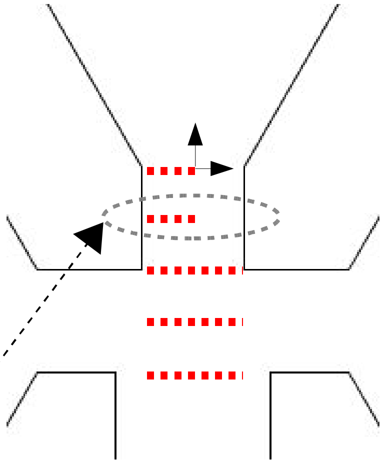

Figure 6.3: Schematic of throat cross sections at y=0, y=−0.5W, y=−W, y=−1.5W, and y=−2W (from top to bottom)

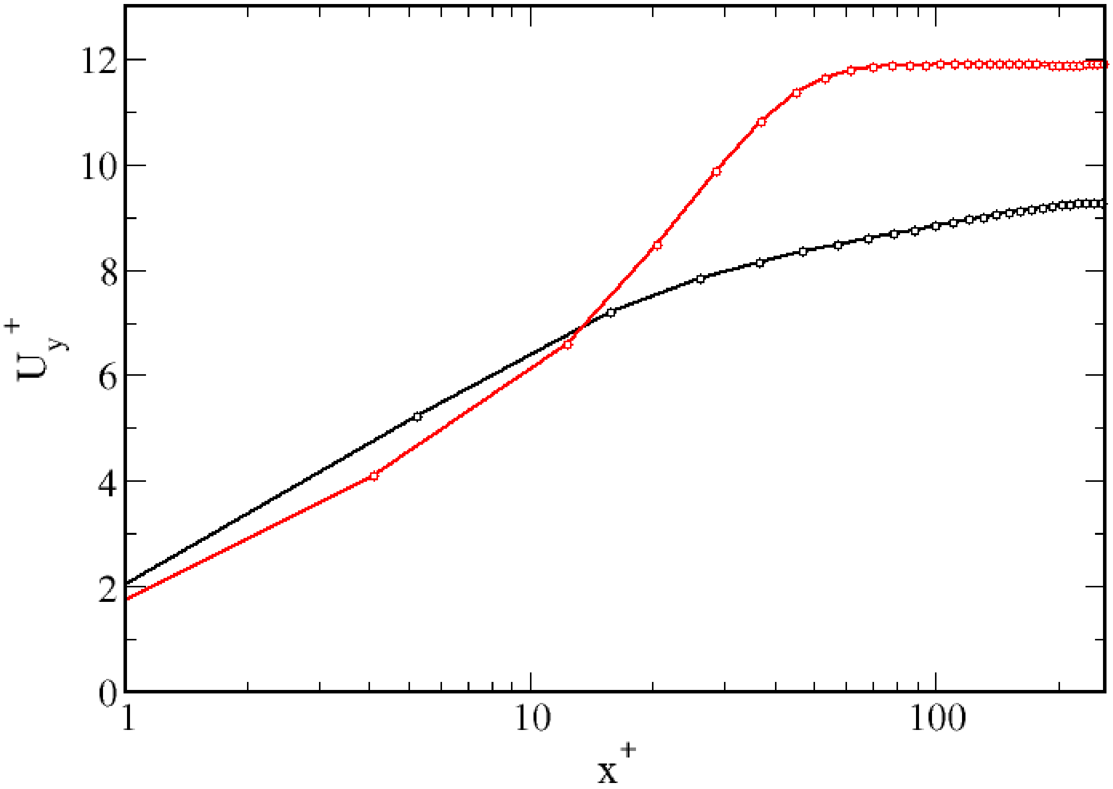

Figure 6.4: Dimensionless streamwise velocity profile in throat, simulation 1 (black), simulation 2 (red)

As can be seen from the figure, and from the contour plots, the Reynolds stress model predicts a steeper gradient than the k-epsilon model. This will eventually influence the behavior of the fluid as it exits the nozzle, and may account for the increase in the ratio Q_m/Q_T (from 0.19 to 0.22) despite the use of the same pressure boundary conditions. The plot is also useful in revealing the grid resolution next to the wall. The first meshpoint ($x^+ \approx 4$ to 5) for each simulation is in the viscous sublayer, and the linear dependence of mean velocity on the wall unit is depicted.

The flow profile at the nozzle, and in the expansion zone (see Figure 5.2) is analyzed by looking at the vector plots. Figures 6.5 and 6.6 reveal the flow structure in the separation zone for the two simulations, respectively. The behavior of the fluid in the streamwise direction out of the nozzle is roughly equivalent in each of the simulations, however, the separation phenomenon into the major flow is predicted differently. In the k-epsilon simulation, the flow separates early in the exit stage, which results in the formation of a relatively large vortex (re-circulation) next to the upper nozzle wall. The Reynolds stress model, on the other hand, predicts a higher entrance velocity to the major flow, thus delays the formation of the vortex, which occurs further away from the nozzle, and at a much smaller scale. To quantitatively verify this observation, the velocity profiles are plotted for both the streamwise and cross-stream directions.

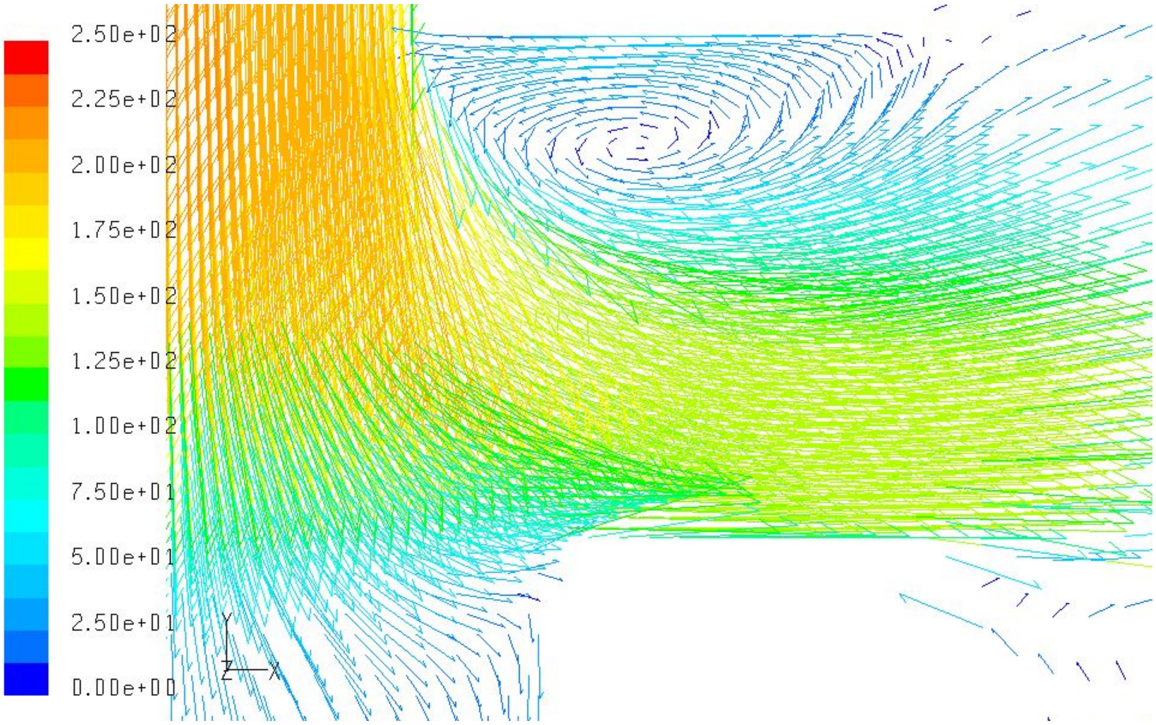

Figure 6.5: Mean velocity vectors (m/s), flow separation, simulation 1

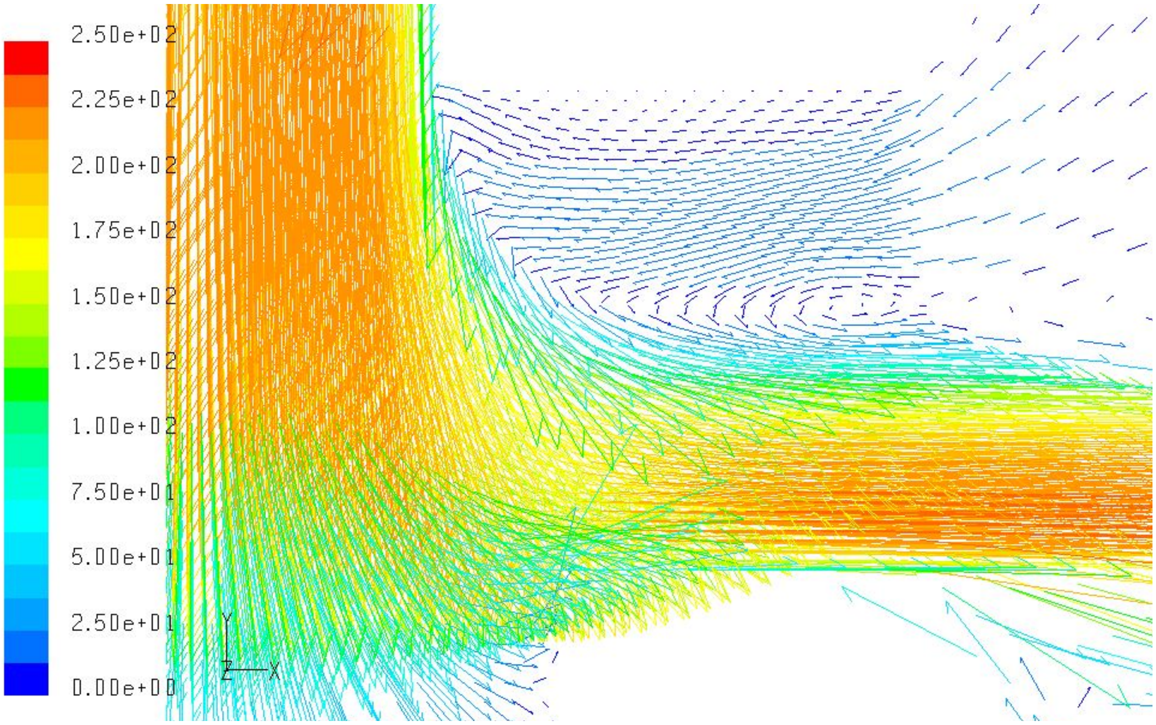

Figure 6.6: Mean velocity vectors (m/s), flow separation, simulation 2

Figure 6.7 shows the streamwise velocity scaled by the average velocity, U_{avg}, at the nozzle, which is taken to be the ratio of the volumetric flow rate at STP to the area of the nozzle (LW). The three horizontal cross-sections where the profiles are plotted are (see Figure 6.3):

- at y=-W: nozzle
- at y=-1.5W: halfway to the collection probe
- at y=-2W: collection probe

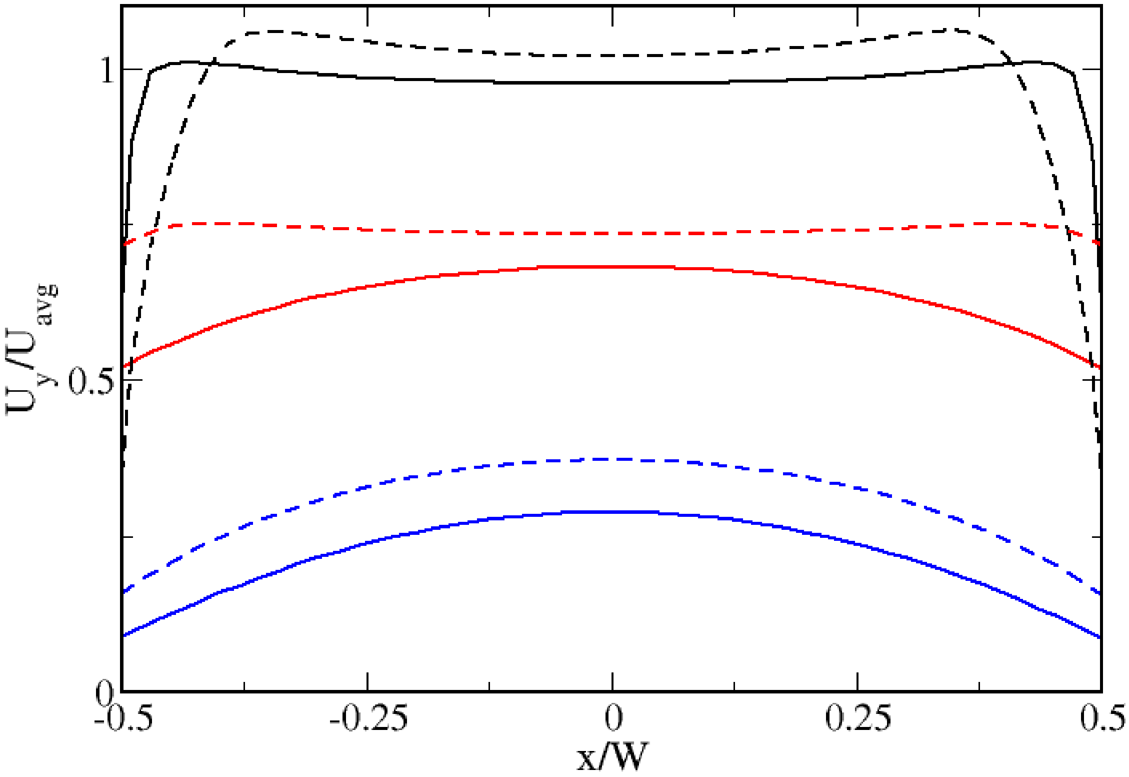

Figure 6.7: Streamwise velocity profile at nozzle and beyond, y=-W, y=-1.5W, y=-2W, simulation 1 (solid lines), simulation 2 (dashed lines)

Evidently, the fluid is moving faster in the RSM simulation, and this slight overshoot is sustained all the way to the collection probe. The flow behavior to the side of the nozzle is examined through Figures 6.9, and 6.10 which show the cross-stream scaled velocity profile at the following consecutive locations (Figure 6.8):

- at x = 0.75W: edge of collection probe
- at x = 1.125W: halfway to major flow
- at x = 1.5W: major flow entrance

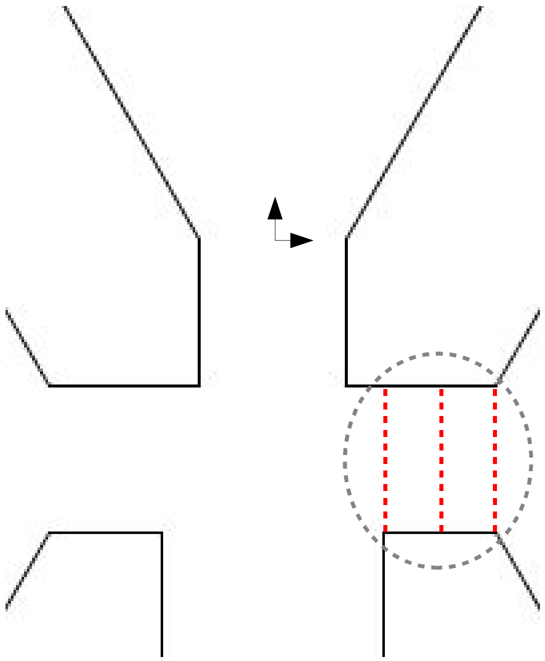

Figure 6.8: Schematic of expansion nozzle cross sections at x = 0.75W, x = 1.125W, and x = 1.5W (from left to right)

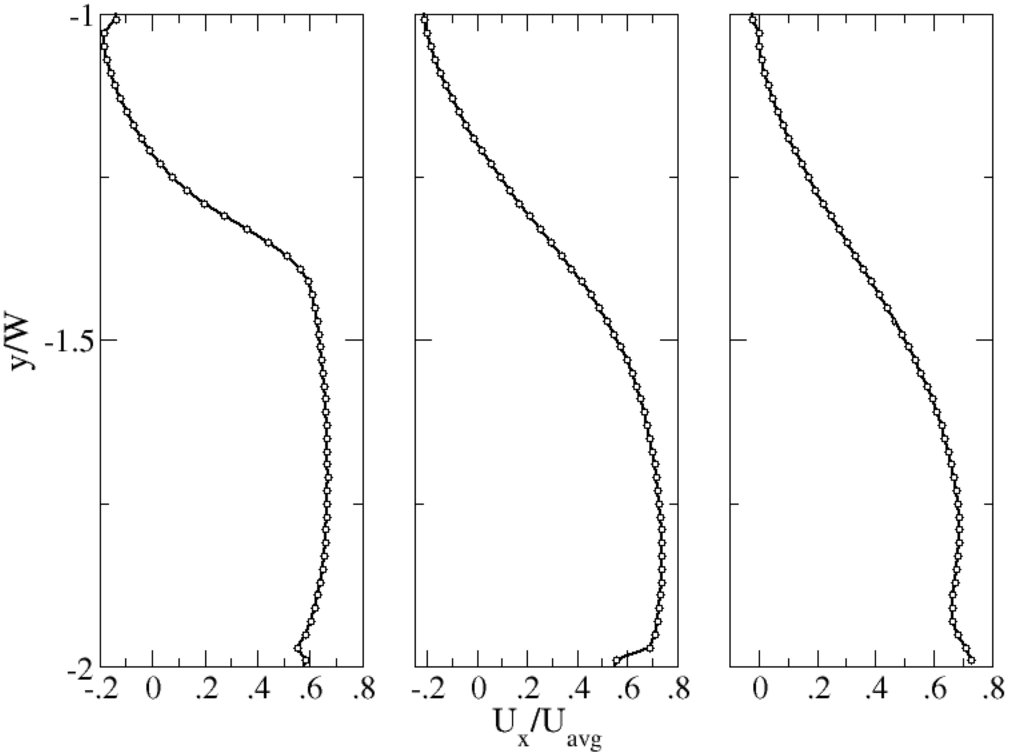

Figure 6.9: Cross-stream velocity profile of exiting fluid, simulation 1

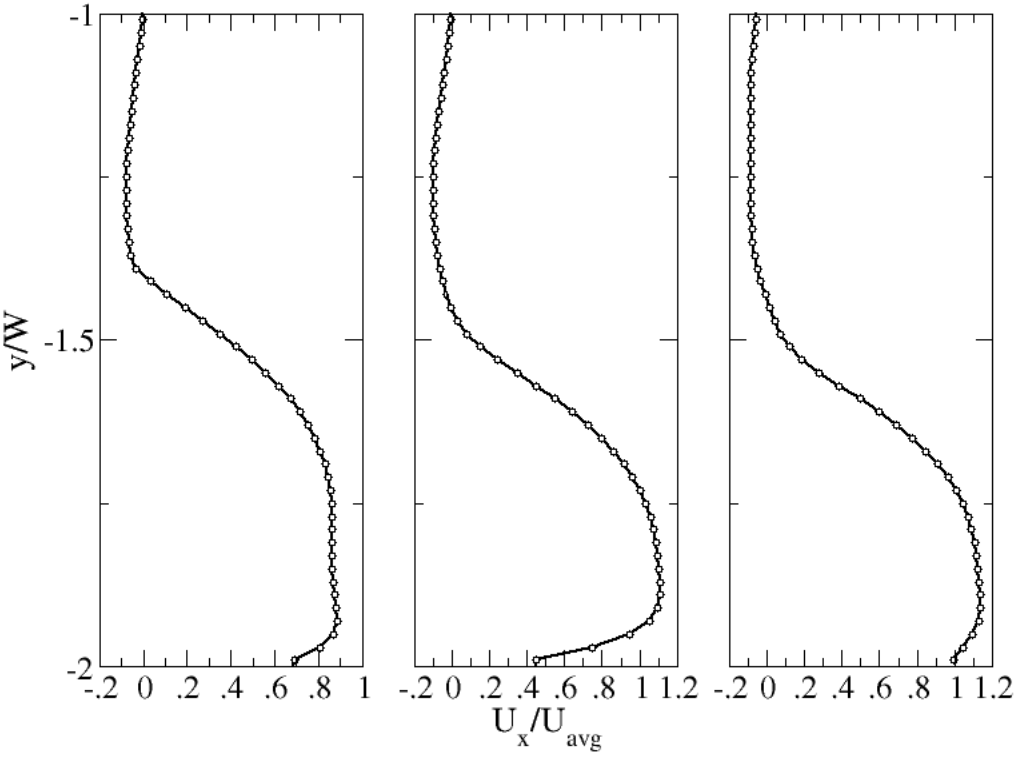

Figure 6.10: Cross-stream velocity profile of exiting fluid, simulation 2

Two striking dissimilarities are evident from these two plots. First, the maximum velocity magnitude of the expanding fluid in the RSM simulation exceeds that in the k-epsilon simulation. Second, the location of the stagnation or inflection point, identified as $U_x/U_{avg}=0$, is consistently moving upwards as predicted by the k-epsilon model, whereas the RSM prediction shows the opposite behavior, as seen by its gradual decent from left to right. The plots are also useful in revealing the role of the wall functions. Apparently, the RSM solution is more inline with the wall function values for the mean velocity, which is evident by the smooth variation away from the bottom wall. The k-epsilon predictions, however, are somewhat lagging to those of the wall function values, which results in a slightly disrupted flow profile next to the bottom wall (first and third locations). Recall that the values of the mean velocity at the first few cells away from the wall are predicted by the wall functions.

Lastly, a comprehensive contour plot of the magnitude of the velocity difference between the two models is shown. Analogous to what is done in Section 5.1, the difference norm is defined as:

$$|\overline{U}(K\epsilon)-\overline{U}(RSM)|=\sqrt{\sum_{i=1}^{3}[\overline{U}_i(K\epsilon)-\overline{U}_i(RSM)]^2} \tag{6.1}$$

The norm is then scaled by the average nozzle velocity ($U_{avg} \approx$ 203 m/s), and plotted for the virtual impaction zone of Figure 5.2 in Figure 6.11.

The contour plot re-enforces the previous findings that there are considerable differences as the flow exits the acceleration nozzle, which can now be quantified as having an upper bound of 60 %.

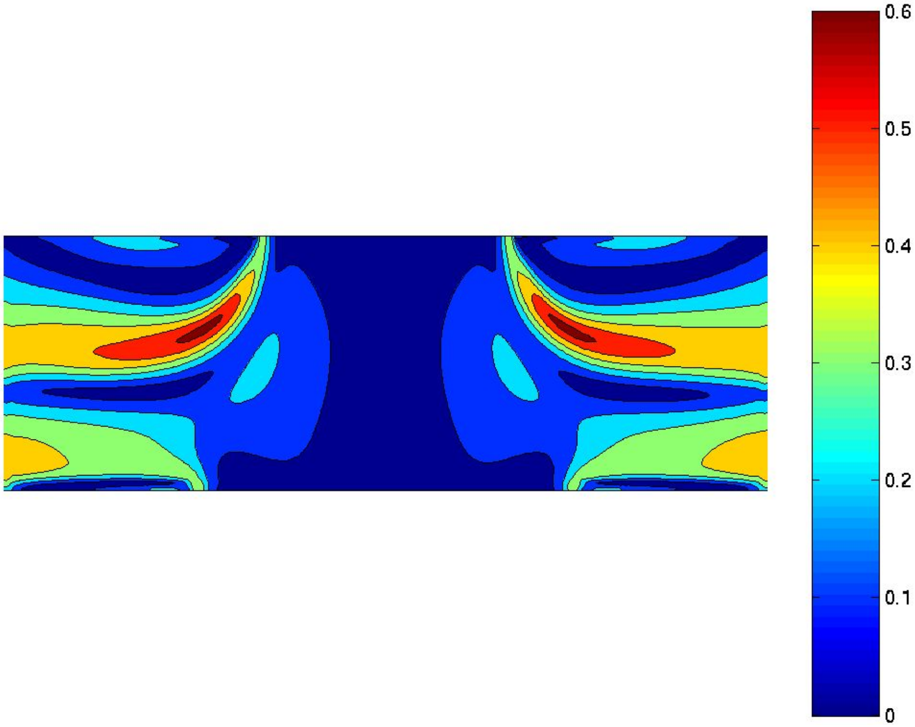

Figure 6.11: Contours of the norm of velocity difference between the predictions of simulations 1 and 2, scaled by U_{avg}

A similar comparison is done for the pressure field, and it is found that the discrepancy is minimal, thus we only show the pressure variation in the virtual impaction zone as predicted by the Reynolds stress model. Figure 6.12 shows the absolute pressure drop in the streamwise direction taken at x = 0 (midpoint of nozzle), while Figure 6.13 depicts the pressure drop in the cross-stream direction taken at y = -1.5W (halfway to the collection probe, see Figure 6.3). Both figures show a pressure behavior consistent with the flow dynamics, and with the pressure boundary condition set at the major flow exit. From Figure 6.12, it is evident that the flow is gradually compressed in the cone section of the device, until it approaches the throat at y=0, where it experiences a significant pressure loss, due to the fluid acceleration. The flow then begins to expand out of the accelerating nozzle at y/W = -1, and gradually regains its ambient pressure at the minor flow exit.

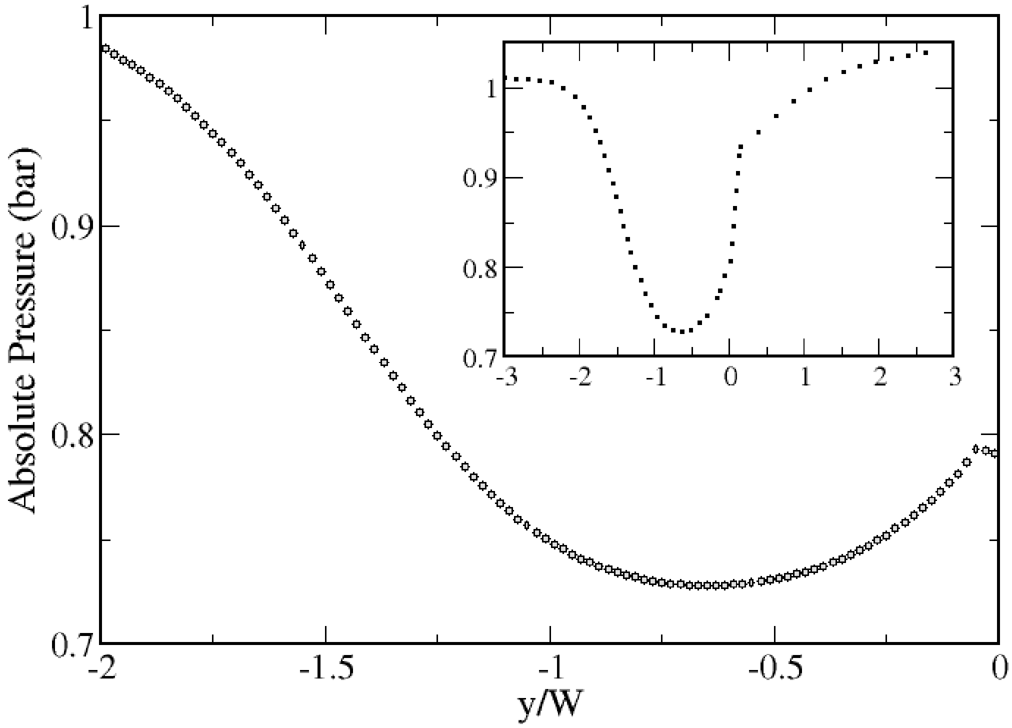

Figure 6.12: Absolute pressure as a function of streamwise distance from nozzle at x=0, simulation 2 (insert shows pre-nozzle data)

Figure 6.13, shows the flow expansion phenomenon in the cross-stream direction. The high speed fluid in the middle of the device experiences a pressure drop equivalent to the major flow outlets pressure boundary conditions, and consequently reduces its speed.

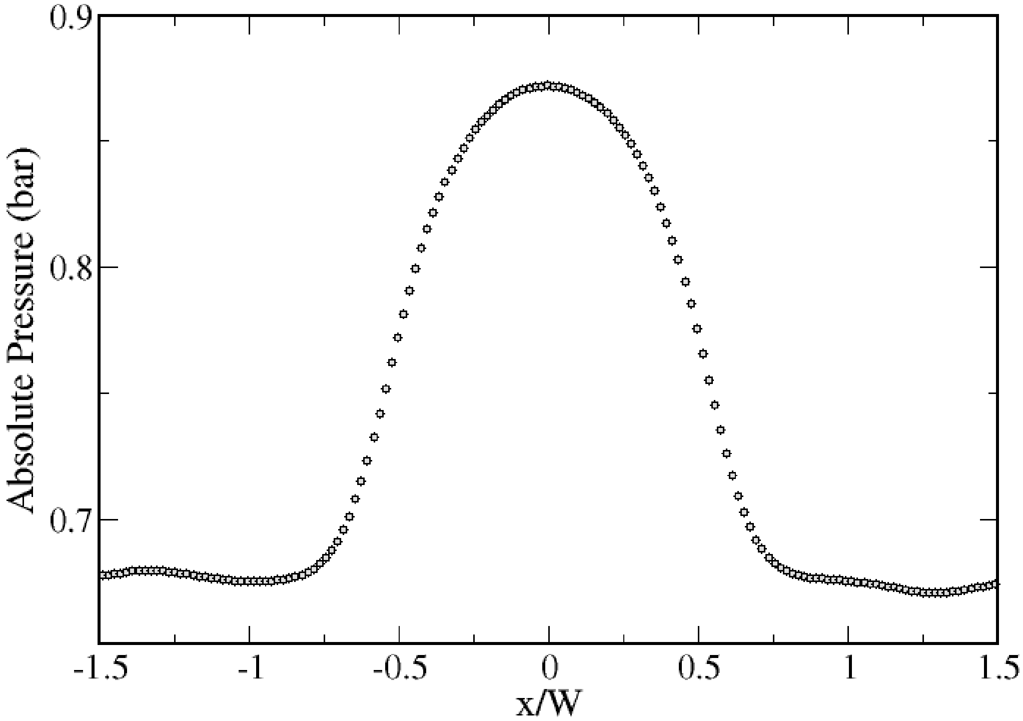

Figure 6.13: Absolute pressure as a function of cross-stream distance, midpoint of virtual impaction zone, simulation 2

6.1.2 Incompressible vs. Compressible Flow

Simulations 1 and 3 are both performed using the k-epsilon model, on the same grid. The only difference, as mentioned before, is that the compressible form of the RANS equations does not consider the density of the fluid to remain constant, but allows it to vary with temperature and pressure as dictated by the ideal gas law. Additionally, an extra term is included in the kinetic energy equation as noted in Section 4.1.2. Both simulations used the same set of boundary conditions. It was found that a higher pressure drop is needed at the major flow exit in order to achieve close to 20% flow separation in the compressible calculation.

Figure 6.14 shows the velocity magnitude contours in the midplane cross-section, and Figure 6.15 is a vector plot of the flow past the acceleration nozzle for simulation 3. When compared to their counterparts of simulation 1 (Figure 6.1 and 6.5), it is evident from the contour values that the maximum velocity of the compressible flow is approximately 100 m/s faster than that of the incompressible flow. This can be attributed to density variations that lead to more significant pressure gradients. The structure of the separation phenomenon in the post nozzle region is similar to that of simulation 1, understandably, because the k-epsilon model is used. To further assess the predictions of simulation 3, a series of profile plots are shown. The expansion of the gas into the major and minor flow is depicted in Figures 6.16 and 6.17, respectively. The dimensionless velocity profile in the throat section is also computed, as before, and compared to that of simulation 1 in Figure 6.18.

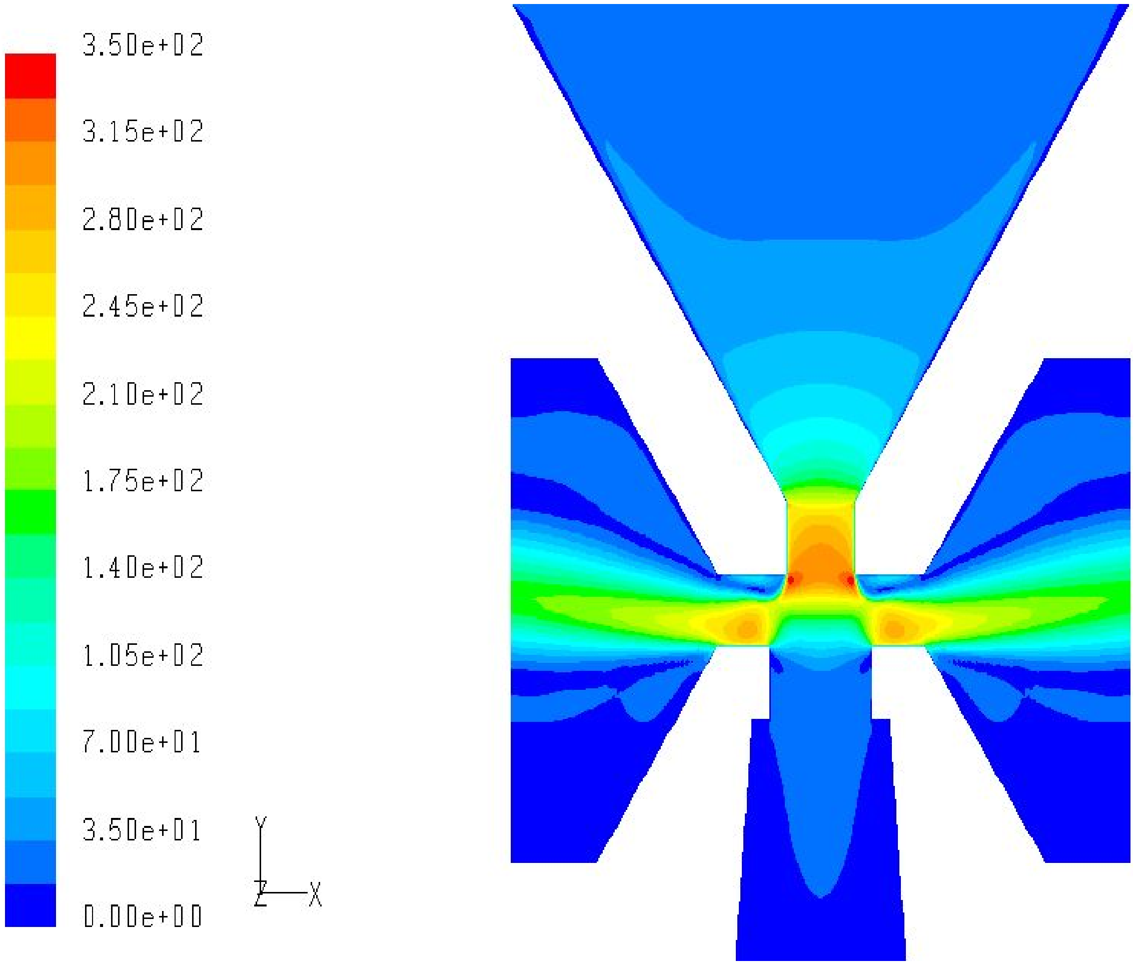

Figure 6.14: Midplane contours of velocity magnitude (m/s), simulation 3

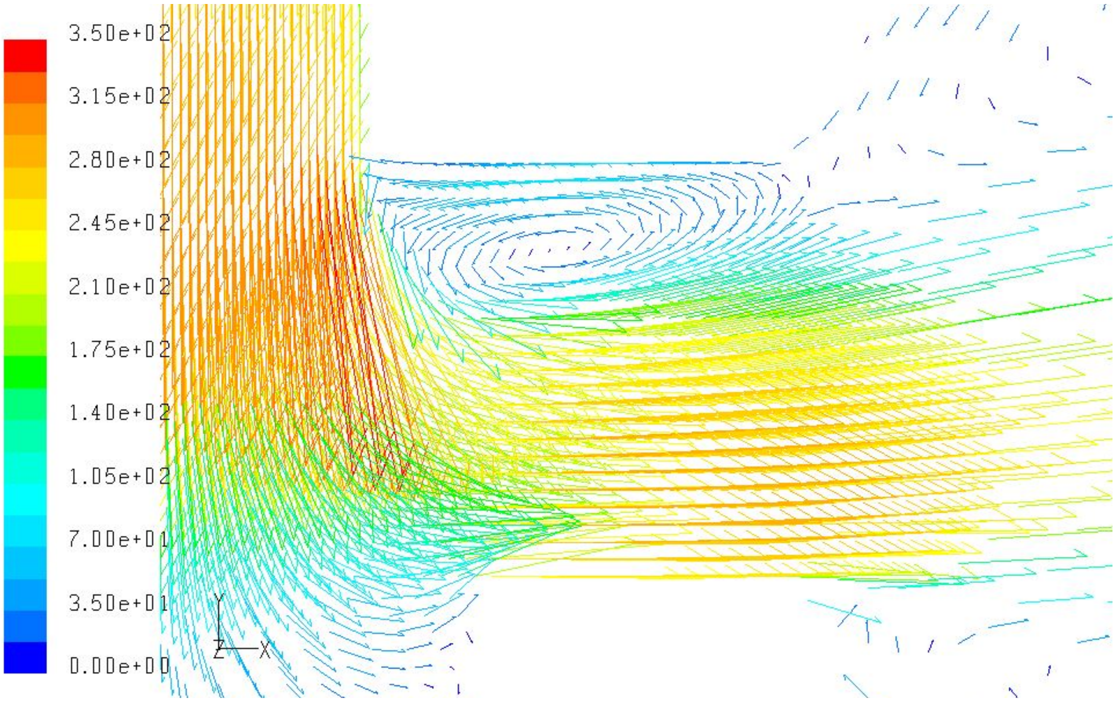

Figure 6.15: Mean velocity vectors (m/s), flow separation, simulation 3

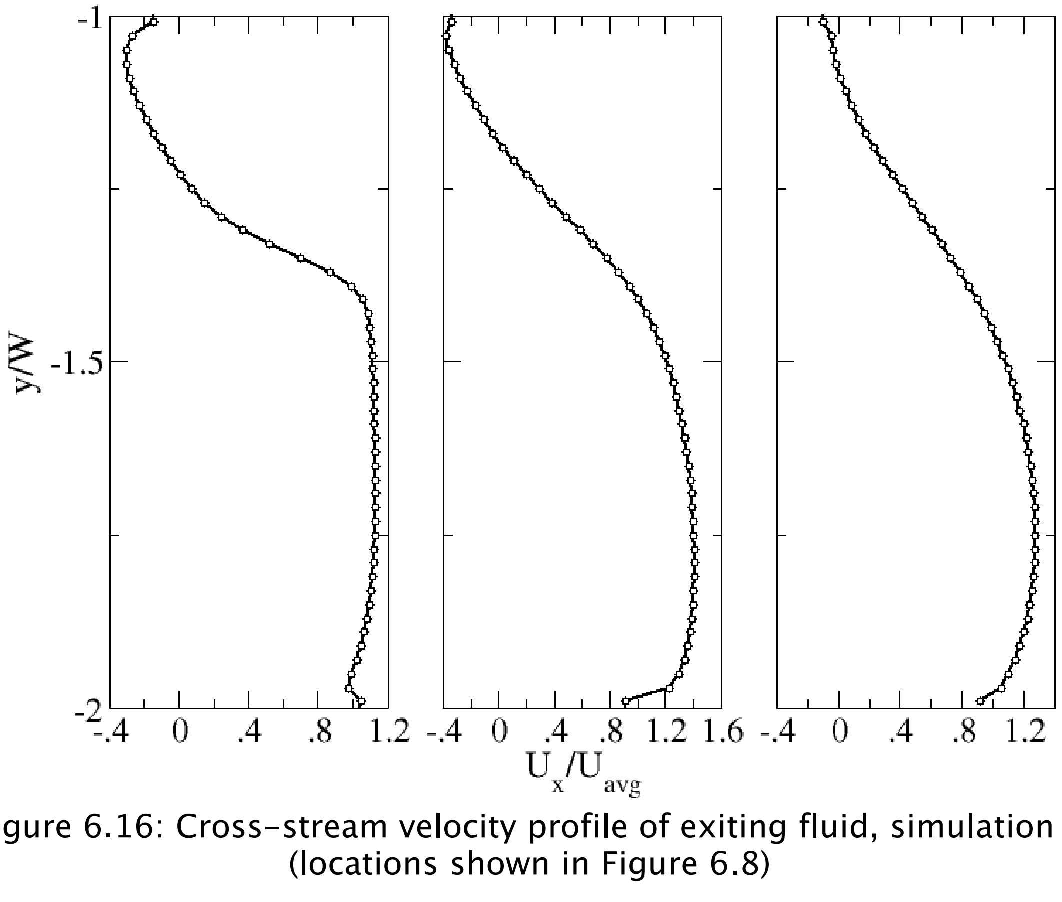

Figure 6.16: Cross-stream velocity profile of exiting fluid, simulation 3
(locations shown in Figure 6.8)

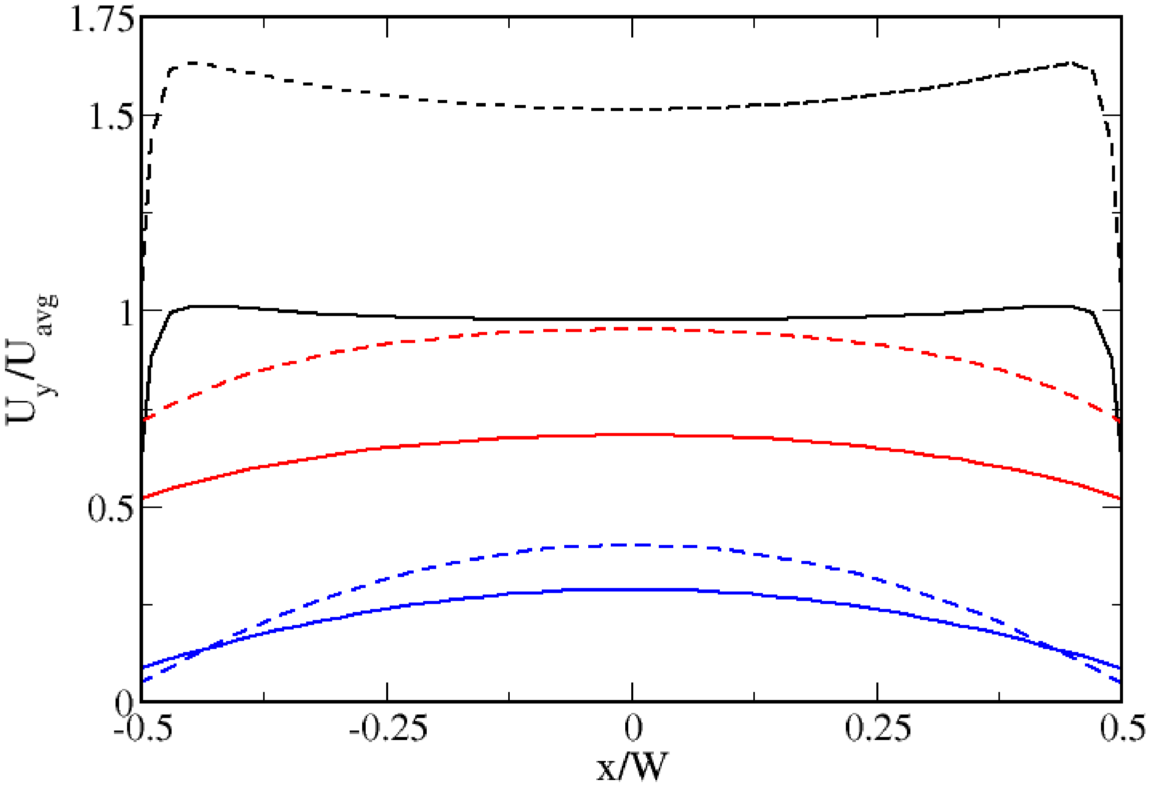

Figure 6.17: Streamwise velocity profile at nozzle and beyond, y=−W, y=−1.5W, y=−2W, simulation 1 (solid lines), simulation 3 (dashed lines) (locations shown in Figure 6.3)

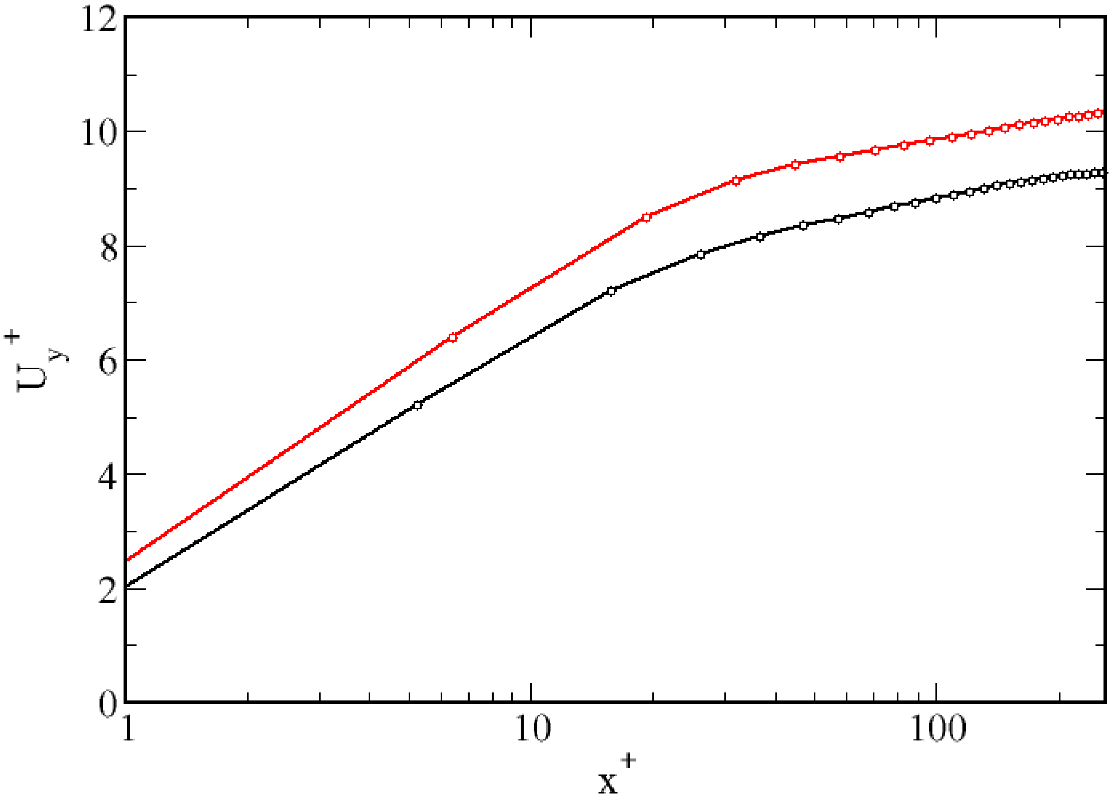

Figure 6.18: Dimensionless streamwise velocity profile in throat, simulation 1 (black), simulation 3 (red)

The velocity profiles of the compressible flow clearly indicate the affinity of the fluid to expand to a much higher velocity out of the accelerating nozzle. The separation or recirculation behavior into the major flow also occurs at a much larger scale as can be seen by the relatively significant backward motion (Figure 6.16), which will naturally contribute to a much higher pressure drop. A comprehensive contour plot of the magnitude of velocity difference between the incompressible and compressible results is shown in Figure 6.19, and a similar plot for the pressure difference in Figure 6.20 for the virtual impaction region of Figure 5.2.

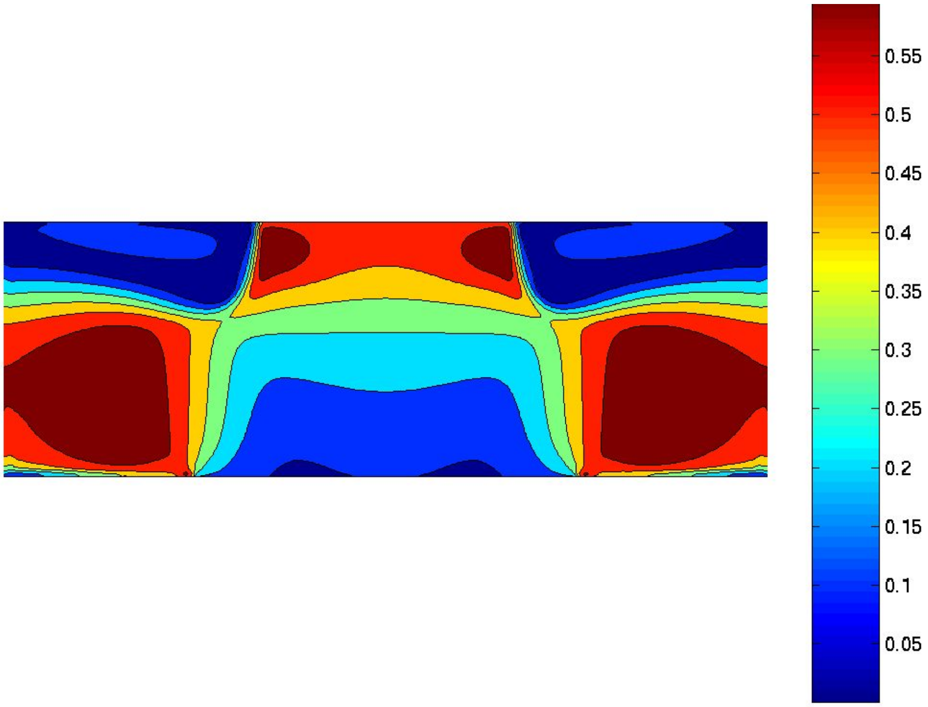

Figure 6.19: Contours of the norm of velocity difference between the predictions of simulations 1 and 3, scaled by U_{avg}

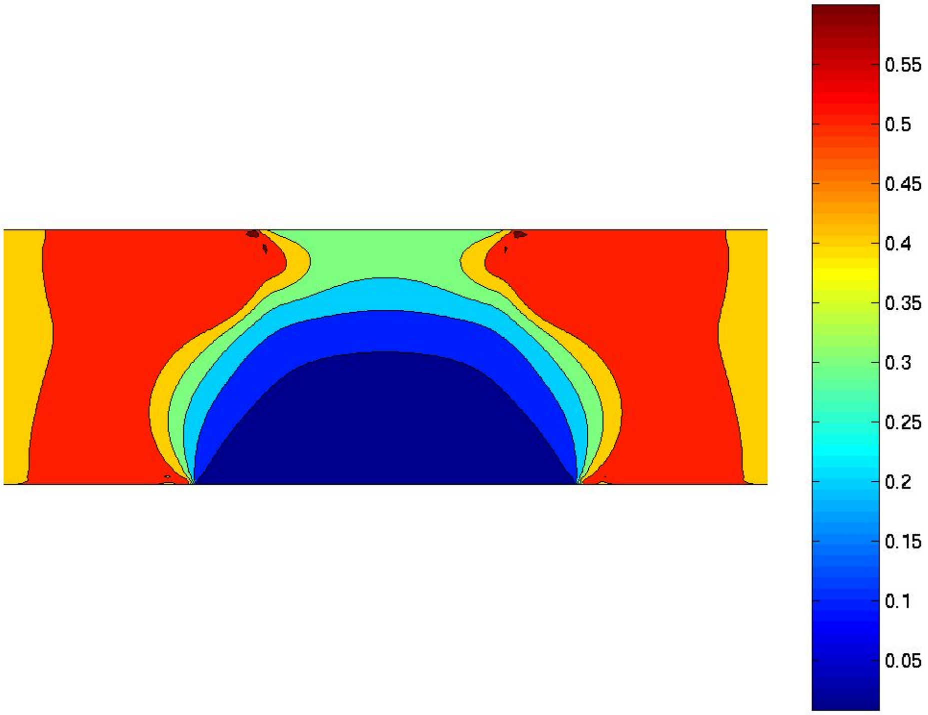

Figure 6.20: Contours of absolute pressure difference between the predictions of simulations 1 and 3, scaled by P_{abs} of simulation 3

The contour plots clearly reveal the regions where compressibility plays a prominent role. It is evident that the fluid expansion right out of the nozzle, and its divergence into the major flow constitute the fundamental dynamics that influence the behavior of the fluid motion. It is interesting to see that the velocity and the pressure differences are correlated by the same upper bound of 60 % at the aforementioned locations. The actual pressure variation will be shown in the next section using the data from simulation 5.

6.1.3 Coarse vs. Fine Grid

The refinement of the computational grid is expected to improve the accuracy of the calculations, however, it should not alter the underlying conclusions made about the flow behavior. As shown in Table 5.1, the separation ratio Q_m/Q_T, does, however, increase from 16% to 22%. To support this argument, velocity profiles obtained for the compressible flow numerical data of simulations 3 and 5 are compared. Figure 6.21 shows the streamwise velocity profile in the throat (y=-W/2), while Figure 6.22 presents the profile at the nozzle and virtual impaction zone (see Figure 6.3). Clearly, the first plot indicates an improvement in resolving the flow close to the wall (first meshpoint at $x^+ \approx 2.5$). The second plot reveals the obvious similarity in the coarse and refined solutions beyond the nozzle.

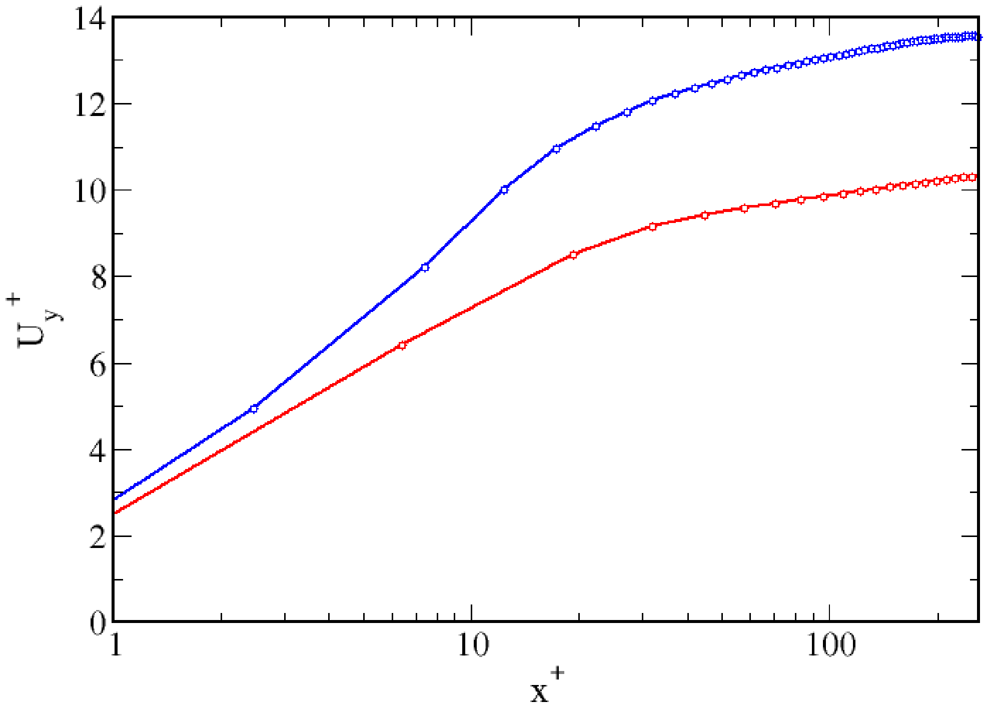

Figure 6.21: Dimensionless streamwise velocity profile in throat, simulation 3 (red), simulation 5 (blue)

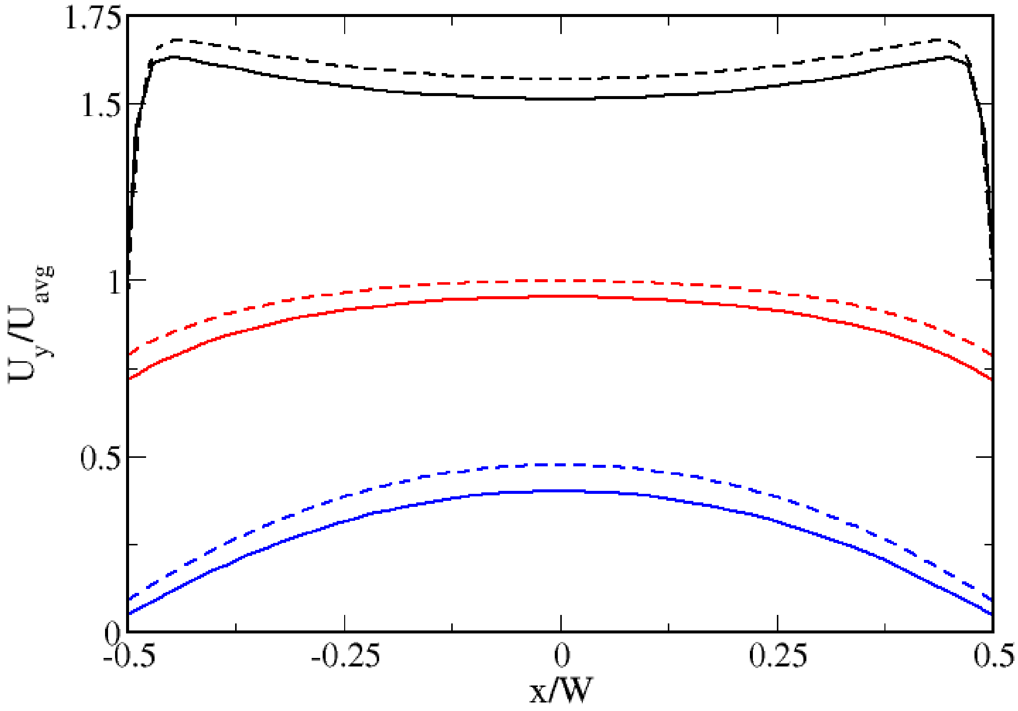

Figure 6.22: Streamwise velocity profile at nozzle and beyond, y=−W, y=−1.5W, y=−2W, simulation 3 (solid lines), simulation 5 (dashed lines)

The discrepancy in the magnitude of the dimensionless velocities of Figure 6.21 is due to the different friction velocities used in the scaling. The computed values of U_τ are 28.36 and 21.82 m/s for simulation 3 and 5, respectively. As expected, the refined grid calculation gives a smaller friction velocity due to the shift of the first mesh point to a closer position to the wall, thus, better estimating the velocity gradient. On the other hand, the differences in Figure 6.22 can be attributed to the change in Q_m/Q_T ratio from 0.16 for simulation 3 to 0.22 for simulation 5, therefore, slightly increasing the velocity of the fluid into the minor flow, despite the identical pressure boundary conditions.

The expansion of the fluid to the peripheral outlet is depicted in Figure 6.23 for simulation 5. The profiles at the three vertical cross-sections (Figure 6.8) are very much similar to the ones shown in Figure 6.16, judging by their shapes and velocity scales.

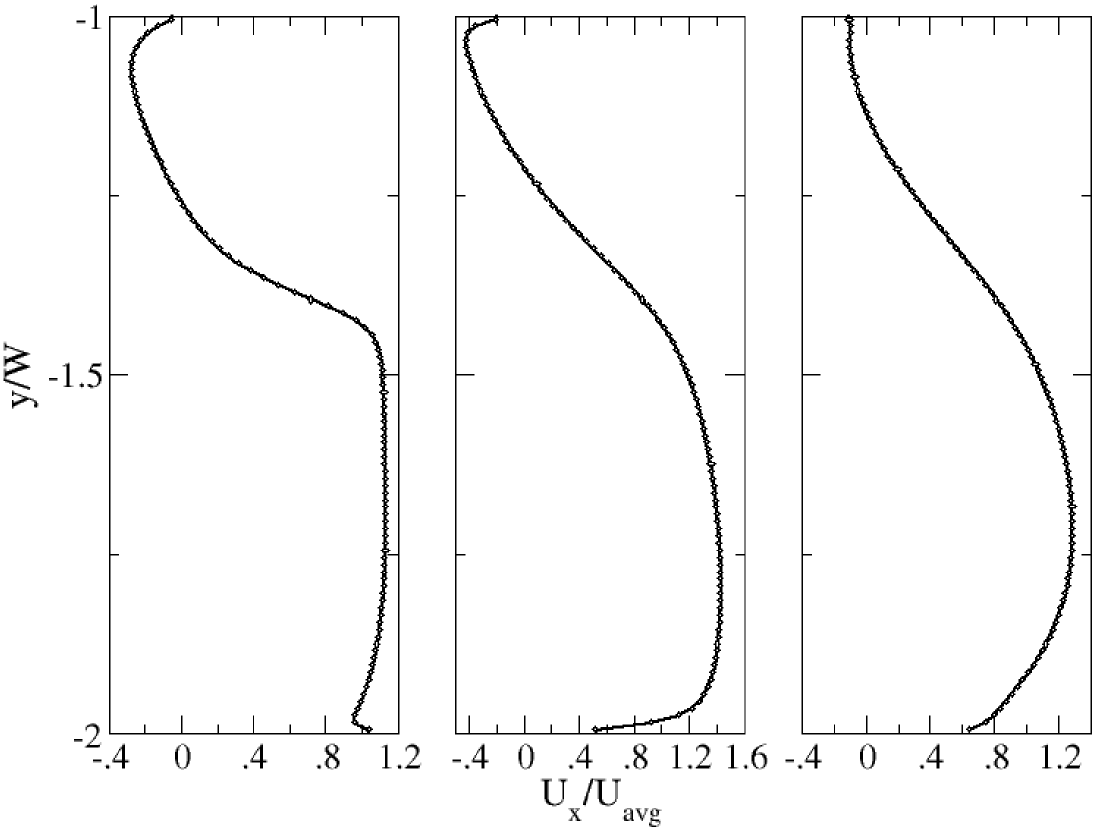

Figure 6.23: Cross-stream velocity profile of exiting fluid, simulation 5

The compressible flow pressure distribution is shown in Figures 6.24 and 6.25 using the numerical data obtained from simulation 5, since no considerable differences were found to exist between the coarse and fine mesh results. As is the case in the incompressible flow simulation, the variation of the pressure in the streamwise and cross-stream directions is indicative of the compression and expansion phenomena experienced by the fluid in the nozzle and post nozzle sections of the device. The pressure drop, however, is slightly higher in this case, approximately 0.4 bar, naturally due to the preset major flow boundary conditions.

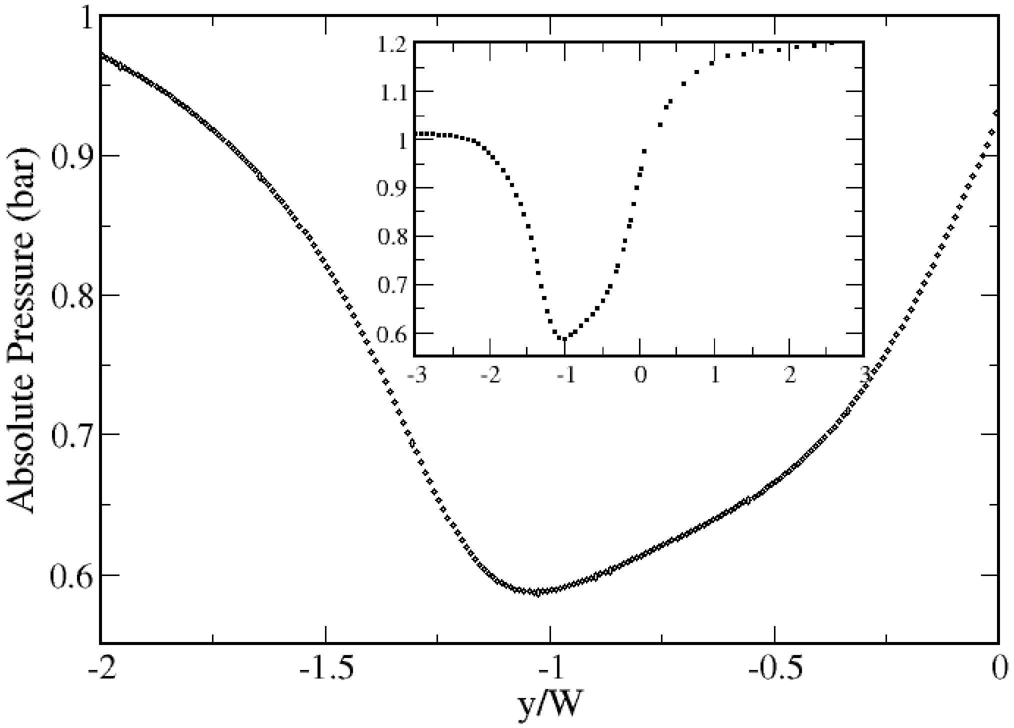

Figure 6.24: Absolute pressure as a function of streamwise distance from nozzle at x=0, simulation 5 (insert shows pre-nozzle data)

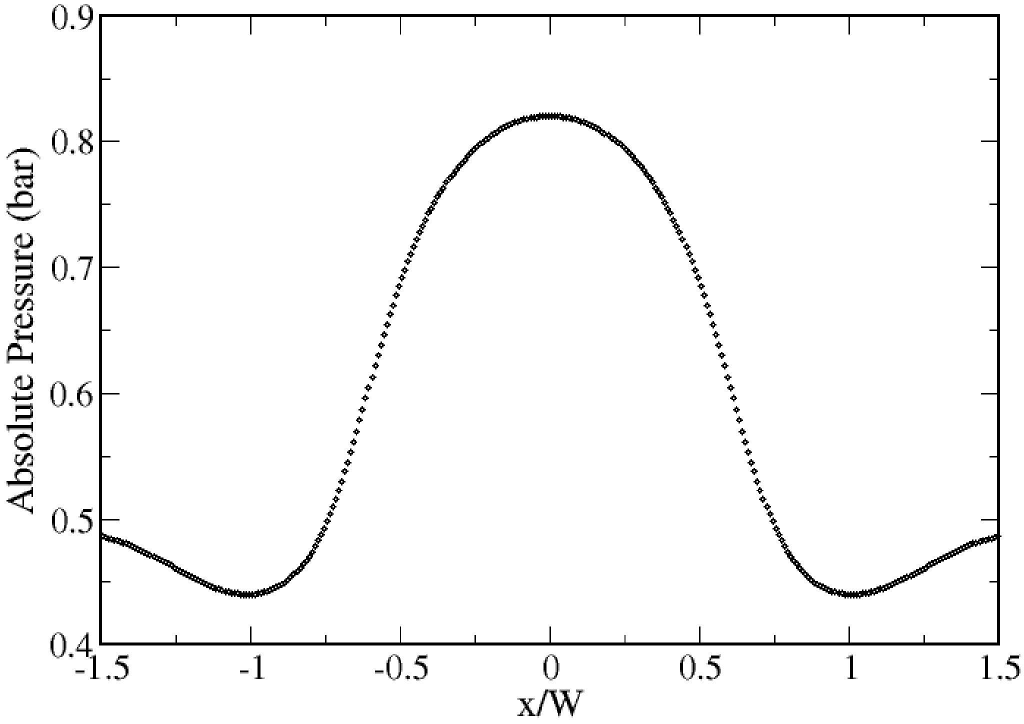

Figure 6.25: Absolute pressure as a function of cross-stream distance, midpoint of virtual impaction zone, simulation 5

6.1.4 1^{st} vs. 2^{nd} Order Discretization

The justification for going to a second order discretization lies in the premise that first order discretization is prone to more numerical diffusion, especially in flows with streamline curvature. For this reason, we compare the results of simulation 4 and 5 which were both conducted using the same governing equations (compressible flow; K-ϵ turbulence model), the same operating and boundary conditions, but with different discretization schemes. Initially, the goal of this study was purely numerical, however as it turns out, the second order discretization scheme does in fact capture some features of the flow that are missed by the first order scheme, thus influencing the discrete phase results, as will become evident in Section 6.2.

The main focus of the comparison between the two simulations deals with the solution predictions in the virtual impaction zone. As done in the previous sections, comprehensive contour plots of the differences in solution variables will be presented. To put the difference plots in perspective, the actual contour plots are first addressed. The velocity contours are similar in appearance to those shown for simulation 3 (Figure 6.14) and will not be presented here. That is, differences due to grid density can not be distinguished from such a plot. Figure 6.26 shows the turbulent kinetic energy, Figure 6.27 is the absolute pressure, and Figure 6.28 is the temperature contours, for the numerical data of simulation 5. The turbulent kinetic energy contour plot reveals the region with the highest energy generation, which is in the shear layer at the bottom walls of the nozzle. The scale of energy generation appears to be proportional to the square of the cross-stream velocity ($\approx \frac{1}{2} U^2_x$). The temperature contour plot shows the cooling of the fluid associated with the high speed regions, as the fluid enters the throat, and as it is carried into the major flow. The temperature profile is further depicted in Figures 6.29 and 6.30, which show the variation in the streamwise and cross-stream directions, respectively. The temperature profiles resemble those of the pressure in Figures 6.24 and 6.25 because of the linear relationship of the ideal gas law.

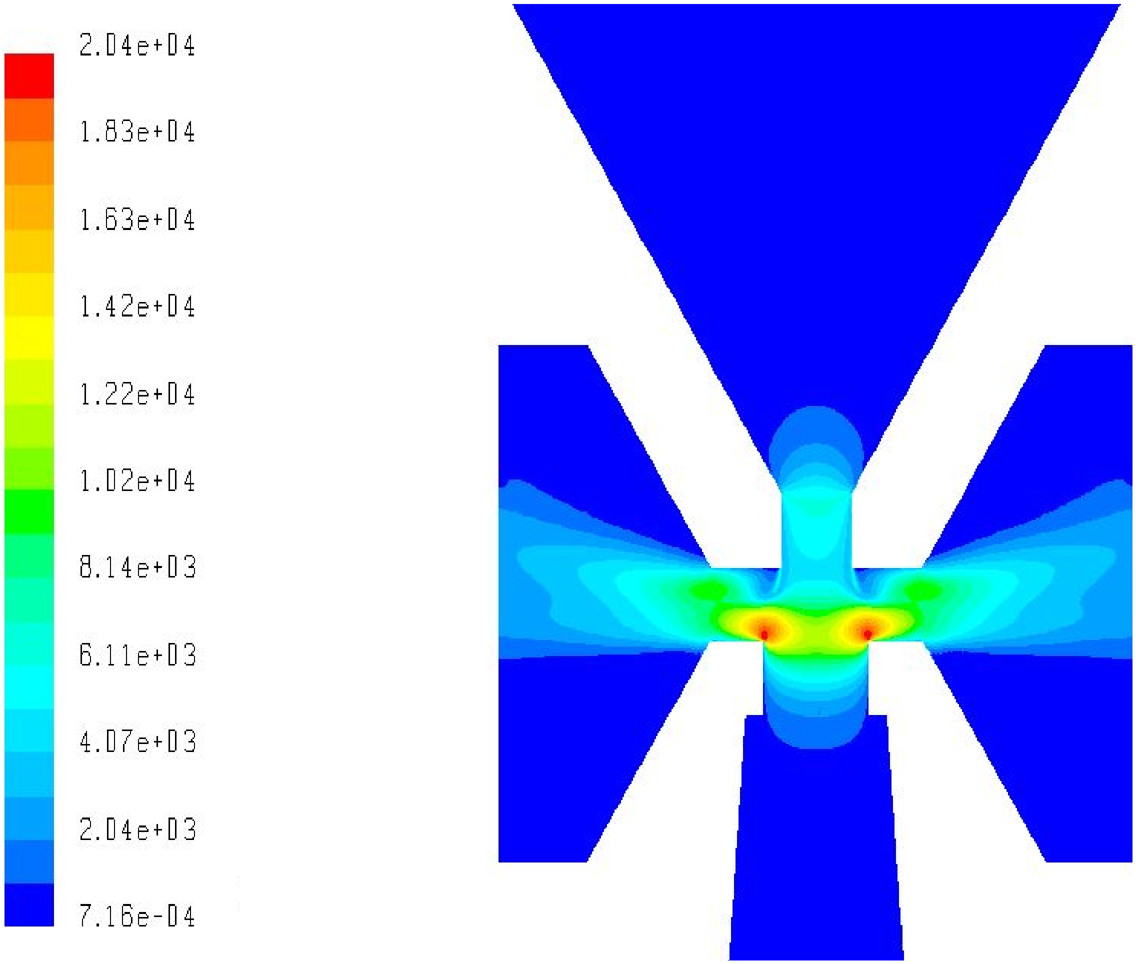

Figure 6.26: Midplane contours of turbulent kinetic energy (m^2/s^2), simulation 5

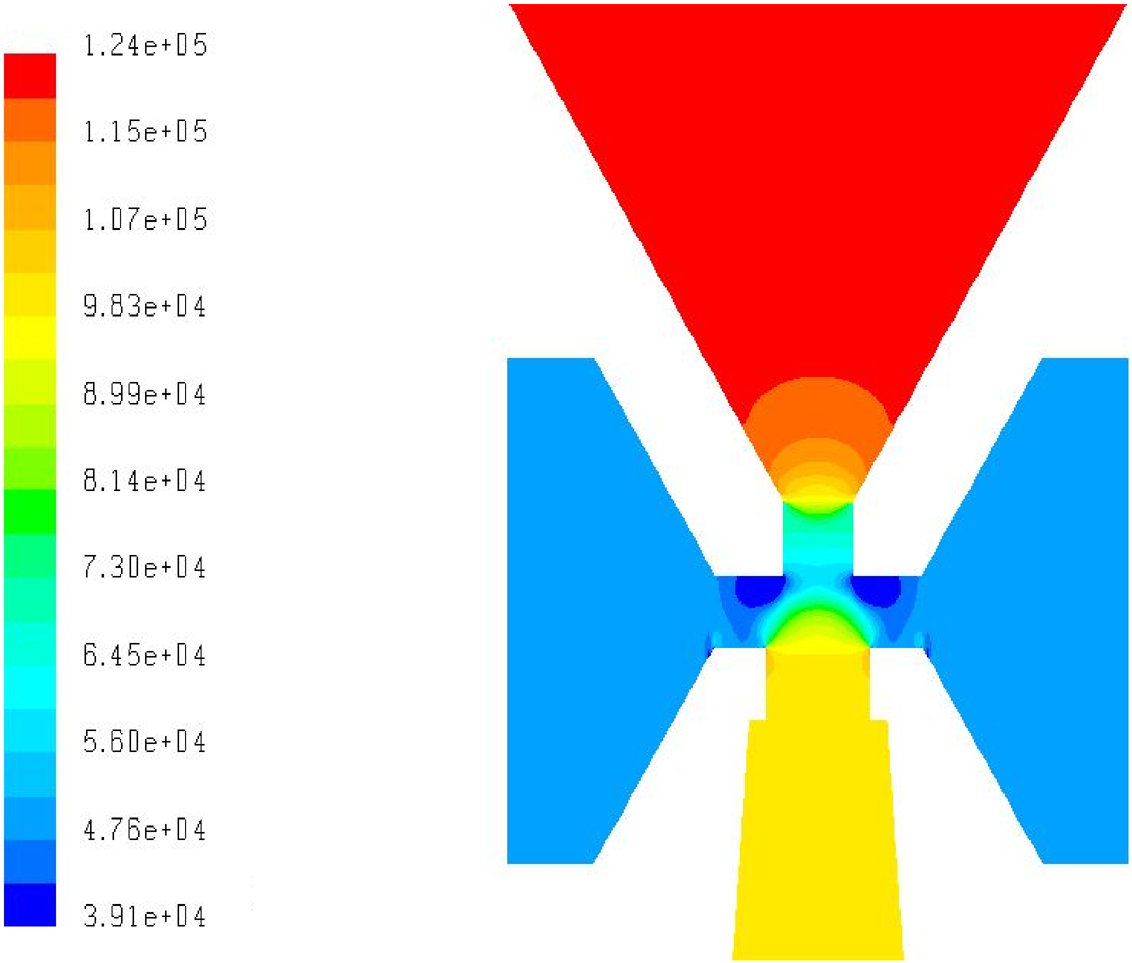

Figure 6.27: Midplane contours of absolute pressure (Pa), simulation 5

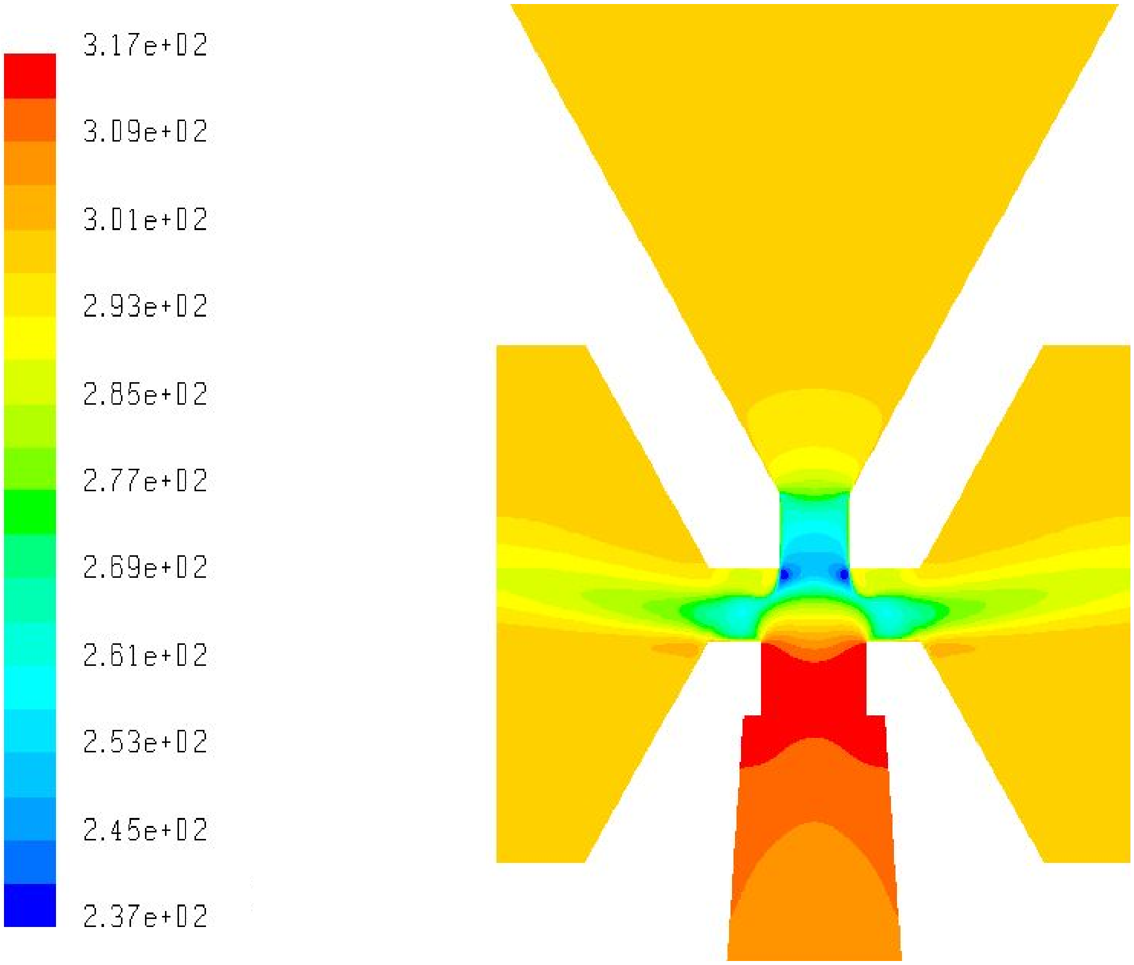

Figure 6.28: Midplane contours of temperature (°K), simulation 5

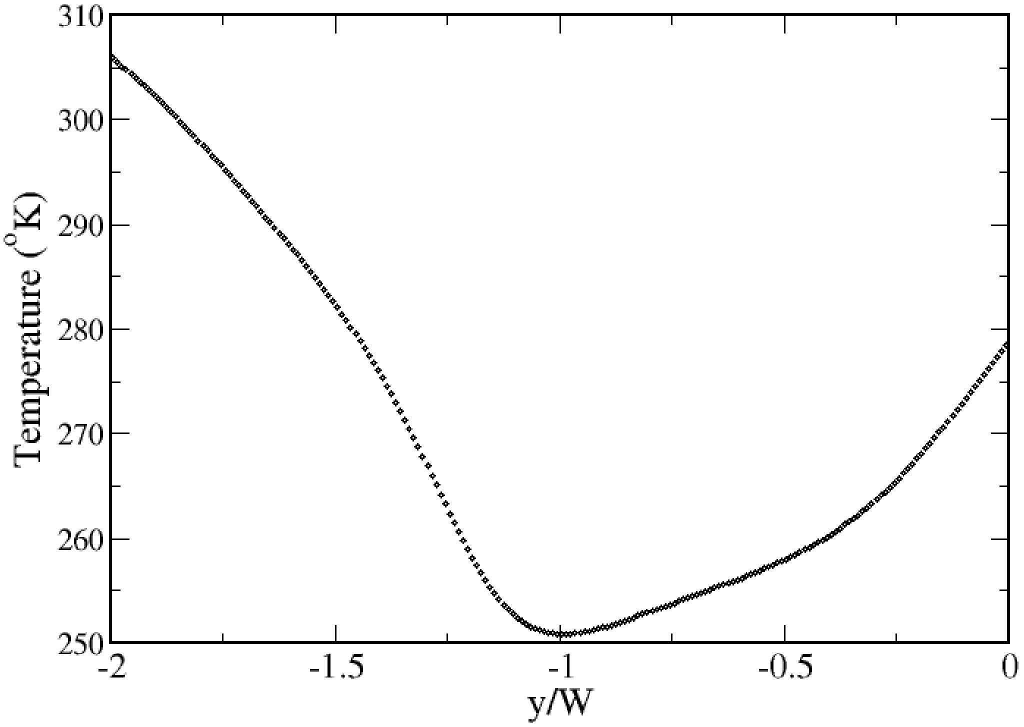

Figure 6.29: Temperature as a function of streamwise distance from nozzle at x=0, simulation 5

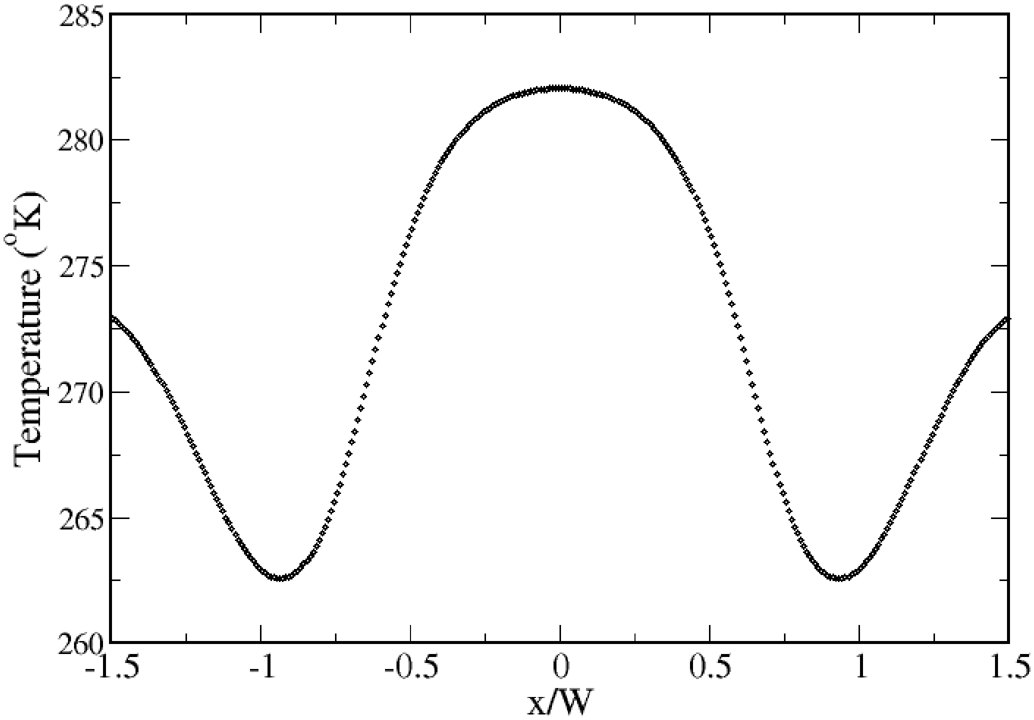

Figure 6.30: Temperature as a function of cross-stream distance, midpoint of virtual impaction zone, simulation 5

The following set of plots is that of the difference between the predictions of a given quantity from the two simulations. First, the norm of the velocity difference between the 1st and 2nd order results, defined as:

$$|\overline{U}(1^{st})-\overline{U}(2^{nd})|=\sqrt{\sum_{i=1}^{3}[\overline{U}_i(1^{st})-\overline{U}_i(2^{nd})]^2}, \tag{6.2}$$

is shown in Figure 6.31 for the virtual impaction zone (Figure 5.2). The contour plot remarkably shows the exact region where the 2nd order discretization solution gives different velocity values than the 1st order discretization solution. The curved streamline region as the fluid exits the nozzle is clearly where the two solutions conflict. Judging by the magnitude of the highest contour, however, the effect of the difference on the total velocity is small. The difference contour plots of absolute pressure, temperature, and turbulent kinetic energy are shown in Figures 6.32, 6.33, and 6.34, respectively. In this case, the contour values are scaled by the actual value of the variable using simulation 4 results. Figure 6.32 exhibits discrepancies in regions other than the curved outflow region, which suggests that the pressure field is influenced by the discretization scheme more than the mean velocity field, which is known to be true in compressible flows.

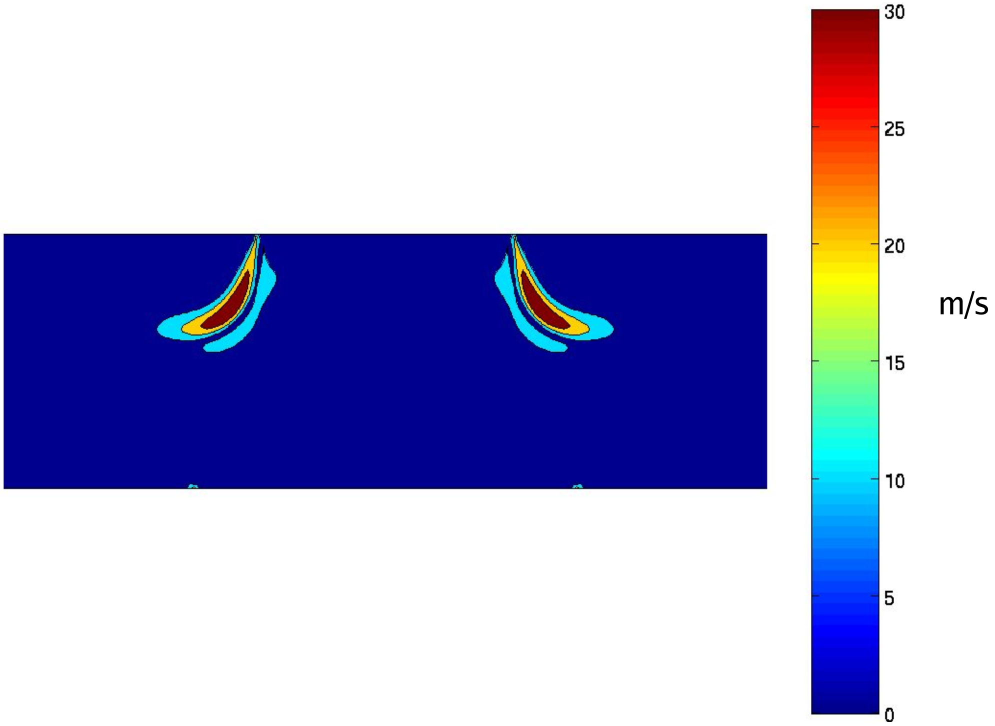

Figure 6.31: Contours of the norm of velocity difference between the predictions of simulations 4 and 5

The temperature difference contour plot of Figure 6.33, exhibits low contour values, indicating minimal offsets between the two solutions. Finally, the turbulent kinetic energy difference contour plot in Figure 6.34 is reminiscent of that of the mean velocity, showing high contour values in the curved streamline area, thus implying the close connection between the computation of mean velocity and turbulent kinetic energy.

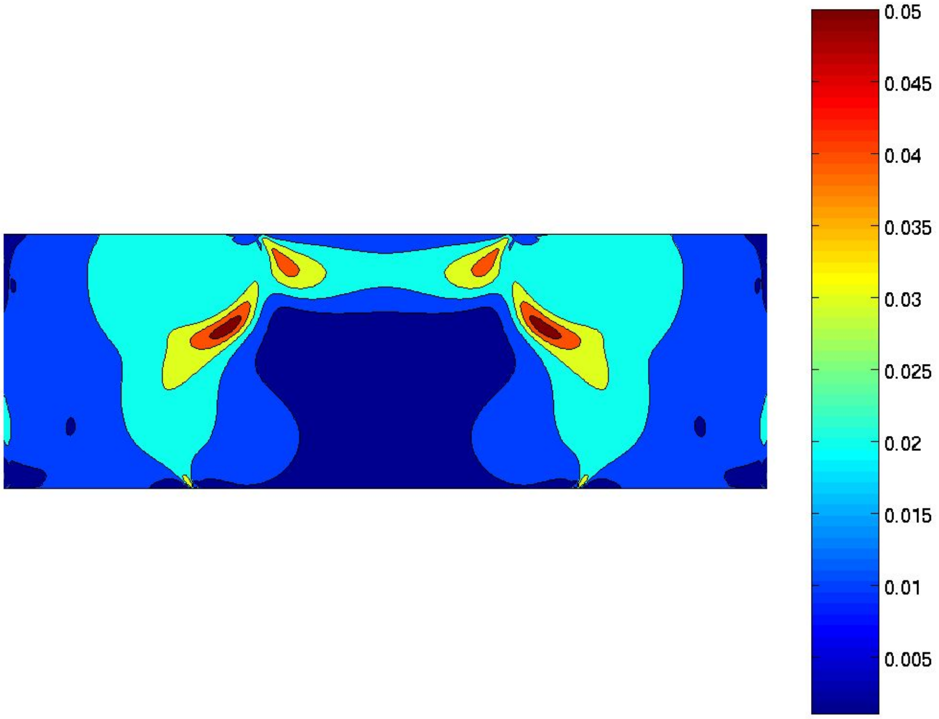

Figure 6.32: Contours of absolute pressure difference between the predictions of simulations 4 and 5, scaled by P_{abs} of simulation 4

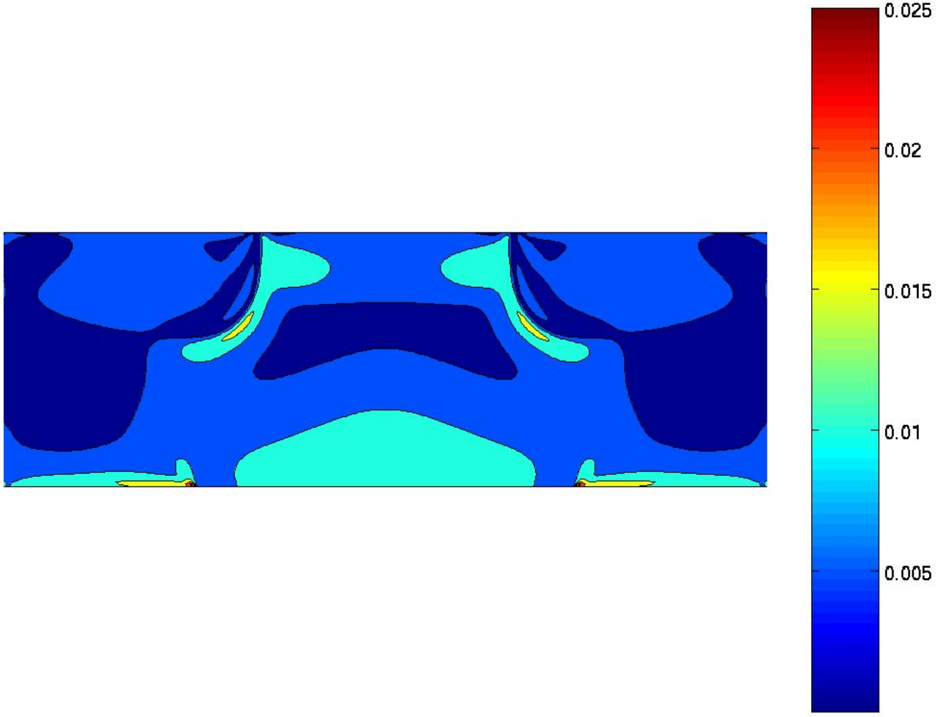

Figure 6.33: Contours of temperature difference between the predictions of simulations 4 and 5, scaled by T of simulation 4

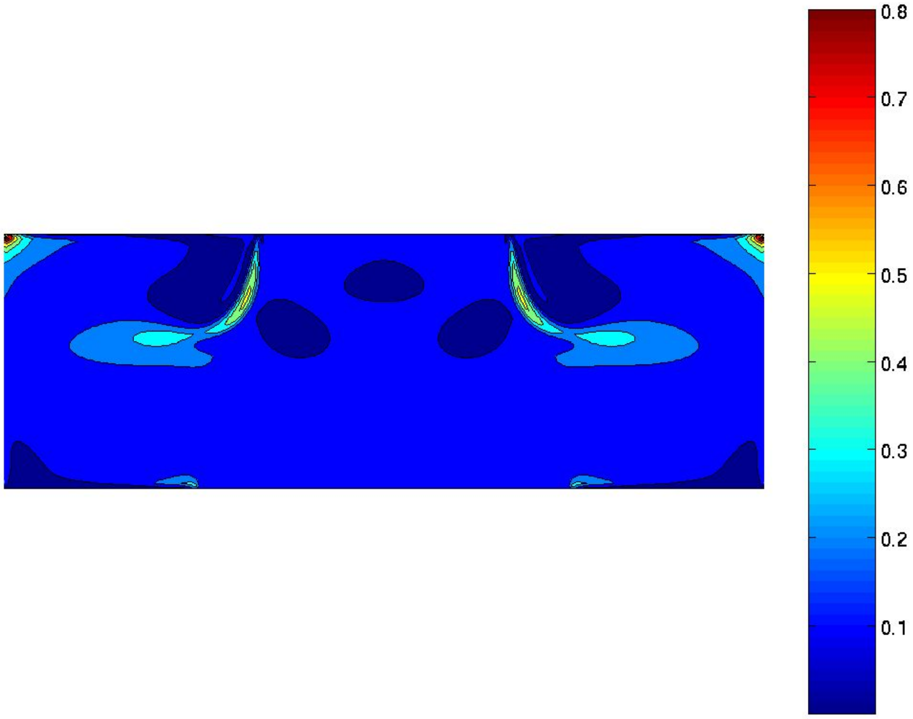

Figure 6.34: Contours of turbulent kinetic energy difference between the predictions of simulations 4 and 5, scaled by K of simulation 4

The impact of the changes in the flow solution on the behavior of the fluid motion may be hard to assess at this point just by looking at the difference contour plots. Nonetheless, it is evident that at least the mean velocity and pressure field are altered by the higher order discretization scheme. The discrete phase results, in Section 6.2, will reveal the impact of those changes on the performance of the device.

6.1.5 Motion of Fluid Particles

In this section we present an overview of the effect that the different simulation conditions have on the motion of fluid elements. AeroTrack is used to track massless fluid particles released at the exit of the accelerating nozzle of the device. The trajectories of fluid points are simply computed by integrating the mean velocity field using the following differential equation:

$$\frac{d\vec{X}_{fp}}{dt}=\vec{U} \qquad (6.3)$$

where X_{fp} is the position vector of a fluid element. The numerical scheme is very much similar to the one used for tracking discrete particles, and is explained in Chapter 4. Figures 6.35-6.37 show the pathlines of 50 fluid points released from a horizontal line spanning half the nozzle width, using the flow field information of simulations 1, 2 and 5, respectively (Table 5.1). It is clear that the Reynolds stress model predicts considerably different flow patterns than the k-epsilon model, primarily in the initial stages of separation, as was seen in Section 6.1.1; and in the expansion into the major flow as evident from the following pathline figures.

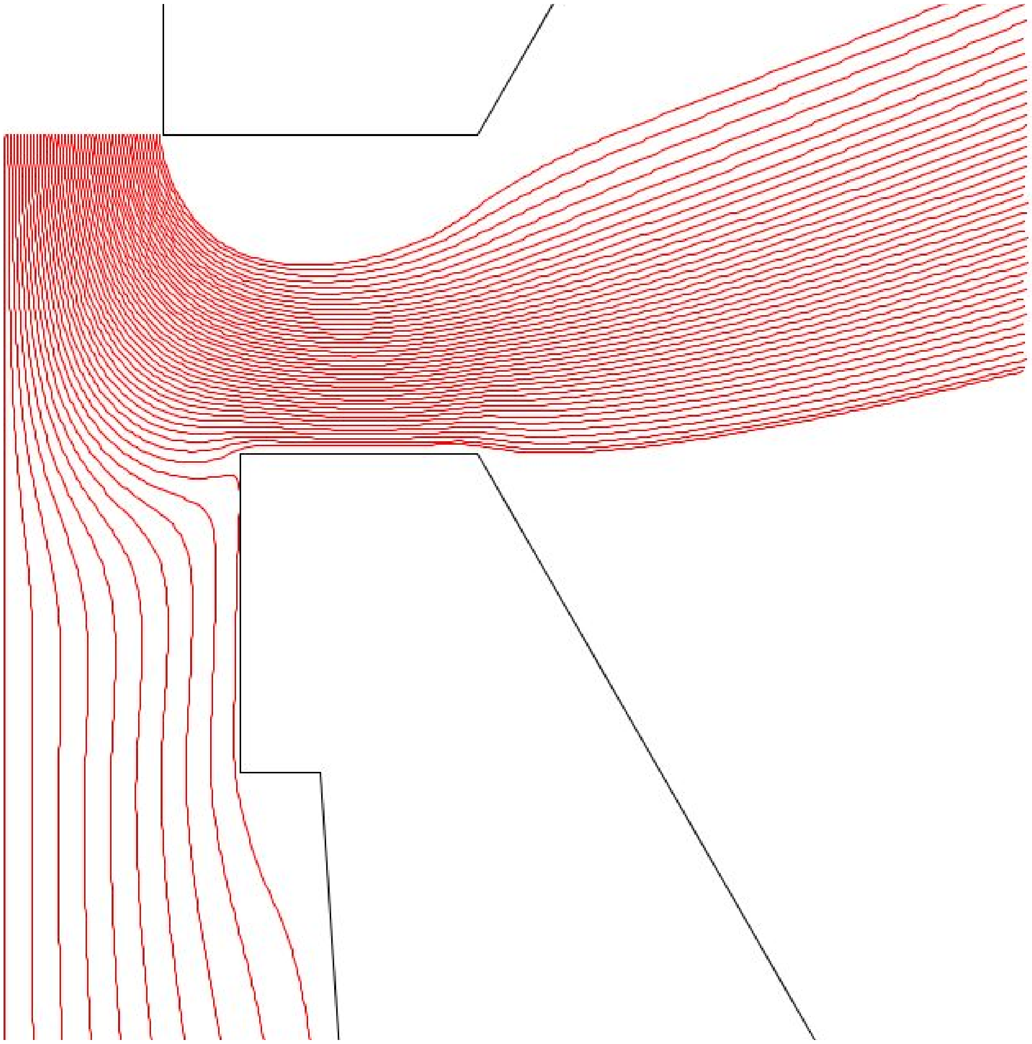

Figure 6.35: Fluid pathlines for simulation 1, incompressible κ-ϵ

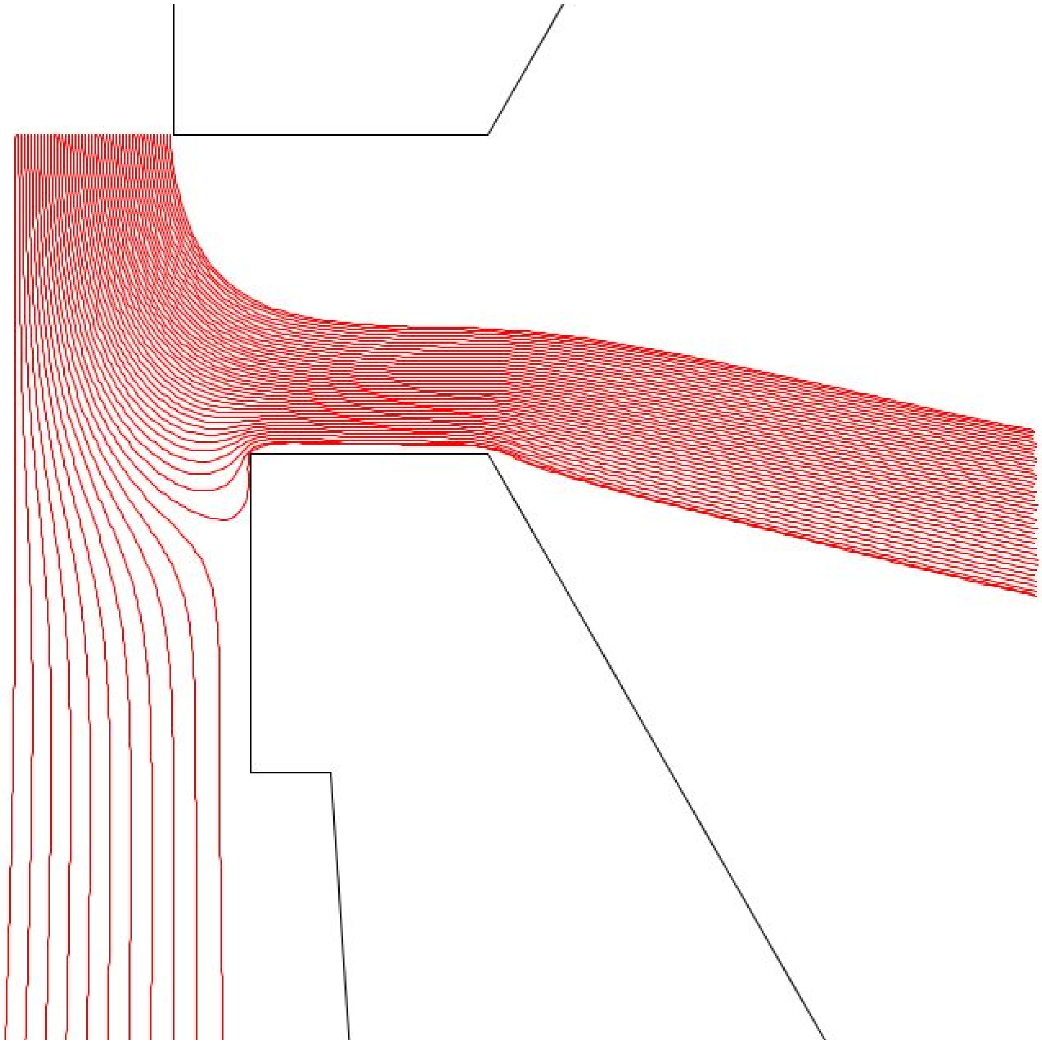

Figure 6.36: Fluid pathlines for simulation 2, incompressible RSM

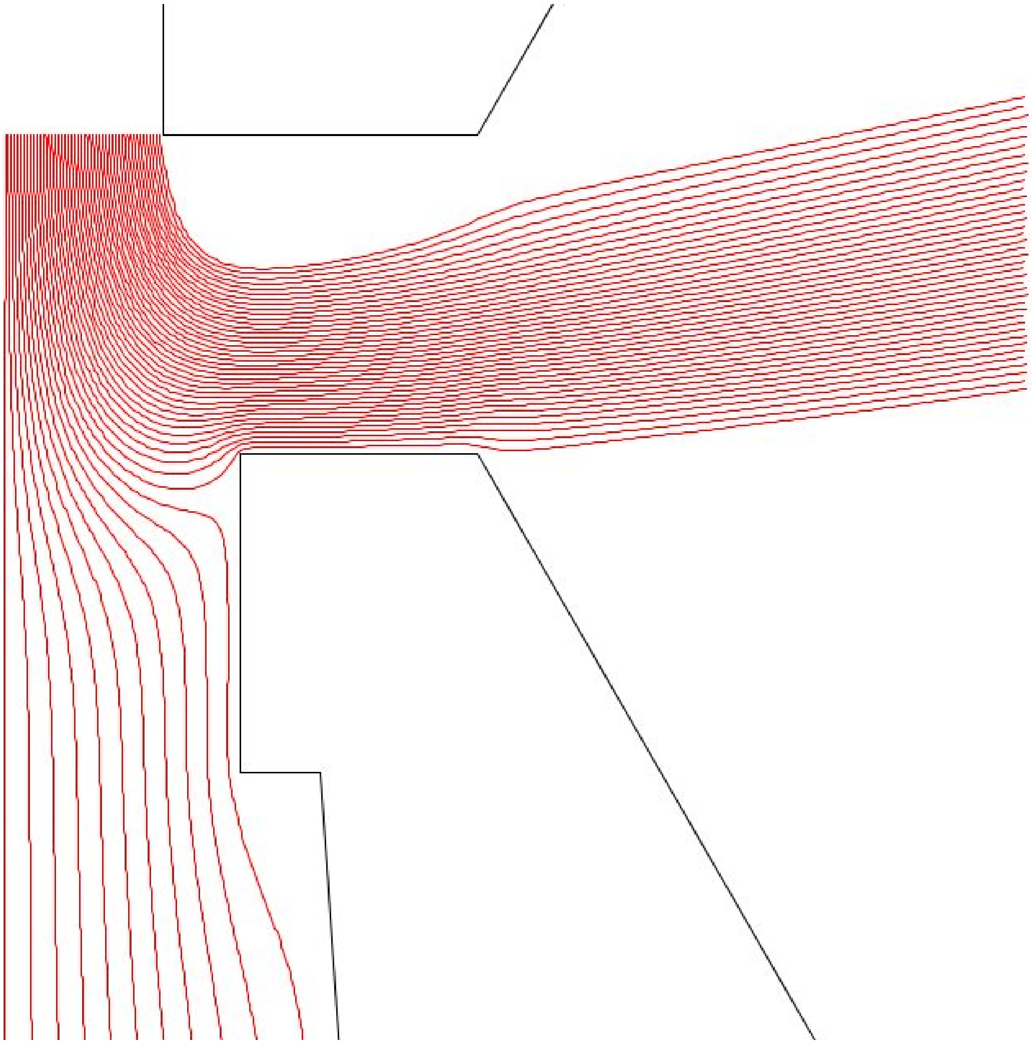

Figure 6.37: Fluid pathlines for simulation 5, compressible κ-ϵ

To capture the quantitative aspects of the flow patterns, the residence time, or the time taken by a fluid point to completely exit the device (either into the major or minor flow) is recorded for each of the above simulations. Figure 6.38 shows the residence time of each fluid particle divided by the mean nozzle time scale, $\tau_{nozzle} = 0.5W/U_{avg}$, versus the dimensionless starting position of the particle.

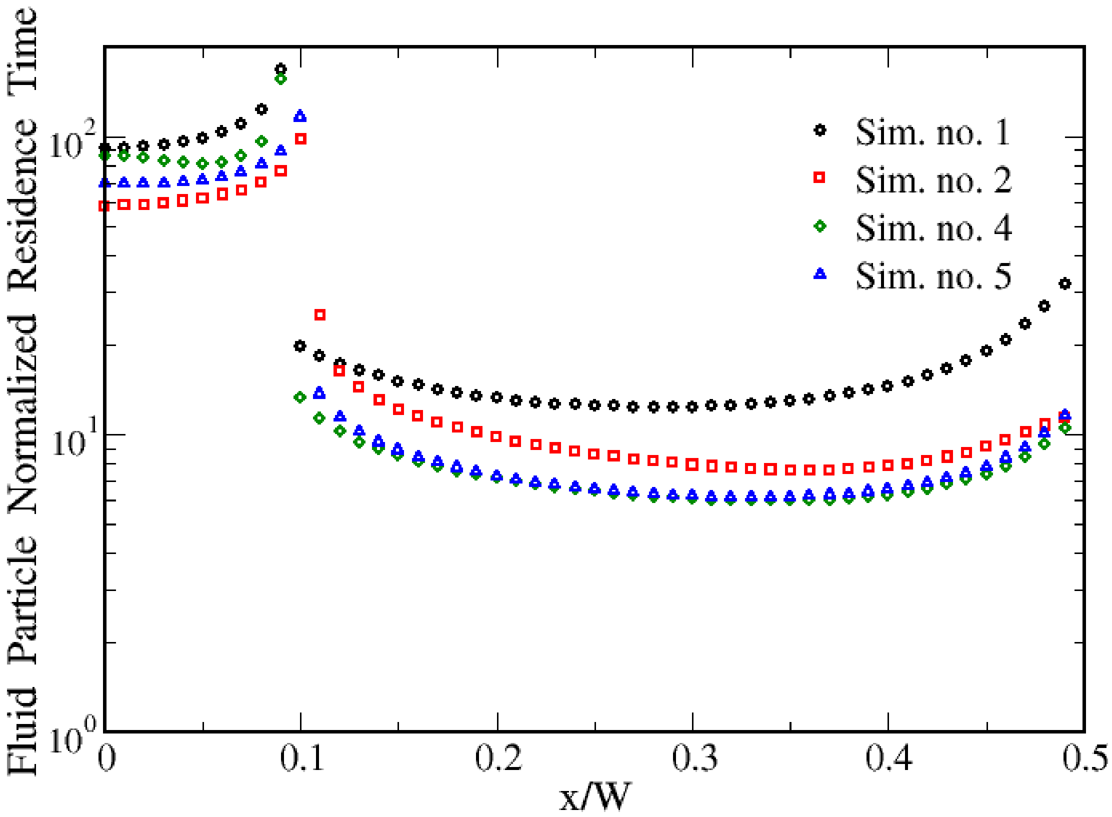

Figure 6.38: Normalized residence time for nozzle fluid particles

The plot is instrumental in revealing many key features of the flow predicted by each simulation. First, the plot decisively shows the cut-off distance from the centerline of the device, beyond which the fluid is bound to diverge into the major flow. Evidently, fluid particles closer to the center are destined to exit into the minor flow, at a much slower pace, thus taking the longest time. Inversely, fluid particles further away from the centerline or closer to the nozzle wall, travel with much higher velocities into the major flow, thus reducing their residence time. It is interesting to note that simulations 2 and 5 slightly shift the cutoff location to about 10% of the nozzle width, whereas simulation 1 and 4 predict a shorter cutoff. This is naturally caused by the higher nozzle velocities achieved by the former two simulations (2 and 5), as we saw in the profile plots in the previous sections. The plot is also indicative of the relative velocity scales predicted by each simulation. Surprisingly, the incompressible Reynolds stress model, predicts the highest velocities in the minor flow, judging by the data points before the cutoff. For the major flow fluid particles, it predicts comparable residence times to the ones predicted by the compressible simulations. The plot also reassures that simulations 4 and 5 predict the same velocity field away from the nozzle.

6.1.6 Vorticity

A brief comparison of the vorticity predictions is presented in this section. The discussion is limited to the data from simulations 2 and 5. The aim is simply to provide a qualitative perception of the capacity of two turbulence models with distinct levels of sophistication, in predicting vorticity. A quantitative comparison is not possible, since the two simulations are computed on different meshes, and using different discretization schemes. Figures 6.39 and 6.40 show the magnitude of vorticity contours in the midplane of the device, for simulation 2 and 5, respectively. The two figures are drawn with the same number of contour levels and ranges, to facilitate the comparison. It appears that the two models agree with respect to the level of vorticity generation at the bottom wall of the virtual impaction zone. They both appear to capture relatively high vorticity in the curved flow stream leaving the nozzle. The k-epsilon model, however, predicts magnitudes that are in excess of those shown on the RSM plot, in addition to the discrepancy near the top wall. This is consistent with the location of the large vortex seen in the velocity vector plots of the k-epsilon model.

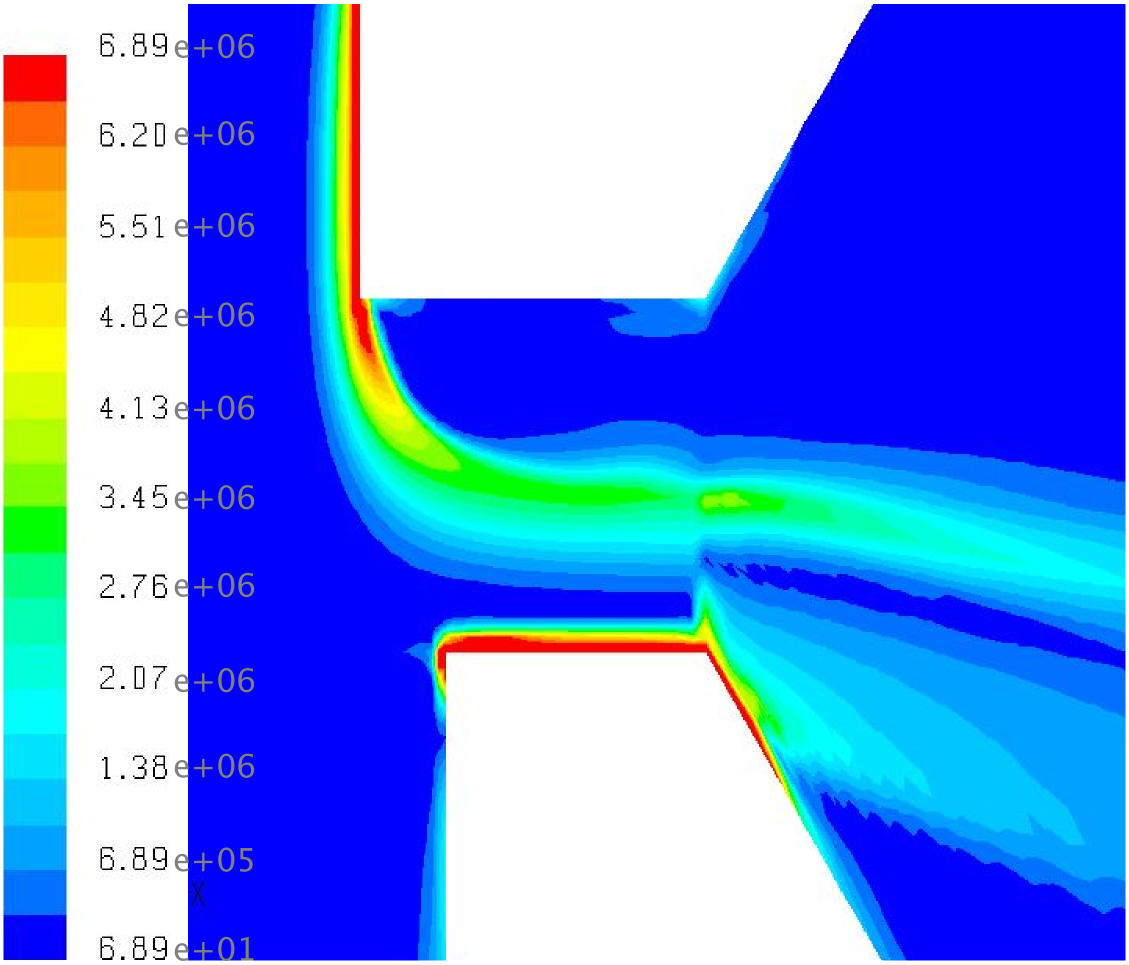

Figure 6.39: Contours of vorticity magnitude (1/s), midplane, simulation 2, incompressible RSM

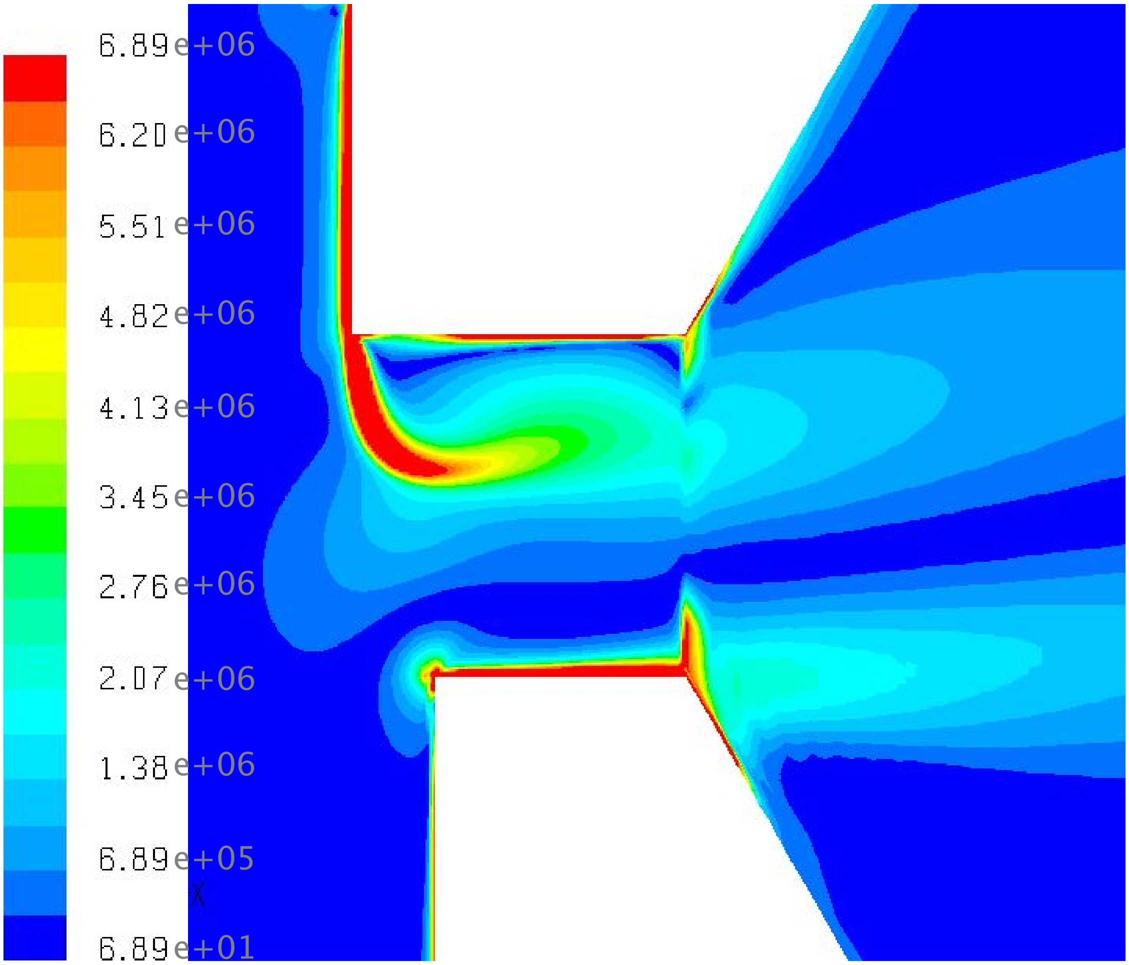

Figure 6.40: Contours of vorticity magnitude (1/s), midplane, simulation 5, compressible K-ϵ

6.1.7 Summary

The highlights of the preceding sections of Chapter 6 are the comparisons among the numerical results of the different simulations of the virtual impactor. The primary parameters or models that constitute the distinct simulations are (see Table 5.1): (1) turbulence model, (2) compressibility, (3) grid resolution, and (4) discretization scheme.

We have found that for an incompressible flow simulation, the κ-ϵ turbulence model, when compared to the Reynolds stress model, predicts quite different re-circulation and expansion patterns in the outflow region of the virtual impaction zone.

The compressible flow simulation was only conducted using the κ-ϵ model, which when compared to the incompressible flow results of the same model, revealed similar separation phenomena but at a much higher velocity scale. The mean velocity and pressure differences between the two types of simulations are both 60% higher than the incompressible flow κ-ϵ predictions.

The analysis of the results on the finer gird, showed that for the same set of boundary conditions, the quality of the computational grid influences the value of the flow separation ratio Q_m/Q_T, however, it does not significantly alter the behavior of velocity or pressure profiles.

Lastly, we have found that the second order discretization scheme is critically essential for accurate mean velocity and turbulent kinetic energy predictions in the curvature region of the flow, and generally important for pressure and temperature everywhere.

6.2 The Discrete Phase Results

The motion of discrete particles, or particles with finite mass, is analyzed in this section. As mentioned before, the particle diameter range of interest is between 0.1 and 0.4 μm. The goal of computing the trajectories of finite size particles is to assess the performance of the device in collecting or separating particles with a given diameter. Several key parameters predispose the accuracy of the particle tracking calculations. We already considered some of these influencing parameters, namely the drag coefficient, and the numerical algorithms. Needless to say, the underlying fluid velocity field plays the biggest role. In this section, a detailed discussion of the key parameters that determine the motion of particles is presented. The collection efficiency and wall loss curves are the primary means of analyzing the results. Secondary, is the analysis of individual particle paths, and the monitoring of flow variables, such as slip velocity and residence time.

The influence of the drag law models presented in Section 4.2.2 will also be discussed in this section. We have seen that the drag coefficient in the particle equation of motion can take several forms, which are essentially dependent on the governing flow dynamics. Table 6.1

summarizes the different drag coefficient models that will be considered in the analysis of the discrete phase results.

Drag Law #	Name	Formula
1	Incompressible or Nonlinear	Equation (4.2.10)
2	Stokes-Cunningham with constant C_c	Equation (4.2.6)
3	Stokes-Cunningham with pressure dependent C_c	Equation (4.2.6)+(4.2.8)
4	Compressible	Equation (4.2.9)

Table 6.1: Summary of drag coefficient models

6.2.1 Incompressible Flow Field

The performance of the device is first classified within the context of the incompressible flow field predictions. This will allow us to assess the impact that the two turbulence models of simulation 1 and 2 (see Table 5.1) have on the mean particle motion, as well as provide insight into the effect of turbulent particle dispersion. The calculations are carried out using FLUENT^{TM} under the guidelines established in Sections 5.2.1 and 5.2.2. Figures 6.41-6.42 are the outcome of a study to determine the appropriate drag coefficient for the incompressible flow field. The particle trajectories are first computed using the non-linear drag formulation (drag law 1), relying solely on the ***mean*** velocity field. The calculation is repeated using the Stokes-Cunningham drag coefficient with a constant slip correction factor C_c (drag law 2), evaluated at entrance pressure conditions.

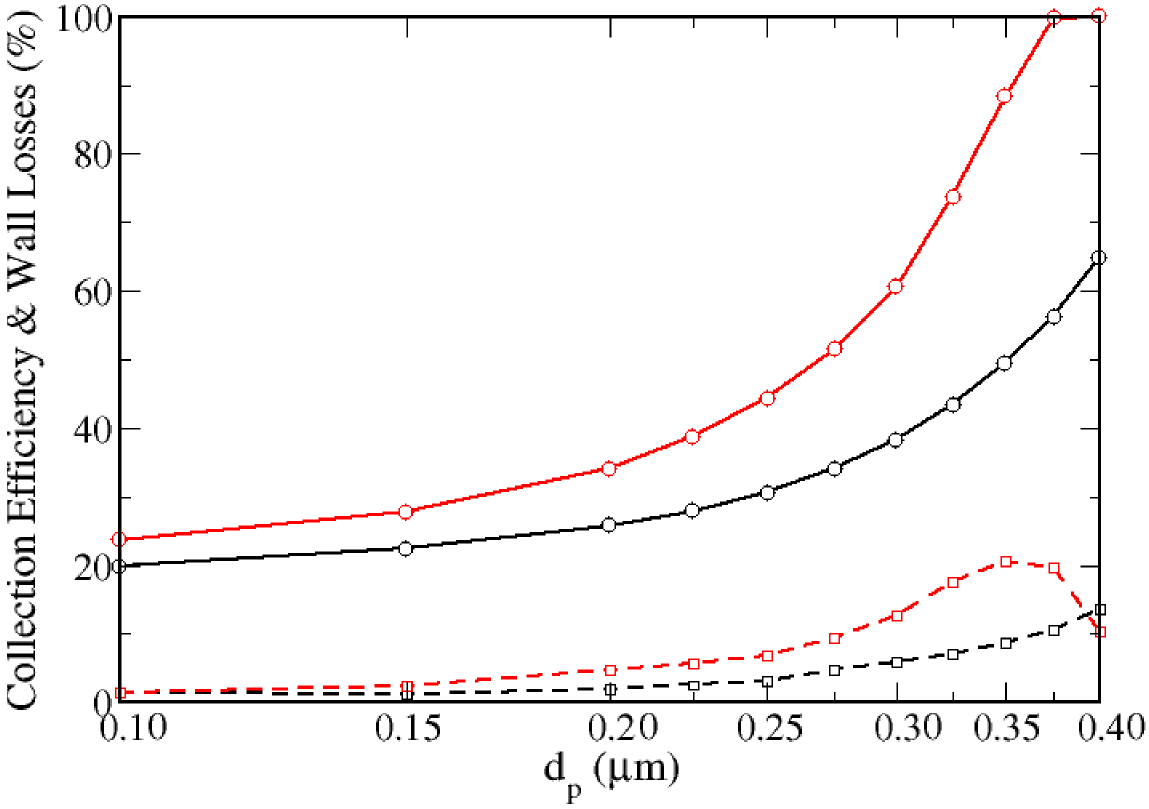

Figure 6.41: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2 (red), drag law 1 (black) – simulation 1 – mean fluid velocity tracking

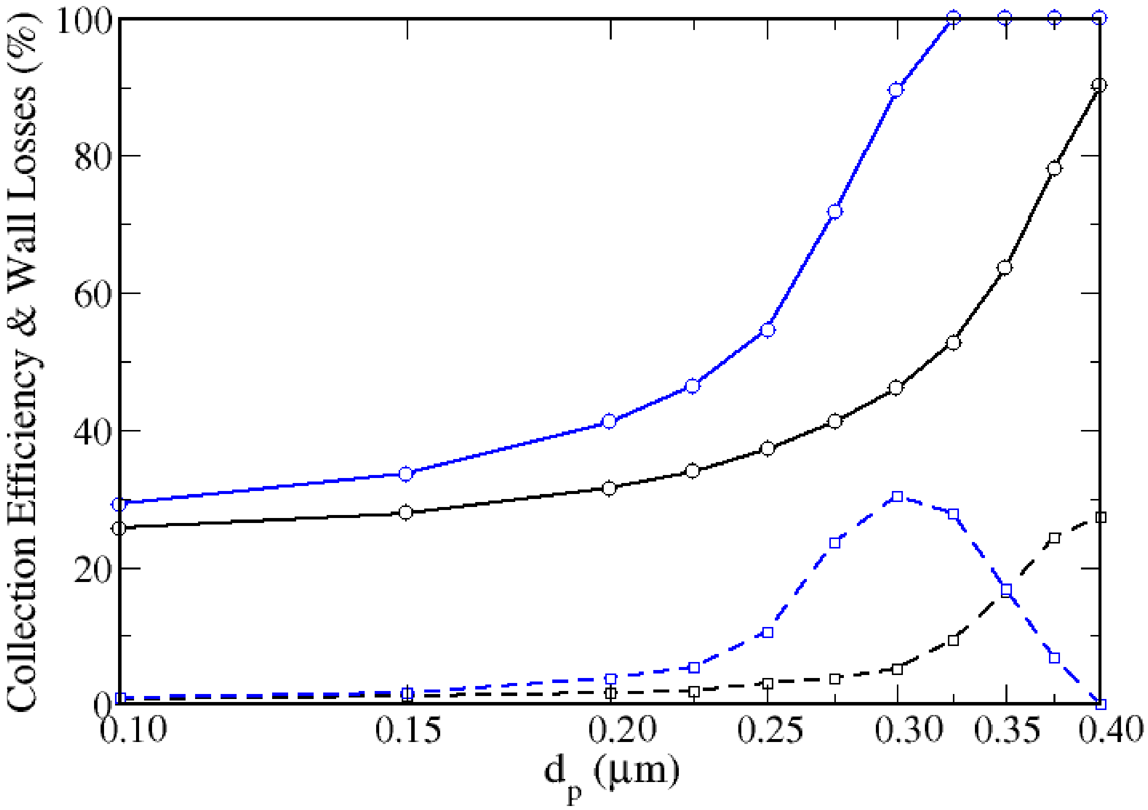

Figure 6.42: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2 (blue), drag law 1 (black) – simulation 2 – mean fluid velocity tracking

It is evident that the nonlinear drag coefficient is not suitable for the particles in the studied range. The efficiency curve is greatly shifted to the right, and the 50% cutpoint diameter is pushed further away from the experimental value ($d_{p\,50}$ = 0.12 μm), which means that more particles are going into the major flow. The reason for this decadence can be drawn from Figures 4.1 and 4.2. The nonlinear drag coefficient overpredicts the drag force on the sub-micron particles, thus forcing them to closely follow the fluid into the major flow, and hence reduce their collection in the minor flow.

The particle data from simulation 1 and 2 are plotted on the same graph in Figure 6.43, in an effort to evaluate the effect of the mean flow field of each turbulence model on the efficiency curve and wall losses.

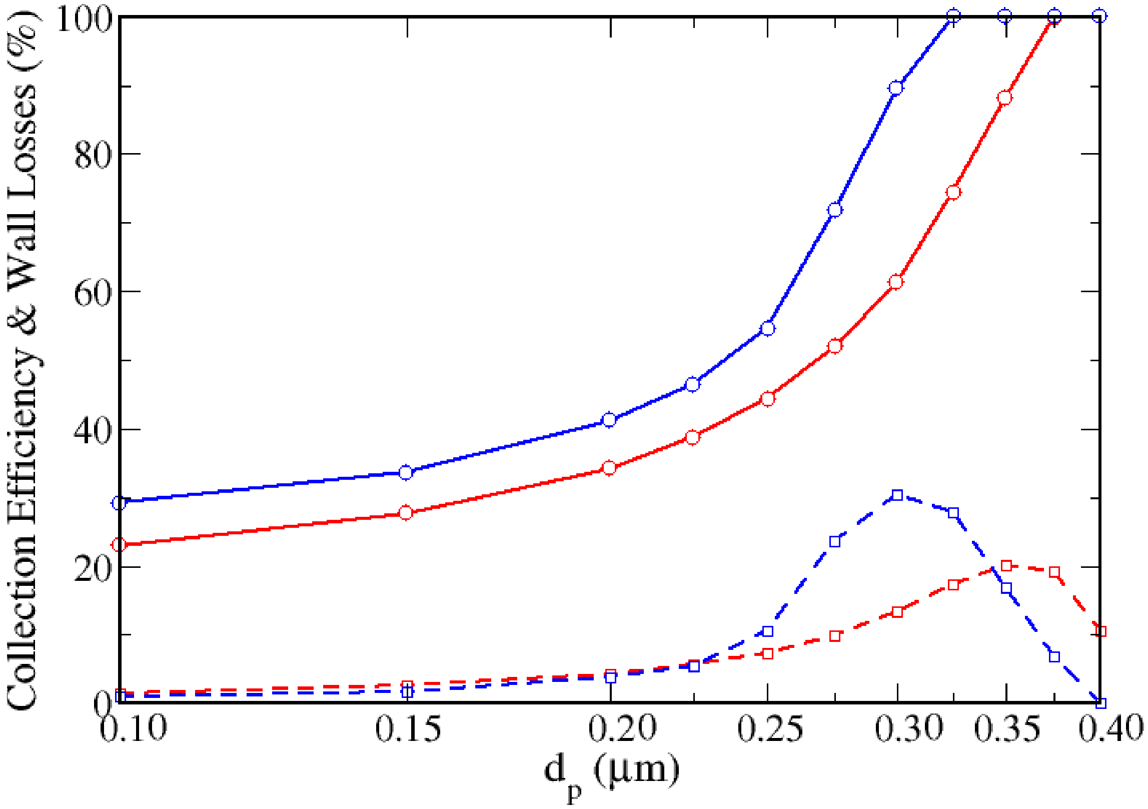

Figure 6.43: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2, simulation 1 (red), simulation 2 (blue) - mean fluid velocity tracking

The efficiency curve naturally benefits from the slightly higher minor-to-total flow ratio in simulation 2 (see Table 5.1), and is consequently shifted to the left of the curve given by simulation 1. It is hard to say at this point, how much of this increase in collection efficiency is due to the Q_m/Q_T ratio, and how much of it is due to the differences in the flow structure predicted by each simulation. This issue will be further addressed later when we consider the data from simulation 5 which has flow features similar to those of simulation 2, particularly for the velocity magnitudes in the virtual impaction zone. The impact on the wall losses is more pronounced since, despite the shift in the location of the peak, the losses are much higher in the Reynolds stress model simulation, reaching a maximum of 30%, as opposed to the 20% peak value given by the $\kappa-\epsilon$ simulation. Recall that the source of wall losses in the virtual impactor is the particles that collect on the inner surfaces of the device. In the current investigation, as mentioned in Section 3.5, we assume that an encounter between a particle and a wall terminates the particle's flight. The question whether this is realistic requires an incorporation of deposition models, and accurate experimental results.

The validity of the stochastic approach to simulate turbulent particle dispersion is queried using the flow field information of simulation 1 and 2. Figures 6.44 and 6.45 show the efficiency and wall loss curves generated by tracking particles using the Random Walk Model (Section 4.2.3) in each of the aforementioned simulations. The curves generated by the mean fluid velocity tracking are also re-drawn for comparison. Undoubtedly, the inadequacy of the stochastic approach in the $\kappa-\epsilon$ model is exposed in Figure 6.44. The stochastic efficiency curve is extremely low compared to the mean velocity curve, and the wall losses are unrealistic, with values much higher than the collection efficiency itself. This fallacy is believed to be due to two basic reasons: (i) the assumption of isotropy is invalid, and (ii) the turbulent kinetic energy predictions are possibly unphysical. Judging by the plot of the Reynolds stress model in Figure 6.45, which depicts reasonable offsets for the stochastic efficiency and losses relative to the mean velocity curve, it appears that more weight should be given to the isotropy assumption, because it is the fundamental difference between the way the two models produce turbulent fluid velocity fluctuations. The issue of judging the turbulent kinetic energy levels, is considered by looking at the contours of the turbulent kinetic energy. A comprehensive contour plot for each of the κ-ϵ and Reynolds stress models is shown in Figures 6.46 and 6.47, respectively. As seen from the figures, the turbulent kinetic energy contours of the κ-ϵ model are significantly more intense than those of the RSM. Figure 6.46 is reminiscent of Figure 6.26 which was shown for the compressible κ-ϵ model, but with lower values for the turbulent kinetic energy due to the lower velocity scales in the incompressible simulation. Apparently, the κ-ϵ model predicts the highest levels of turbulent kinetic energy at the side walls of the inlet to the collection nozzle.

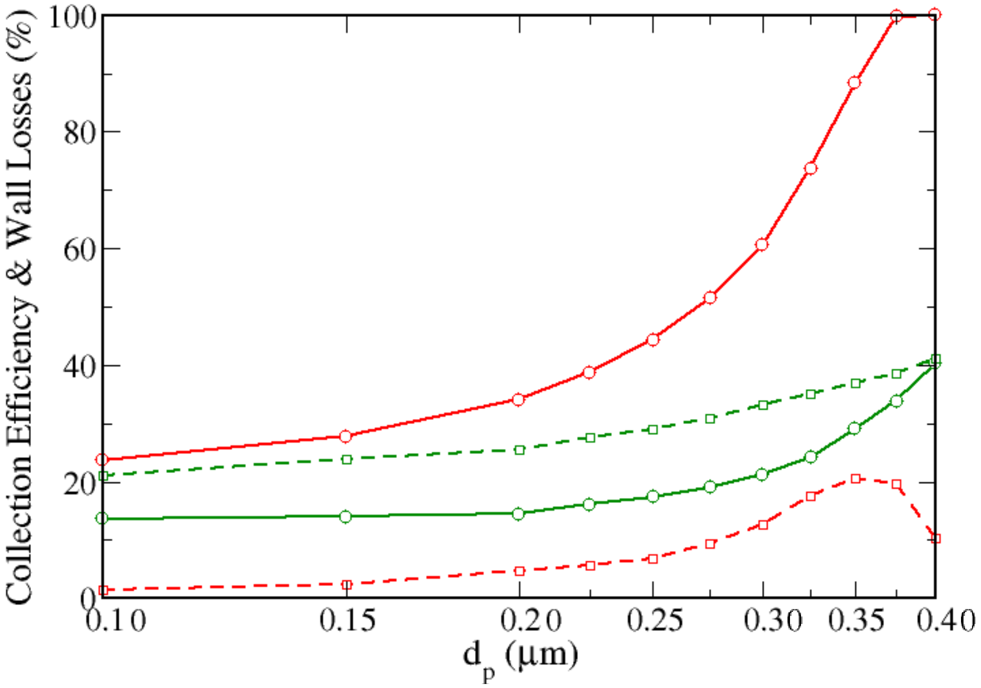

Figure 6.44: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2, mean fluid velocity tracking (red), stochastic tracking (green), simulation 1 – K-ϵ

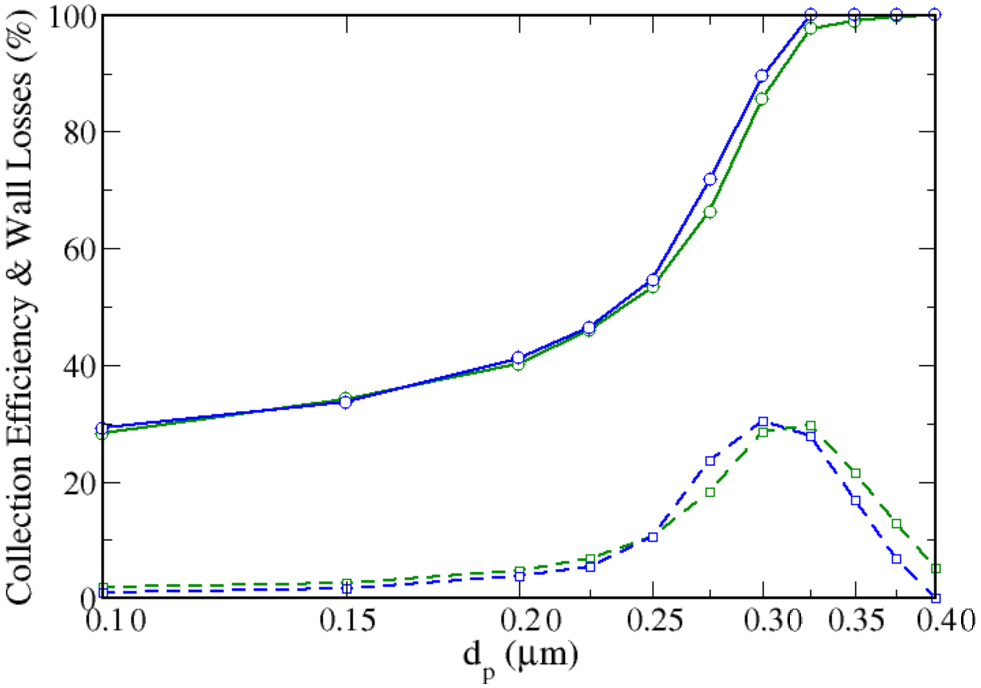

Figure 6.45: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2, mean fluid velocity tracking (blue), stochastic tracking (green), simulation 2 – RSM

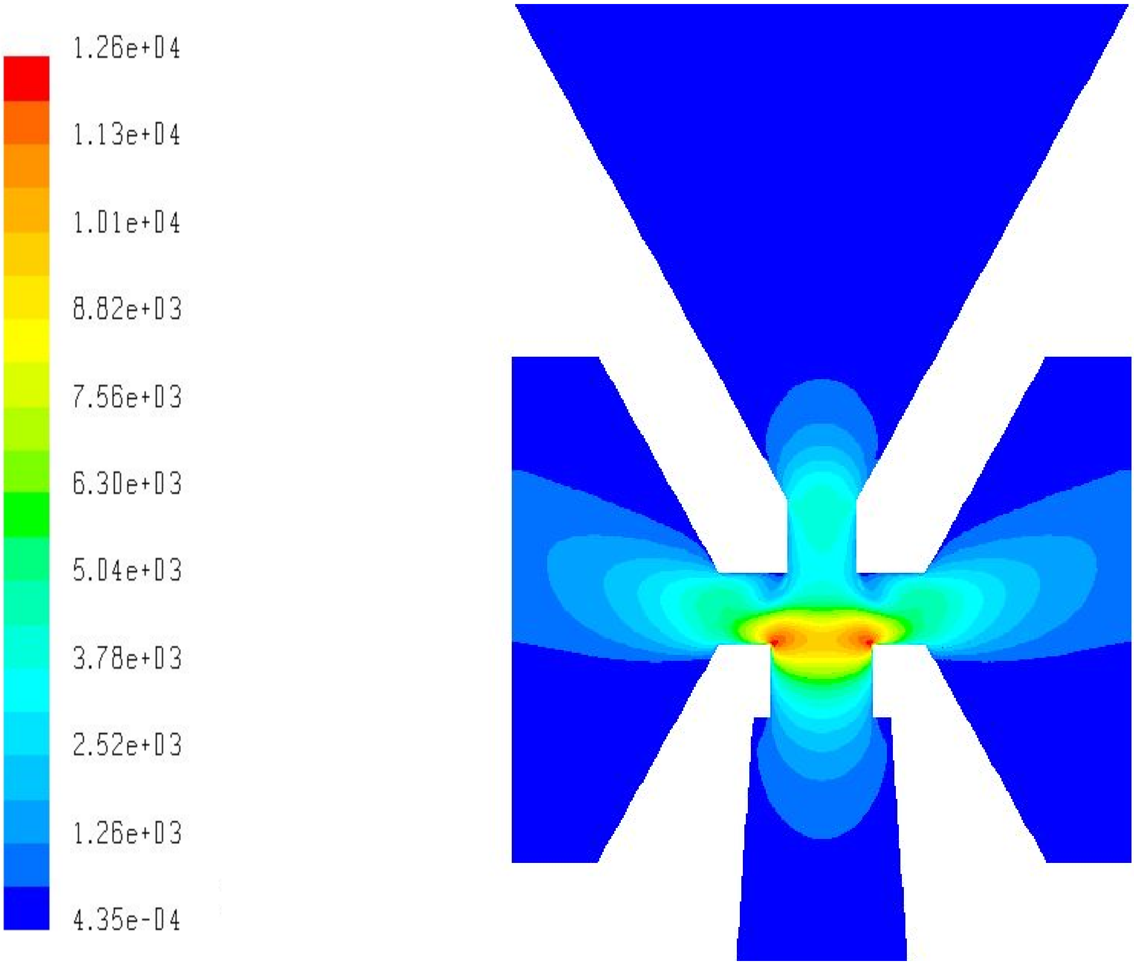

Figure 6.46: Midplane contours of turbulent kinetic energy (m^2/s^2), simulation 1 – incompressible Ƙ–ϵ

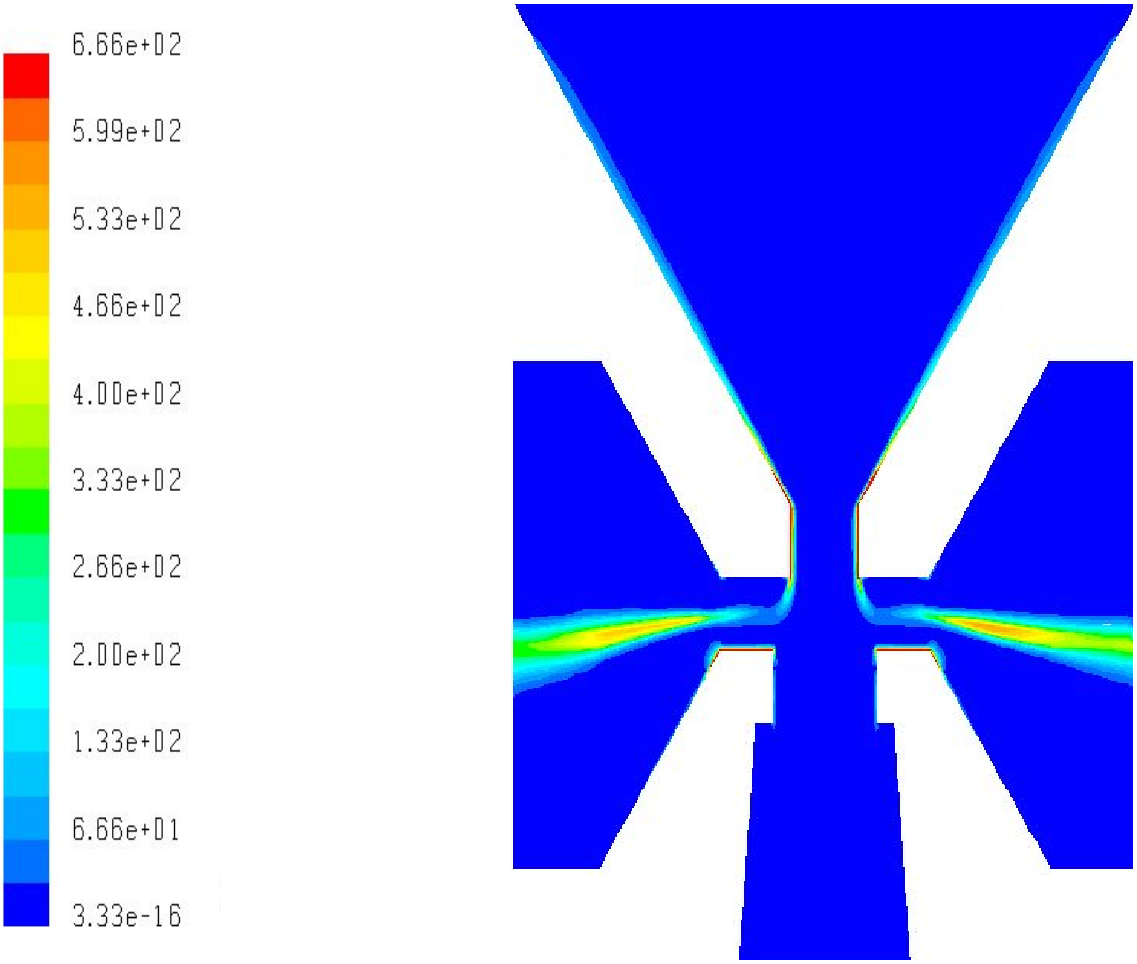

Figure 6.47: Midplane contours of turbulent kinetic energy (m^2/s^2), simulation 2 - incompressible RSM

In addition, a considerable amount of turbulent kinetic energy is present in the accelerating nozzle, and in the virtual impaction zone. A quantitative assessment of the discrepancy in the turbulent kinetic energy predictions of each model is obtained by looking at the difference norm defined below:

$$|\mathrm{TKE}(K\epsilon)-\mathrm{TKE}(\mathrm{RSM})|. \tag{6.4}$$

A difference contour plot is shown in Figure 6.48, for the virtual impaction region (see Figure 5.2). It appears that most of the discrepancy lies near the bottom walls of the nozzle. The K-ϵ model gives turbulent kinetic energy values that are 60% higher than those calculated by the Reynolds stress model. The differences, however, go down away from the bottom wall, to about 20 to 30% as the fluid exits into the major flow. This observation, was also confirmed by monitoring the number of particles that impact the nozzle walls, and it was expectedly higher in the K-ϵ stochastic tracking simulation. Despite the turbulent kinetic energy deviations between the two models, it is still believed that the isotropy assumption is the most decapitating disadvantage of the K-ϵ random walk model. A comparison between the turbulent viscosity contours (not shown here) of each model shows comparable predictions. In fact, the highest turbulent-to-molecular viscosity ratios are of the order 10^2, and are associated with the fluid near the bottom of the cone-inlet as it accelerates into the throat. To fully settle this problem, it would be beneficial to conduct particle tracking calculations in a K-ϵ simulation using a more sophisticated stochastic technique, like the use of Lagrangian correlations with non-isotropic second moments [9], however, this is not currently feasible in FLUENT™, and was not implemented in AeroTrack. Another simpler approach that can slightly improve on the total isotropy assumption of the K-ϵ stochastic model is the use of the turbulent viscosity and the Boussinesq hypothesis of Equation (4.1.9). However, it is not very accurate especially that in a turbulent channel flow where the only mean velocity gradient is normal to the wall, the full equation will still predict an isotropic spectrum of normal stresses, which is known to be unphysical [22].

On the other hand, the results of the stochastic Reynolds stress model of Figure 6.45, despite their agreement with the mean fluid velocity curve, do not necessarily represent the physical effect of turbulence on the efficiency and wall loss curve. In the lack of experimental fluid flow data, it is hard to affirm the turbulent kinetic energy predictions of the RSM. Granted there is some turbulent kinetic energy generation in the flow field of the this model, the effect of turbulent fluctuations is manifested as a slight reduction in the collection efficiency of particles in the higher range of diameters, and a general increase in wall losses.

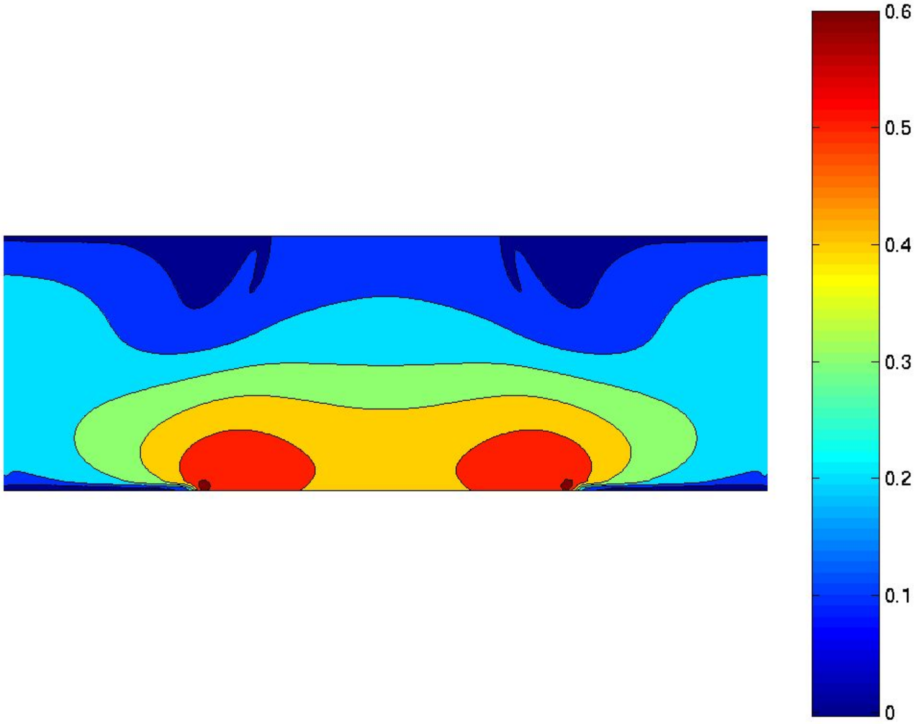

Figure 6.48: Contours of the turbulent kinetic energy difference between the predictions of simulations 1 and 2, scaled by ½ U^2_{avg}

6.2.2 Compressible Flow Field

The effect of the compressible flow field is investigated in this section. For the reasons discussed above, the efficiency and wall loss curves presented hereinafter all rely on particle tracking using the mean fluid velocity. Computations conducted in FLUENT™ as well as AeroTrack will be discussed, and other models for the drag coefficient will be used.

The flow field information from simulation 4 is first used to compare between the particle tracking calculations of FLUENT™ and AeroTrack. The computations are conducted using the same drag law with the same number of particles, starting from the same initial conditions, and advanced in time by the same stepping technique described in Section 5.2.1. Figure 6.49 shows the efficiency and wall losses curve obtained by this study.

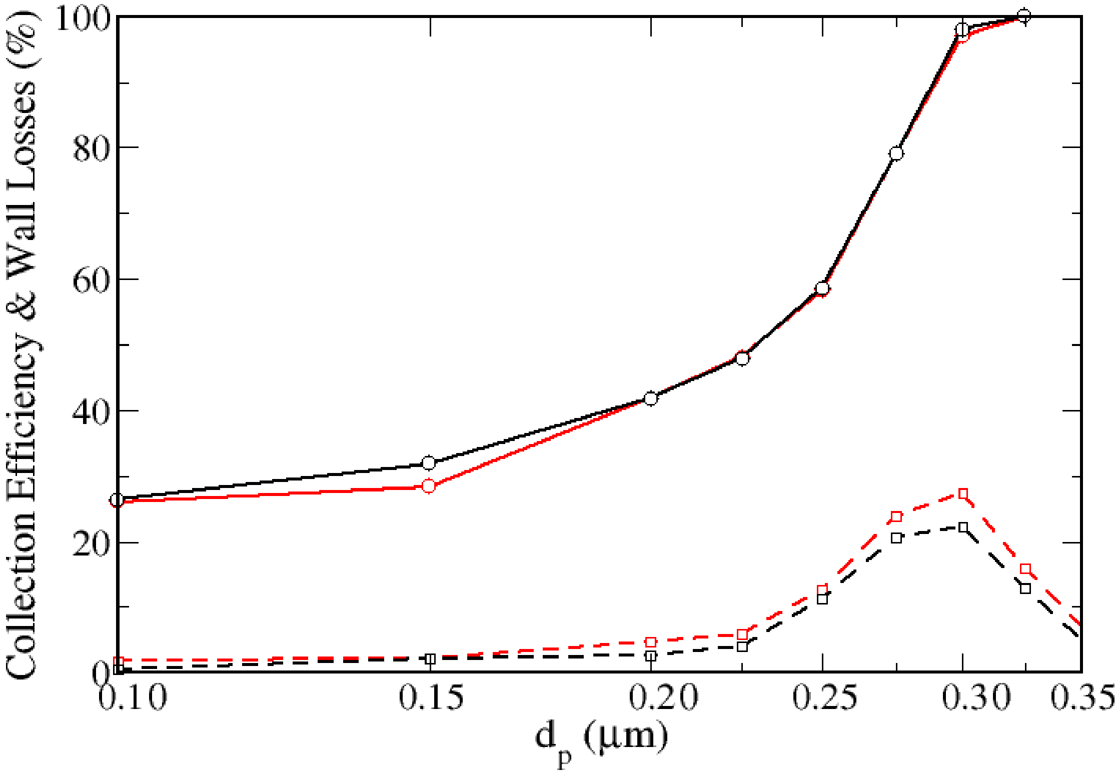

Figure 6.49: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2, AeroTrack (red), FLUENT (black) – simulation 4

The two tracking routines appear to match almost perfectly for this particular case, especially in the outcome of the efficiency curve. The wall losses, however, are in disagreement for most of the larger particle sizes, deviating most at the peak value. To further understand the effect of the particle tracking numerical scheme, a similar calculation is done using the flow field of simulation 5. Figure 6.50 shows the result of such a run.

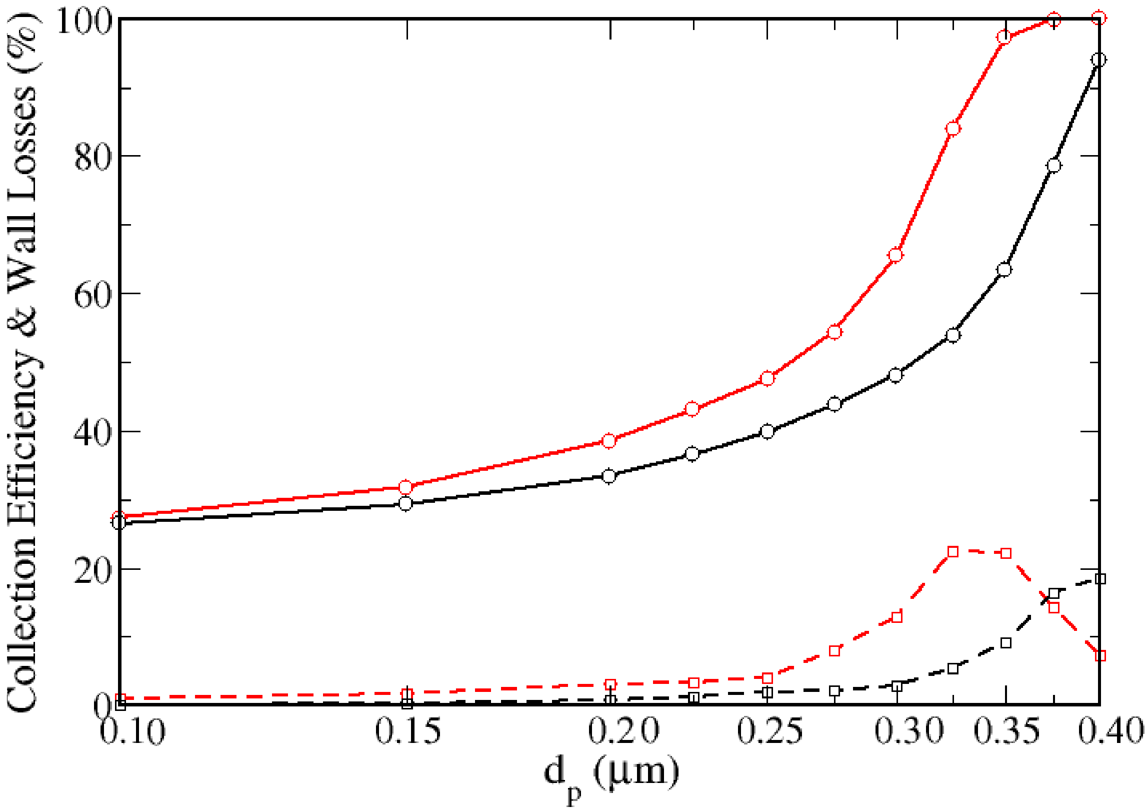

Figure 6.50: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2, AeroTrack (red), FLUENT (black) – simulation 5

The discrepancies in this case are much more pronounced, resulting in a shift in the efficiency curve, as well as some major differences in the wall loss values computed by each program. In addition, the curves computed by either program are not the same between simulation 4 and 5, as illustrated in Figure 6.51, which shows the AeroTrack results.

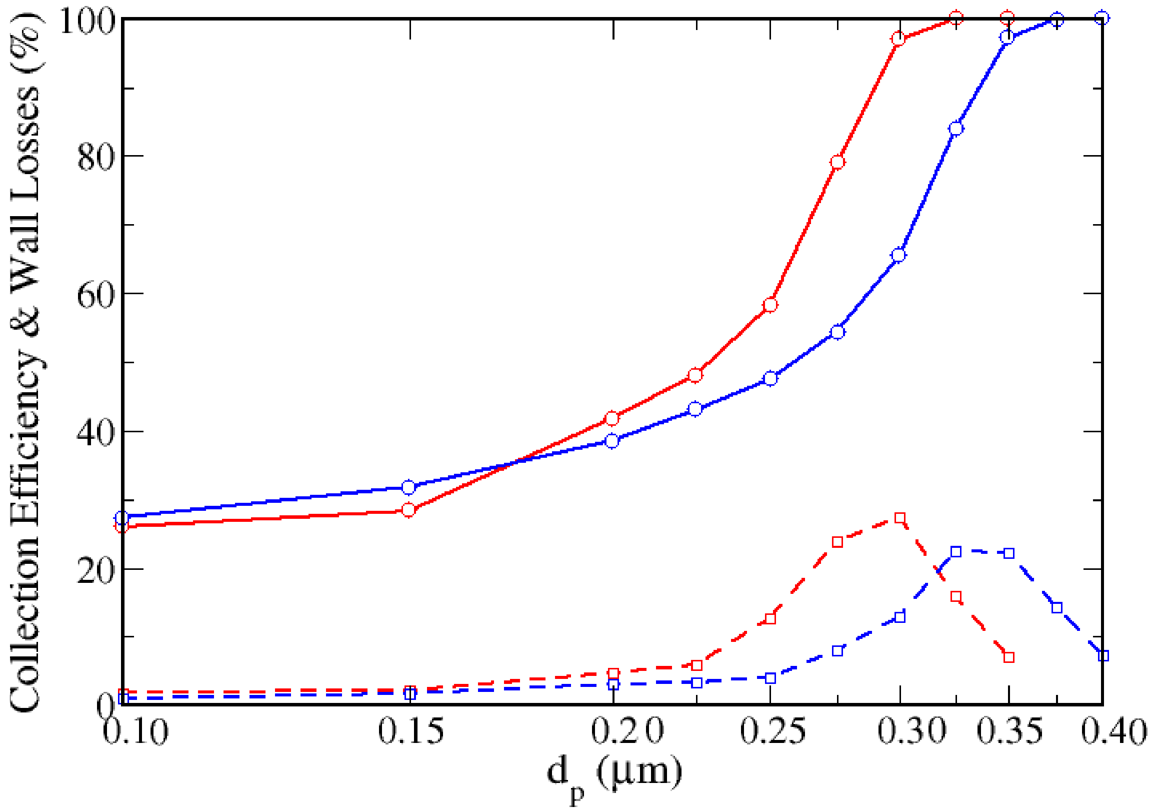

Figure 6.51: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2, AeroTrack, simulation 4 (red), simulation 5 (blue)

Before we consider the causes of the difference between the discrete phase results obtained using the two flow field solutions, an explanation of the differences between the results of FLUENT™ and AeroTrack is sought. We shall focus our attention on simulation 5 data, since it displayed more pronounced dissimilarities. The paths for particles with diameters equal to 0.10, 0.25 and 0.40 μm, computed by each algorithm, are compared in Figures 6.52-6.54, respectively. The particles are released from the same location near the top inlet of the device. The motion of the particles in the cone and throat section of the device is predicted identically by the two algorithms. As the particles exit the acceleration nozzle into the high speed virtual impaction zone, differences are observed in the particle paths.

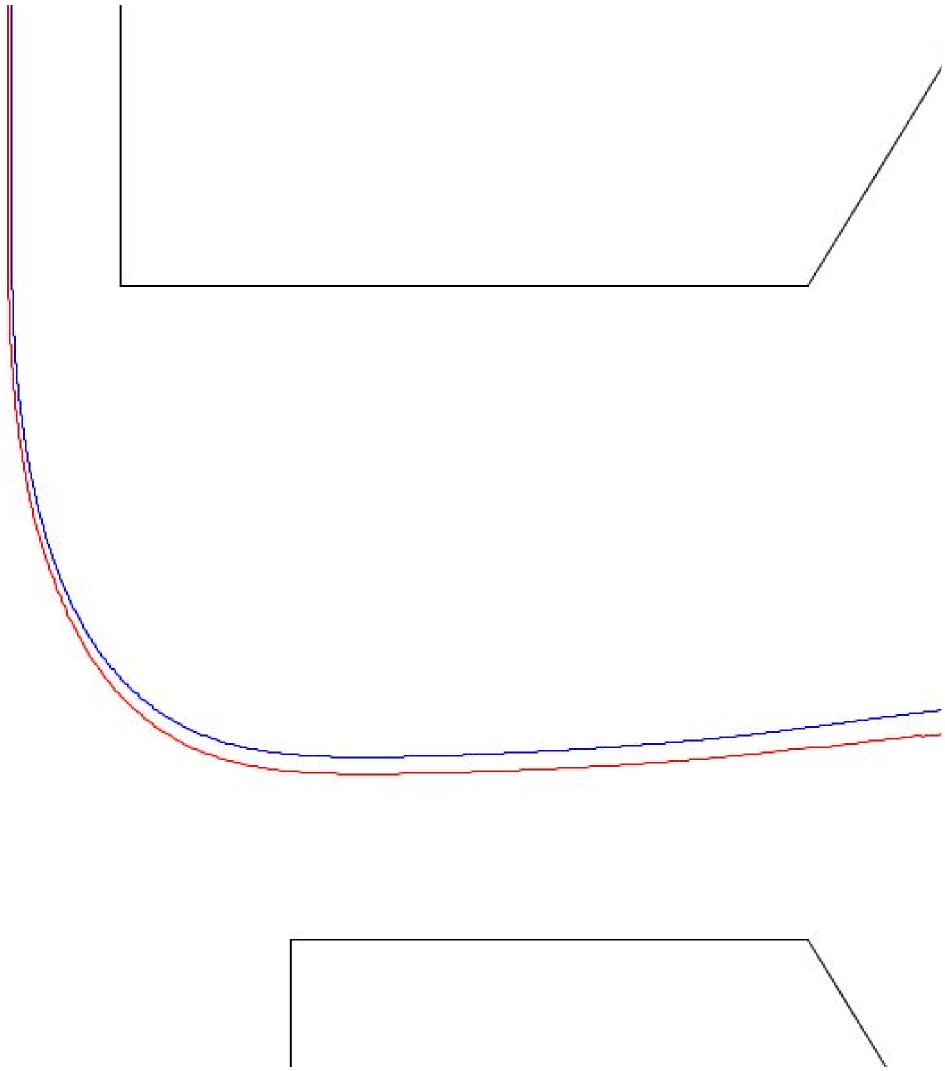

Figure 6.52: 0.10 μm particle path, drag law 2, AeroTrack (blue), FLUENT (red), simulation 5

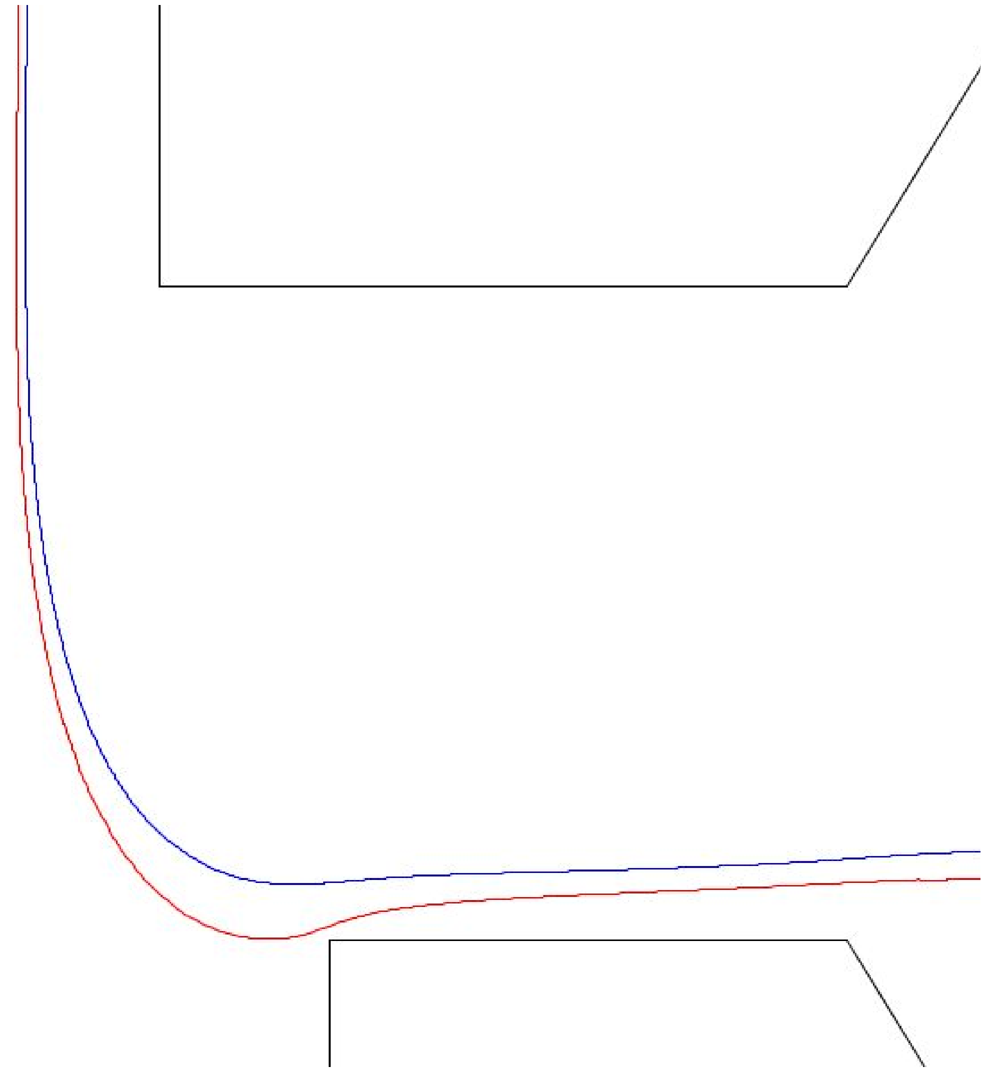

Figure 6.53: 0.25 μm particle path, drag law 2, AeroTrack (blue), FLUENT (red), simulation 5

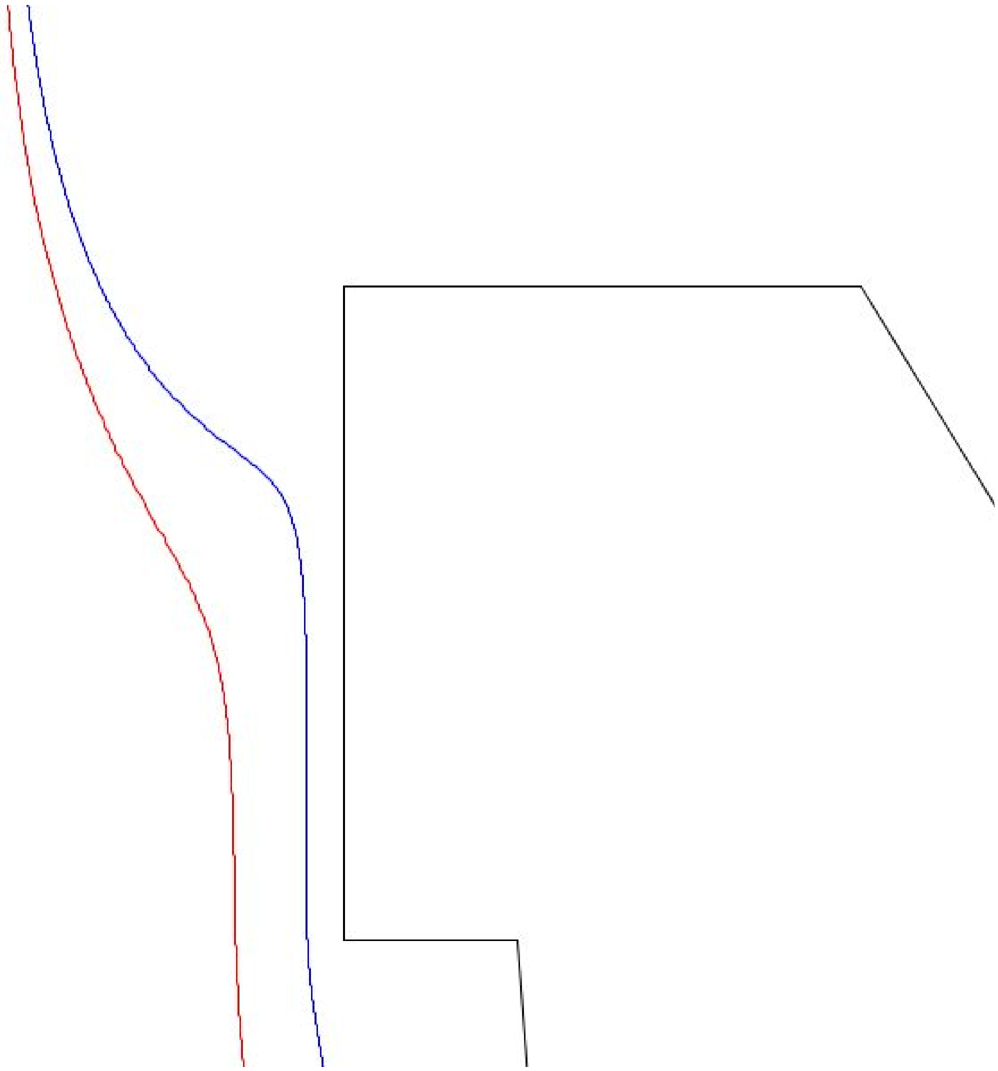

Figure 6.54: 0.40 μm particle path, drag law 2, AeroTrack (blue), FLUENT (red), simulation 5

Recall that the two programs differ in the integration scheme, and in the interpolation of the fluid velocity to the particle position. In order to convey the root of the disagreement between the two programs, despite the use of the same time stepping technique, we present a series of plots that show the particle velocity, and particle travel time. Figures 6.55-6.57 show the components of particle velocity as a function of residence time for the 0.10 μm particle path given in Figure 6.52. At time zero the particle starts at the top cone inlet, and gradually experiences an increase in velocity (Figure 6.56). Approximately 6e-05 seconds later, the particle reaches the farthest point down into the virtual impaction zone (lower apex of Figure 6.55) before it starts to deflect sideways into the major flow, thus increasing its cross-stream velocity. Figure 6.58 is a plot of the time step values computed by each program along the aforementioned path.

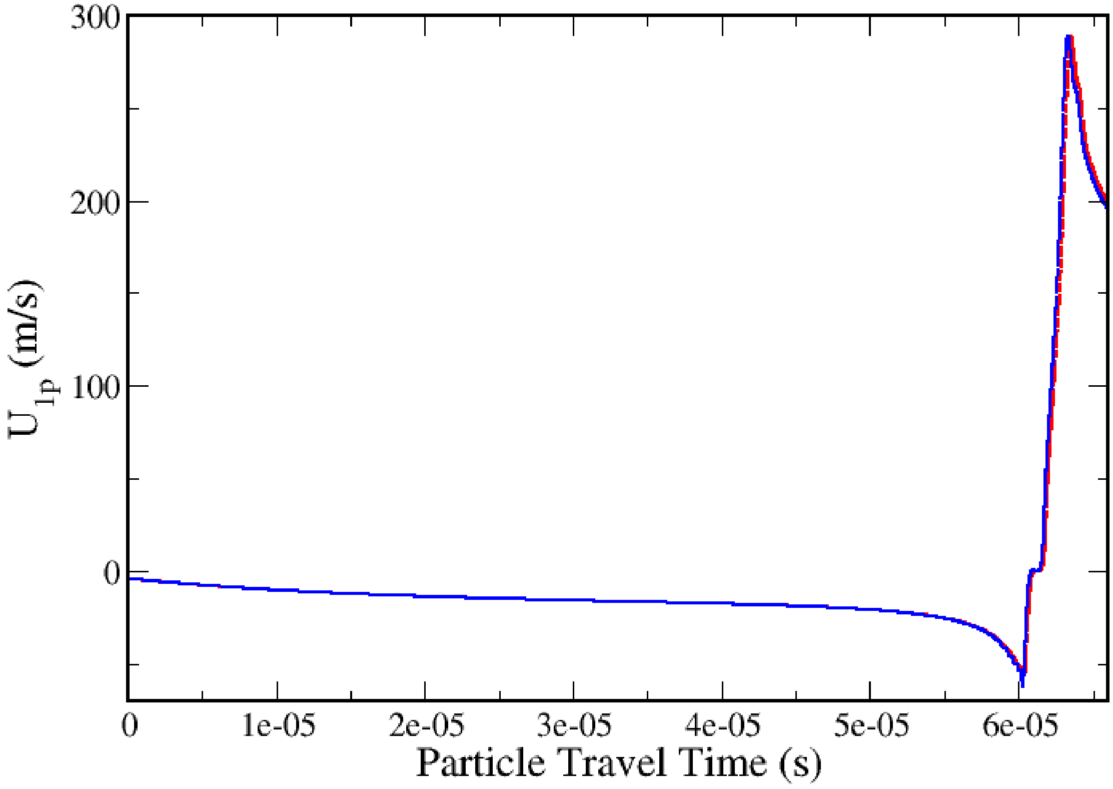

Figure 6.55: 0.10 μm particle cross-stream velocity, AeroTrack (blue), FLUENT (red), path shown partially in Figure 6.52

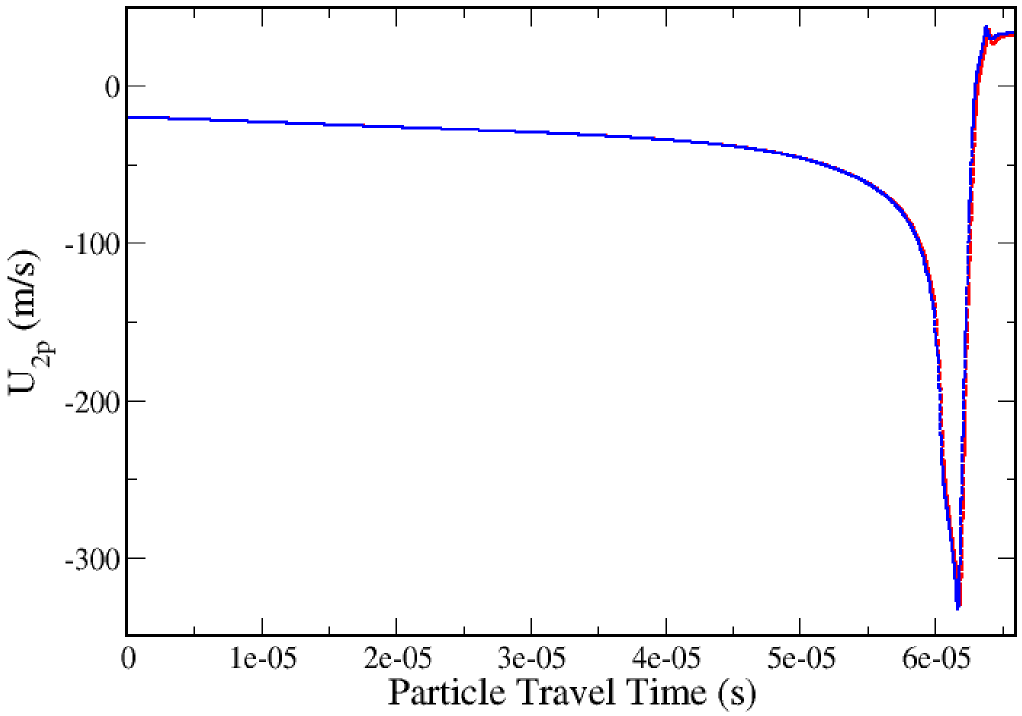

Figure 6.56: 0.10 μm particle streamwise velocity, AeroTrack (blue), FLUENT (red), path shown partially in Figure 6.52

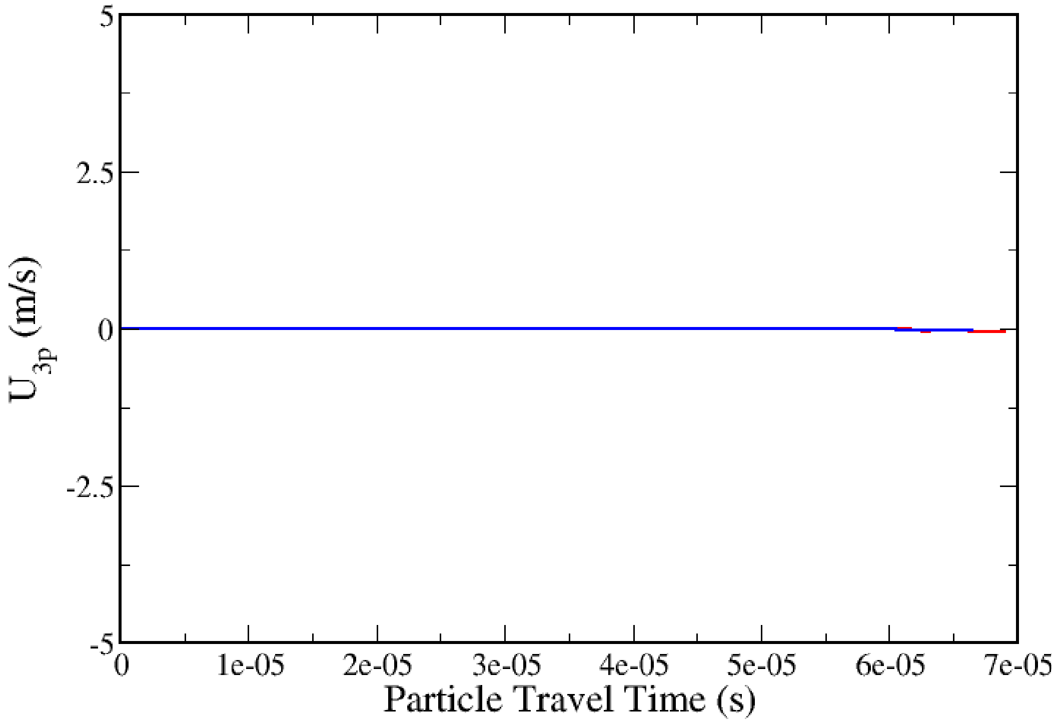

Figure 6.57: 0.10 μm particle spanwise velocity, AeroTrack (blue), FLUENT (red), path shown partially in Figure 6.52

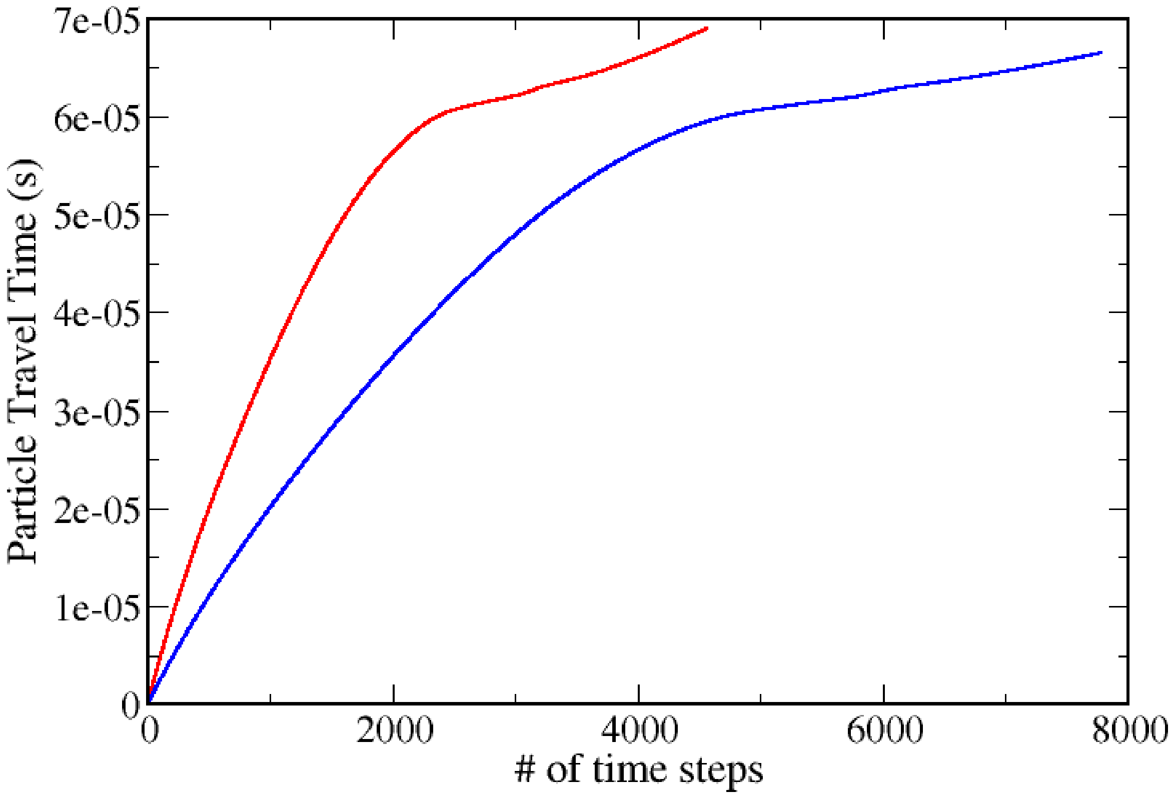

Figure 6.58: 0.10 μm particle time steps, AeroTrack (blue), FLUENT (red) path shown partially in Figure 6.52

The dynamic time stepping equation in Section 5.2.1, relates the time, length, and velocity scales by:

$$\Delta t = \frac{\Delta L}{\left|\vec{U}_p + \vec{U}\right|}. \tag{6.5}$$

Since the two programs utilize the same ΔL value, and compute identical particle velocities (Figures 6.55-6.57), it is clear that the fewer number of time steps taken by the FLUENT tracking program (Figure 6.58) is due to the difference in interpolating the fluid velocity. The same stance is applicable to the other particle paths. Here, we only show the time steps plot for the 0.4 μm particle path in Figure 6.59.

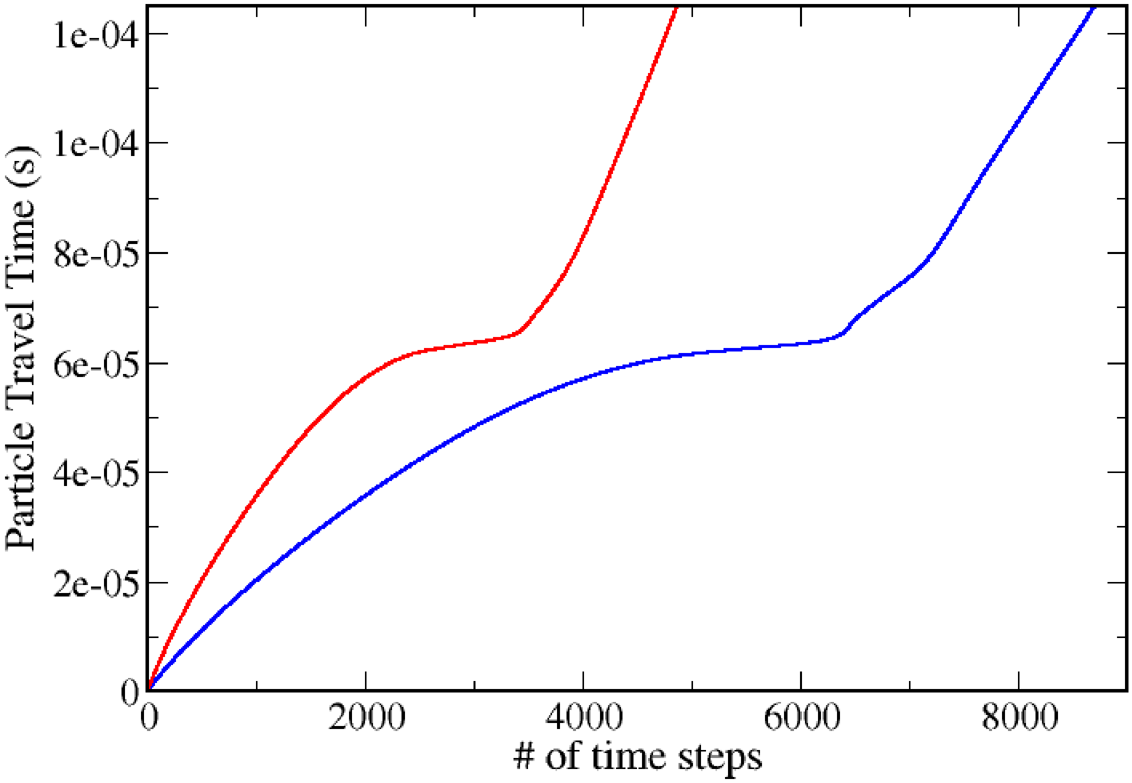

Figure 6.59: 0.4 μm particle time steps, AeroTrack (blue), FLUENT (red), path shown partially in Figure 6.54

The time steps plots are useful in revealing the rate of change of the dynamic time step Δt. It is evident that both programs depict time stepping profiles consistent with the flow dynamics. That is, initially the velocities in the inlet cone are small, so the Δt values are high until the particle reaches the virtual impaction zone. Here, the velocities are high, so the Δt values begin to increment very slowly as seen in the plateau regions of Figures 6.58 and 6.59. There is, however, a significant difference in the rate of change of Δt as computed by each program (slope of the time curves). It seems that AeroTrack requires twice as many time steps to arrive at the same final residence time. Clearly, FLUENT^{TM} has the advantage of rapidly calculating the overall path, but that makes it susceptible to producing quite large Δt values, thus running the risk of "skipping over" some important features of this rapidly changing flow field. Of course, a counter argument may suggest reducing the fixed length step ΔL. However, this can result in excessive or redundant interpolation calls, especially if the grid size is significantly larger than ΔL. Albeit, one is still faced with incrementing the numerical error by resorting to linear interpolation. The issue of "skipping over" is most crucial near a solid boundary, and if Δt is large close to the wall, then most likely the particle motion in such a region, and consequently the wall losses, will not be predicted accurately. Furthermore, since FLUENT's linear interpolation relies solely on one nearest Eulerian meshpoint as opposed to the collection of scattered points utilized by AeroTrack's interpolation scheme, the particle is more susceptible to experience discontinuities in the fluid velocity in the former method (node values at the wall carry a zero velocity). Based on the findings presented here, on the theoretical results of Section 4.2.4b, and on the interpolation tests of Section 5.2.3, the remaining work in this study will only consider discrete phase results computed by AeroTrack. It will focus on a comparison of different drag laws, most of which are not available in the FLUENT^{TM} program.

6.2.3 The Drag Coefficient Revisited

In the previous sections, we adopted the Stokes-Cunningham drag law (drag law 2 in Table 6.1) to account for "slip" on the sub-micron particles. In this section, we investigate the assumption of the constant slip correction factor incorporated into the aforementioned model. Moreover, the compressible form of the drag coefficient is assessed. The particle Reynolds and Mach numbers are two critical quantities that project the influence of the slip velocity, and compressibility. Figures 6.60 to 6.62 show the variation of these two dimensionless variables along the paths of three particles of size 0.10, 0.25, and 0.40 μm, respectively. Although the paths were computed using the mean velocity field of simulation 4, they are very much similar to the ones shown in Figures 6.52 to 6.54, starting from the same initial position. The plots are labeled with the locations in the device where those quantities exhibit peak values. The two quantities are linearly related due to the constant physical properties used in their respective definitions (Section 4.2.1). Not surprisingly, the regions where the most slip occurs are at the entrance of the throat, at the acceleration nozzle, and at the crucial moment when the particle makes the turn to either the major or minor flow, which is referred to as Virtual Impaction (VI). What is surprising however, is the scale of the slip velocity, which yields unimaginably high particle Reynolds numbers that are in excess of unity for the larger particles. The relative Mach numbers are also high, and well into the realm of compressible particle-fluid flow. Further insight into the effect of inertia can be gained by looking at Figure 6.63, which shows the dimensionless vertical position of the particles as a function of travel time, and can be used as an indicator of the locations of the peaks in the Re_p and Ma_r plots. It is evident that the particle experiences a range of slip conditions during its flight, and undoubtedly reaches high enough relative velocities to enter into the compressible regime. The magnitude of the fluid velocity, as well as that of the particle along its traveled path, is shown in Figures 6.64 and 6.65 for the 0.25 and 0.40 μm particle, respectively. The magnitudes are non-dimensionalized by the speed of sound at ambient conditions (v_{sound} = 331.4 m/s). The smaller particle reaches a Mach number close to one at the exit of the acceleration nozzle and as it is ejected into the major flow. The larger particle only reaches the Mach one condition at the nozzle exit, before it goes into the relatively stagnant minor flow.

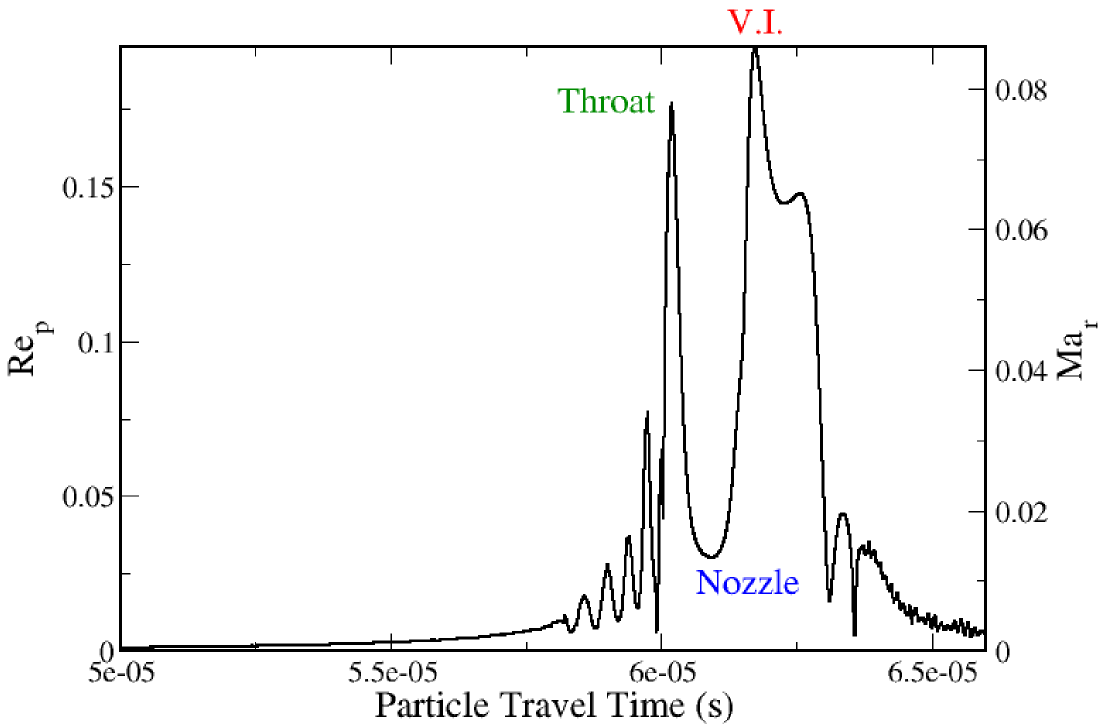

Figure 6.60: Dimensionless slip velocity as a function of time, 0.10 μm particle, drag law 2 – simulation 4

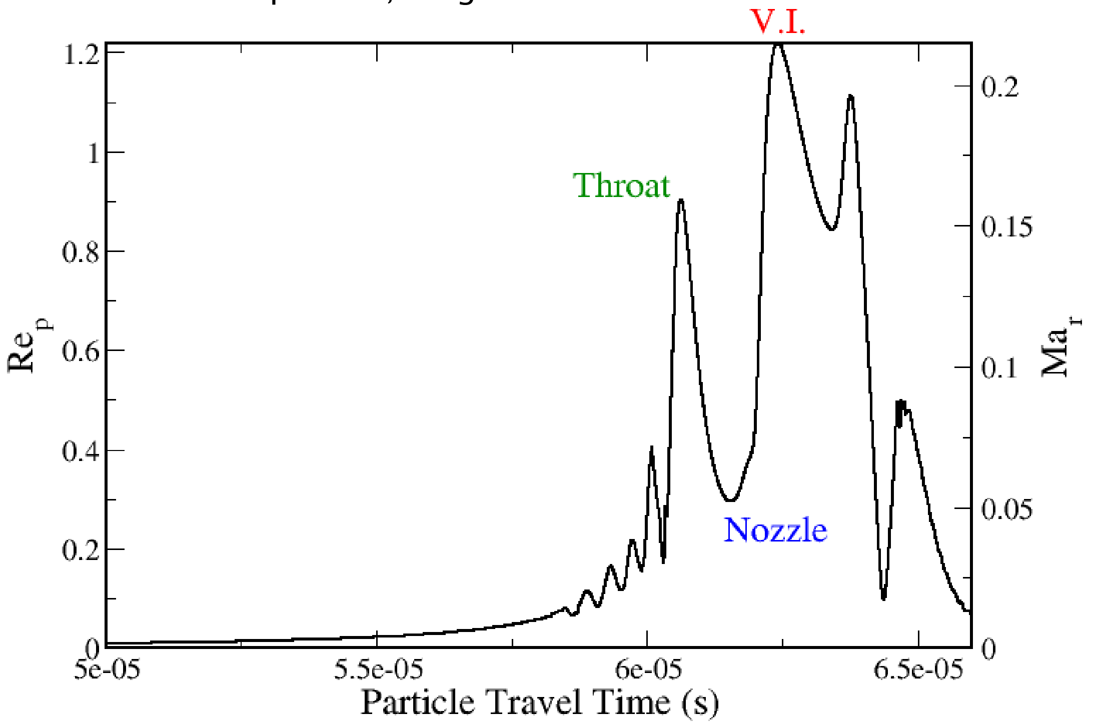

Figure 6.61: Dimensionless slip velocity as a function of time, 0.25 μm particle, drag law 2 – simulation 4

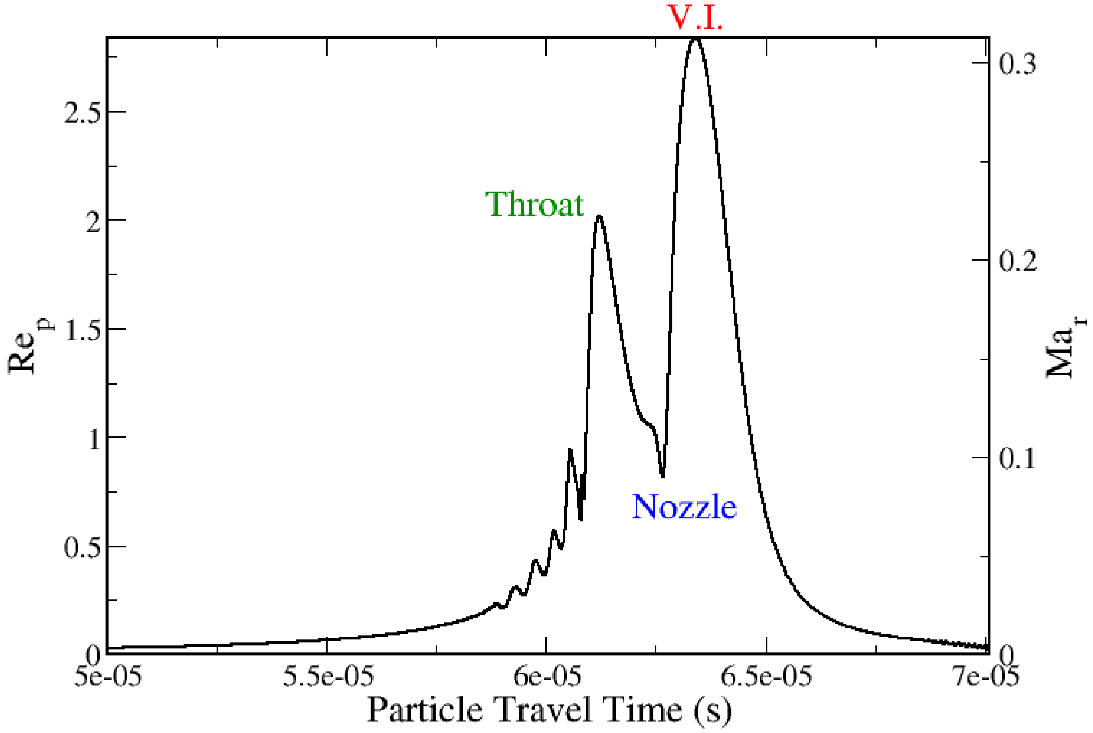

Figure 6.62: Dimensionless slip velocity as a function of time, 0.40 μm particle, drag law 2 - simulation 4

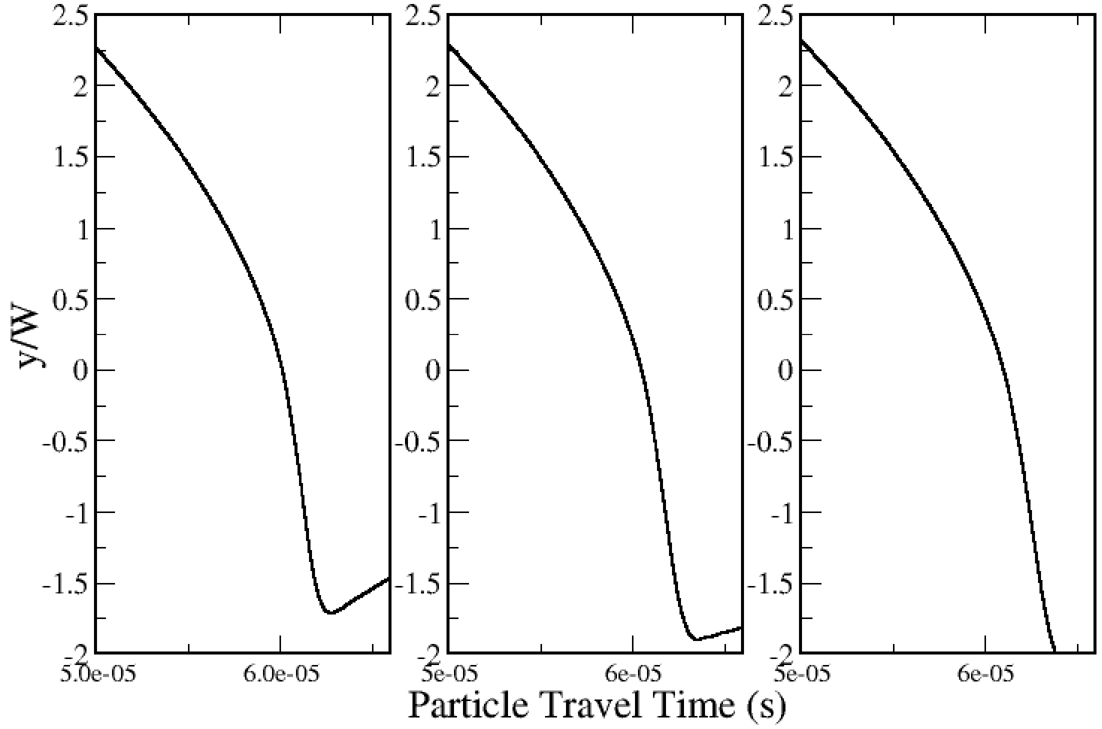

Figure 6.63: Dimensionless particle descent as a function of time, 0.10, 0.25, and 0.40 μm particle, respectively. drag law 2 - simulation 4

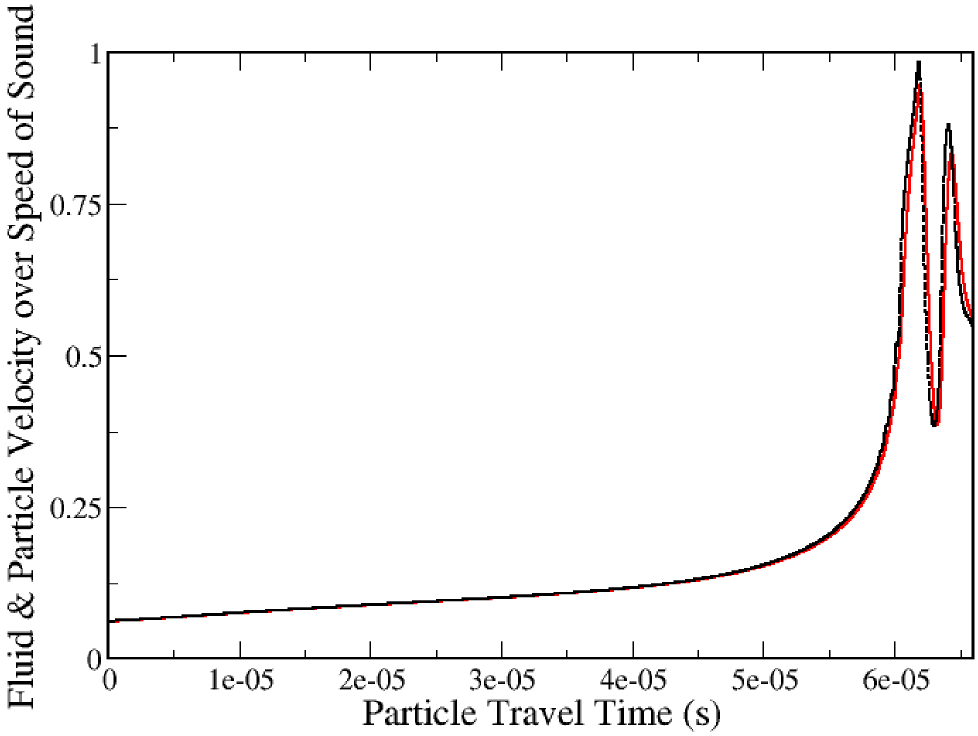

Figure 6.64: Fluid velocity (black) and 0.25 μm particle velocity (red) drag law 2 - simulation 4

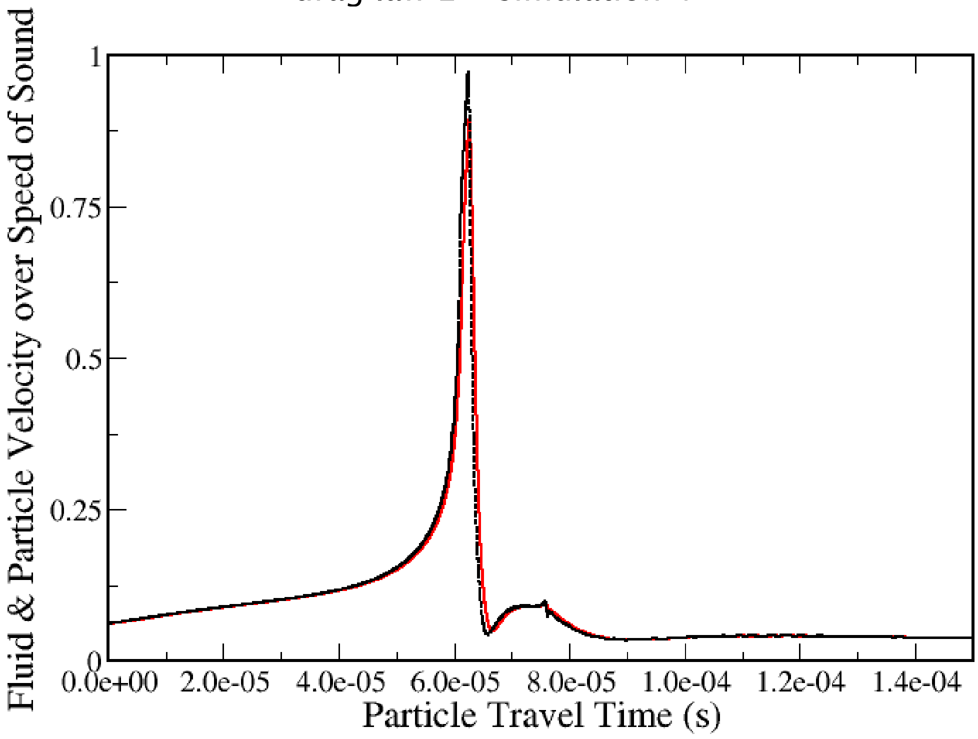

Figure 6.65: Fluid velocity (black) and 0.40 μm particle velocity (red) drag law 2 - simulation 4

6.2.3a More Realistic Models for the Drag Coefficient

Based on the discovered compressibility conditions, the compressible form of the drag coefficient is employed (drag law 4), and a new efficiency and wall loss curves are generated. Furthermore, another form of the Stokes-Cunningham law is investigated by allowing the slip correction factor C_c to vary based on the surrounding pressure field (drag law 3) (see Table 6.1). Figure 6.66 compares the predicted performance curves for the three drag laws: Stokes-Cunningham (#2), Stokes-Cunningham with a pressure dependent C_c (#3), and the compressible drag coefficient (#4) using the flow field information of simulation 4. Clearly, the two new formulations of the drag coefficient alter the efficiency curve by shifting it to the right. In other words, the collection efficiency of larger particles is reduced. This indicates that the particles are more closely following the fluid for drag law 3 and 4, thus collecting less in the minor flow. The increased drag on the particles can also be concluded from Figure 4.2 which re-enforces the fact that the traditional Stokes-Cunningham law gives lower drag coefficient values at relative Mach numbers greater than 0.1 than the compressible law. We have demonstrated above that the particles do in fact encounter relatively higher Mach number conditions (Figures 6.61-6.62). The adequacy of the compressible drag coefficient is further emphasized when we compare the performance curves of simulation 4 and 5, in Figure 6.67.

In reference to Figure 6.51, the efficiency curves given by the Stokes-Cunningham drag law contradict what is known experimentally about the effect of the minor-to-total flow ratio. The slightly higher Q_m/Q_T ratio of simulation 5 ought to enhance the collection efficiency and reduce the wall losses. This behavior is certainly not depicted in Figure 6.51. However, it is visibly the case in Figure 6.67. As mentioned before, the flow fields of the two simulations are minimally variant, so the outcome of the efficiency curve should not be expected to differ significantly. Fortunately, the use of the compressible drag coefficient model was able to confirm this principle, while the traditional Stokes-Cunningham law could not.

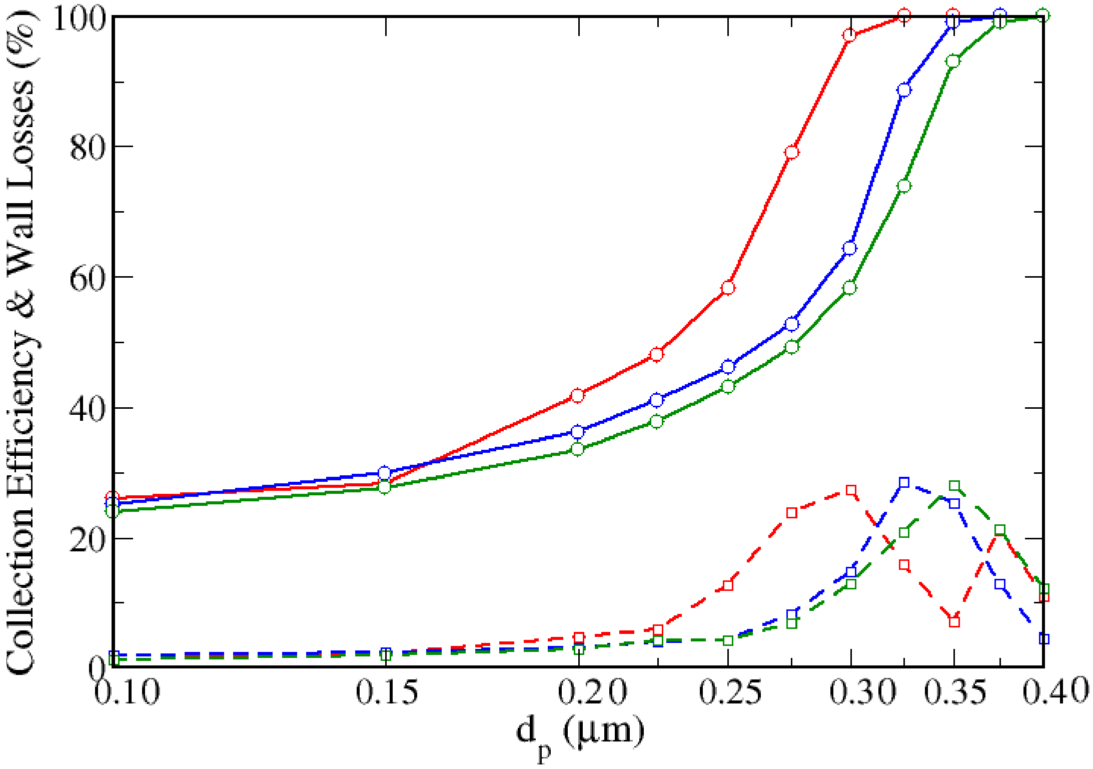

Figure 6.66: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 2 (red), drag law 3 (blue), drag law 4 (green)-simulation 4

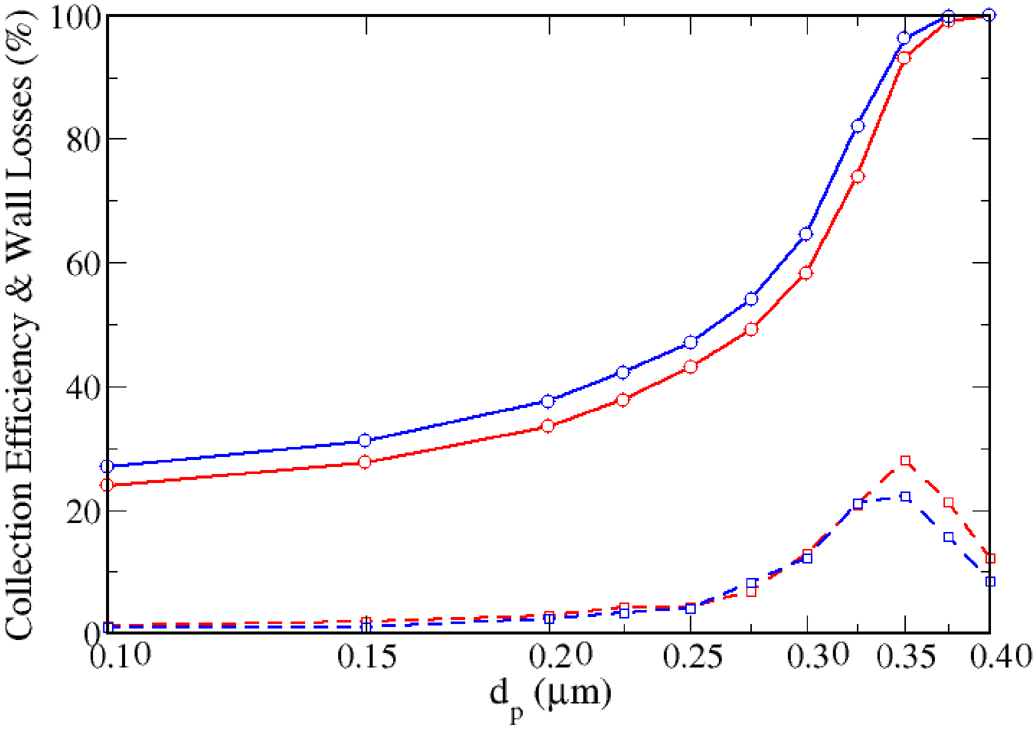

Figure 6.67: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 4, simulation 4 (red), simulation 5 (blue)

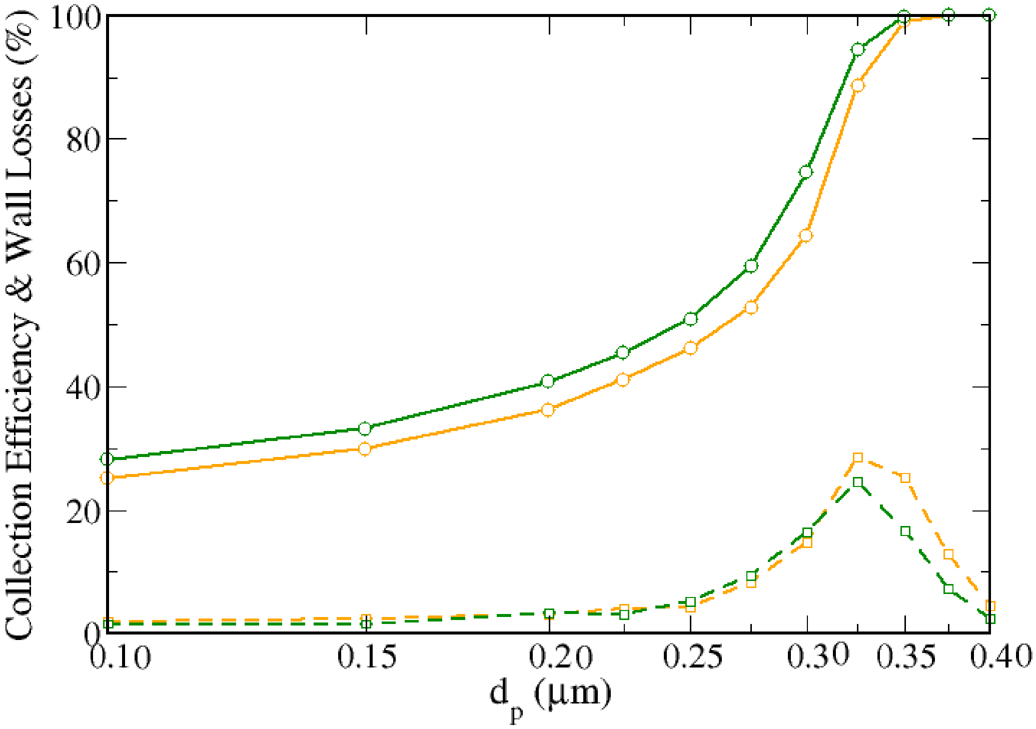

Figure 6.68: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 3, simulation 4 (orange), simulation 5 (green)

The other alternative drag coefficient model, namely the pressure dependent form of the Stokes-Cunningham law (drag law 3), appears to produce quite viable results. The proper influence of the Q_m/Q_T ratio is predicted correctly, as shown in Figure 6.68. The collection efficiency and wall losses are reasonably in agreement with those of the compressible drag coefficient, as shown in Figure 6.69. The emergence of this formulation as a superior substitute to the original formulation (drag law 2) is surely demonstrated. After all, keeping the slip correction factor constant is an assumption that is not universally valid for all types of flows. On the other hand, the question of whether this formulation or the compressible one is superior is not currently addressable, due to the lack of reliable experimental evidence.

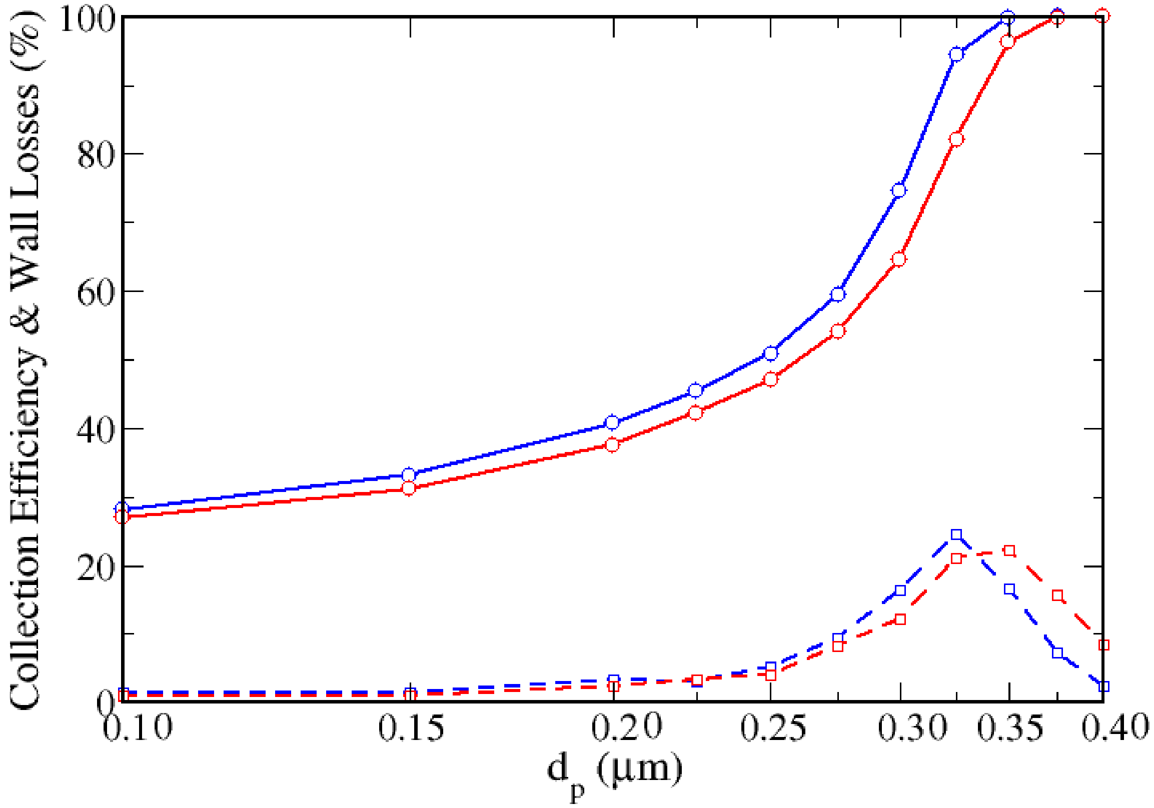

Figure 6.69: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 4 (red), drag law 3 (blue), simulation 5

6.2.4 Incompressible vs. Compressible Flow

In this section we further analyze the influence of the predicted fluid velocity field on the particle trajectories, and on the efficiency and wall loss curves. The two most prominent candidates for this task are the results of simulations 2 and 5 of Table 5.1. The Reynolds stress model demonstrated interesting flow features that were absent from the incompressible κ-ϵ model simulation, and it appeared to allow the fluid to flow at comparable velocities to the compressible simulation. The two simulations attained equal flow separation ratios, Q_m/Q_T = 22%, and exhibited comparable fluid residence times (Figure 6.38). To obtain a general view of a characteristic particle motion, Figures 6.70 to 6.72 display the paths undertaken by particles of size 0.10, 0.25 and 0.30 μm, respectively, released from the same location near the inlet of the device, and tracked on the two mean flow fields using the pressure dependent Stokes-Cunningham drag coefficient (drag law 3).

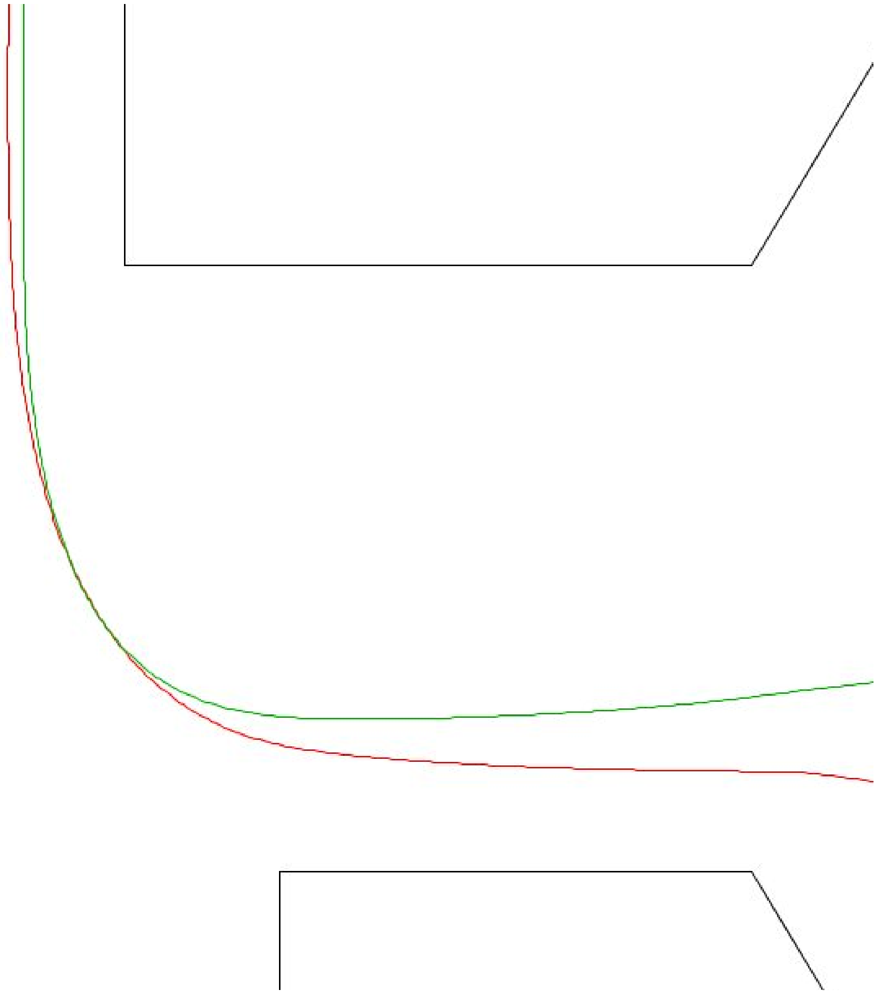

Figure 6.70: 0.10 μm particle path, simulation 2 (red), simulation 5 (green), drag law 3

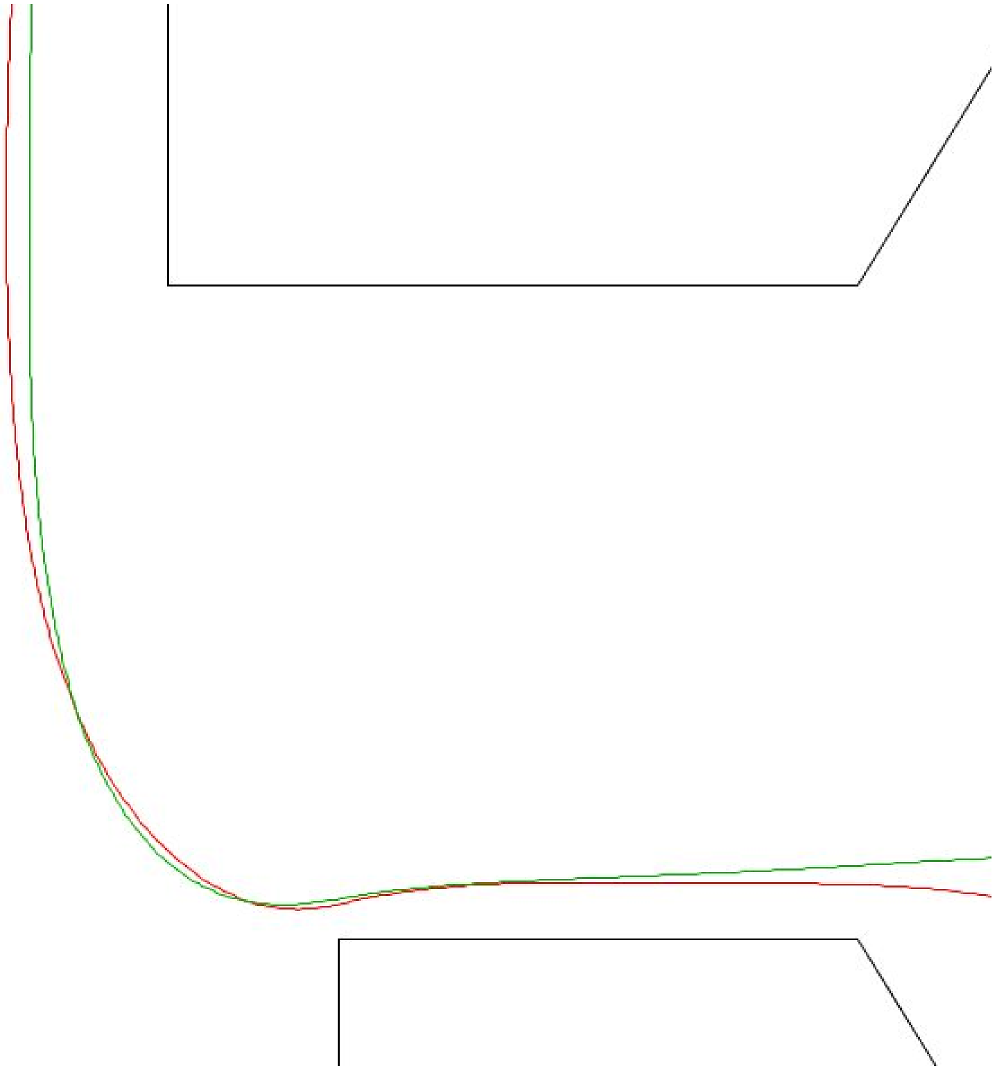

Figure 6.71: 0.25 μm particle path, simulation 2 (red), simulation 5 (green), drag law 3

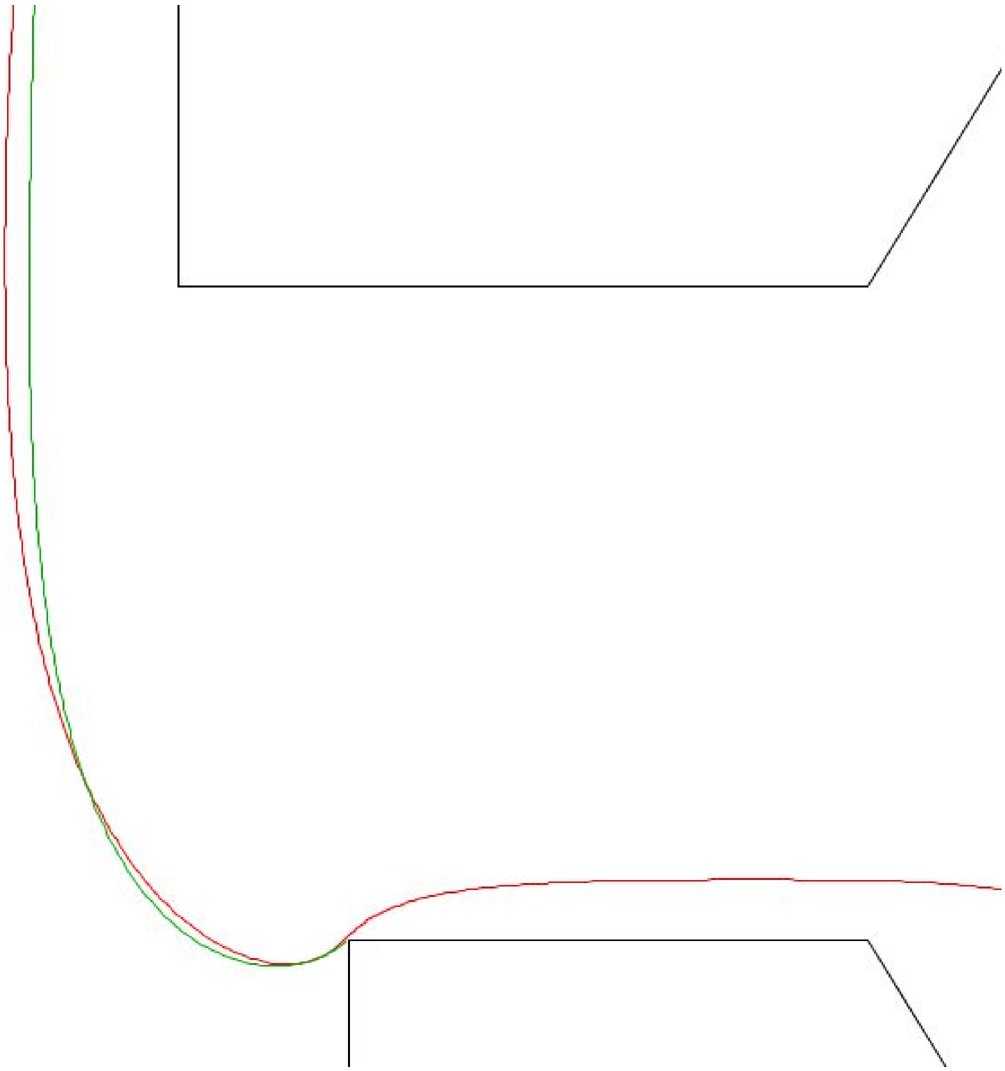

Figure 6.72: 0.30 μm particle path, simulation 2 (red), simulation 5 (green), drag law 3

The motion in the entrance cone section (not shown) is identically predicted for the two flow fields, until the particle reaches the throat, where there is an unsubstantial mismatch due primarily to the difference in the fluid velocity profile in the throat. Notably, the deflection into the major flow is very much alike for the all three particles, despite the discrepancy in the final fate of the largest particle. The fact that simulation 5 forces the 0.30 μm particle to hit the collection nozzle wall, whereas simulation 2 allows it to continue onto the major flow, can be attributed to numerical error due to the coarseness of the grid in the latter simulation. Nonetheless, the two velocity fields seem to have a comparable influence on the particle motion, and hence are expected to yield not so different efficiency curves. In fact, the outcome of tracking the entire sample of particle sizes is shown in Figure 6.73. The improved collection efficiency of simulation 5 is naturally due to the higher velocities. The wall losses are in disagreement, primarily because of differences in the flow structure, and possibly because of the numerical error associated with interpolating on a coarser grid. Yet they are still within reasonable limits. Finally, similar to the analysis done for the motion of fluid elements from the nozzle in Section 6.1.5, Figure 6.74 presents the normalized residence time in each flow field calculation for a sample of 50 particles with three characteristic particle sizes of diameter 0.10, 0.25 and 0.40 μm released from the exit of the accelerating nozzle. The graphs are labeled from top to bottom with the the particle size and corresponding square-root Stokes number.

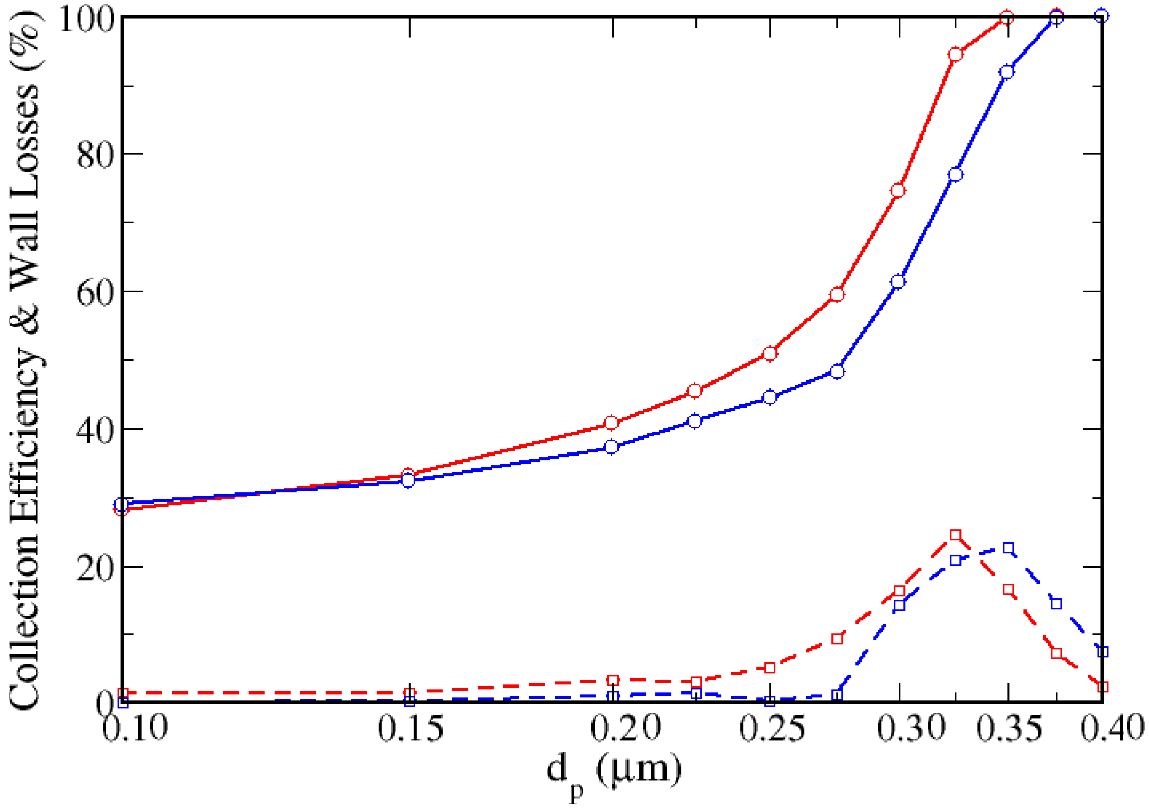

Figure 6.73: Collection efficiency (solid lines), and wall loss (dashed lines), drag law 3, simulation 2 (blue), simulation 5 (red)

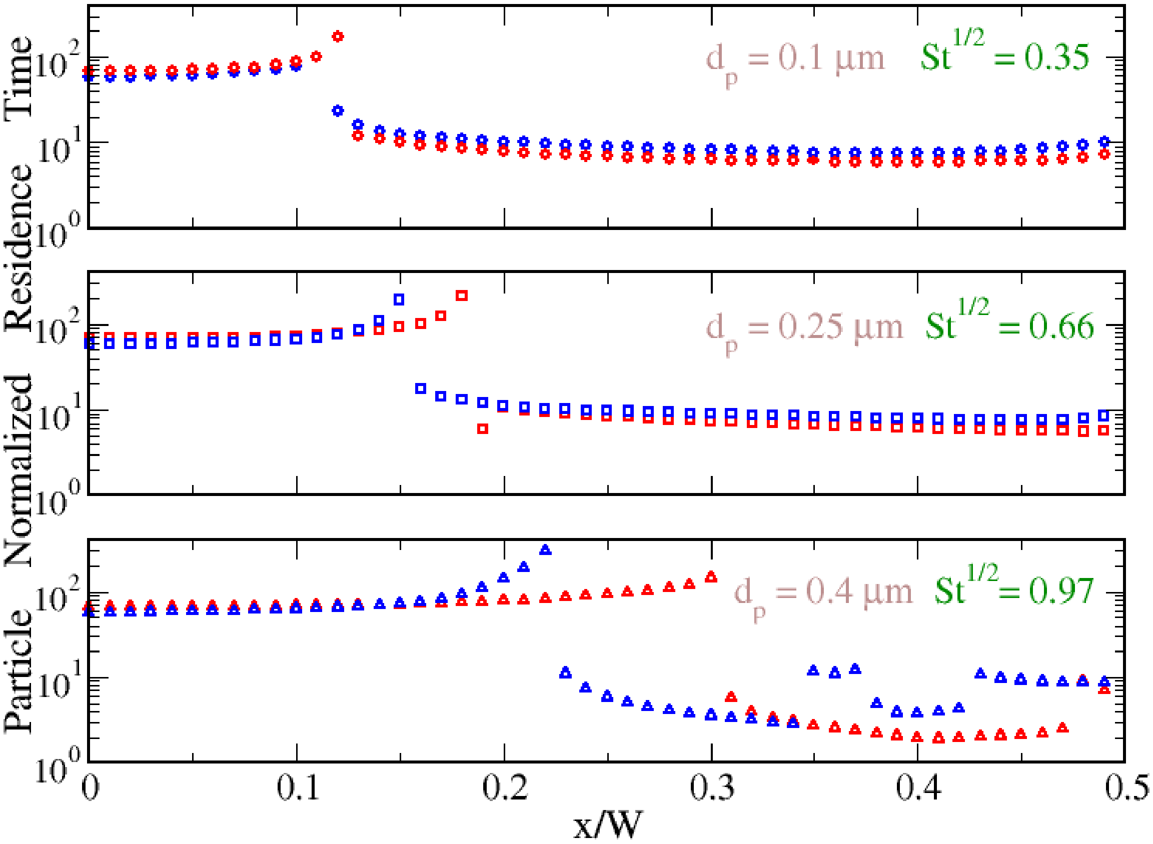

Figure 6.74: Normalized residence time for discrete nozzle particles, drag law 3, simulation 2 (blue), simulation 5 (red),

The residence time plot provides a definitive account of the flow dynamics in the virtual impaction zone. First, it is evident that the effect of fluid motion on the particle motion, as predicted by either simulation, is invariant for small particles ($St^{1/2} \approx 0.35$). In comparison to the motion of fluid points in Figure 6.38, the cutoff distance, which marks the distance from the center of the nozzle beyond which the particles are bound to deflect into the major flow, is shifted further away from the 10% position observed for fluid particles. For the medium size particles ($St^{1/2}$

≈ 0.66), the flow patterns are different for particles entering the virtual impaction zone between 15 to 20% of the nozzle width. It is clear that the flow field of simulation 5 will extend the cutoff distance for these particles beyond that of simulation 2, thus enhancing their collection efficiency. This occurs at the expense of a wall loss, as seen from the “lonely” particle with the smallest residence time in the middle plot. This observation is also conspicuous in Figure 6.73 which shows higher collection efficiency and losses for the 0.25 μm particle diameter. For the largest particles ($St^{1/2} \approx 0.97$), the two flow fields predict quite distant cutoff distances. However, they both yield equivalent collection efficiencies (Figure 6.73, 100% data points). The losses are higher in the Reynolds stress model simulation, which is indicated by the larger number of particles with relatively shorter residence times. The issue of the losses can be further clarified by looking at the particle trajectories of 0.40 μm particles shown in Figures 6.75 and 6.76. The former figure reveals that the particles in the collection nozzle closer to the wall have the tendency to move upwards, therefore, hitting the side walls more often. In addition, theres seems to be an accumulation of particles near the lower boundary of the impaction zone, which also contributes to higher losses in the Reynolds stress model simulation.

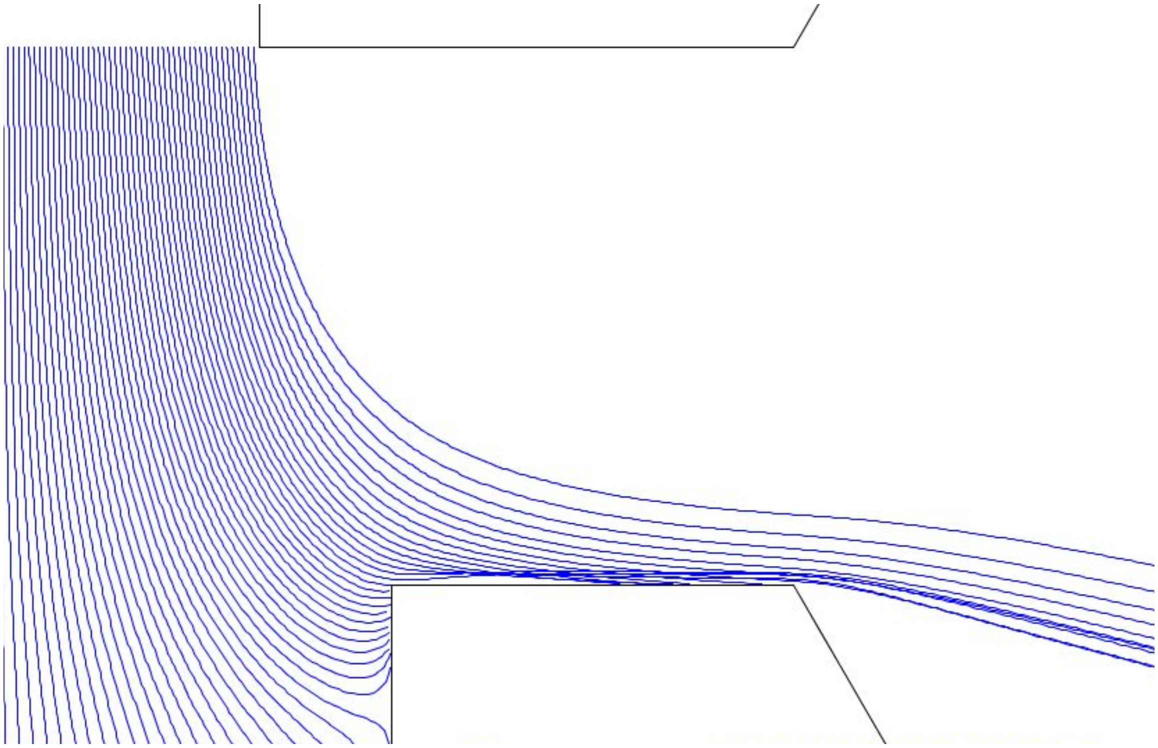

Figure 6.75: 0.40 μm particle paths from nozzle, simulation 2, drag law 3

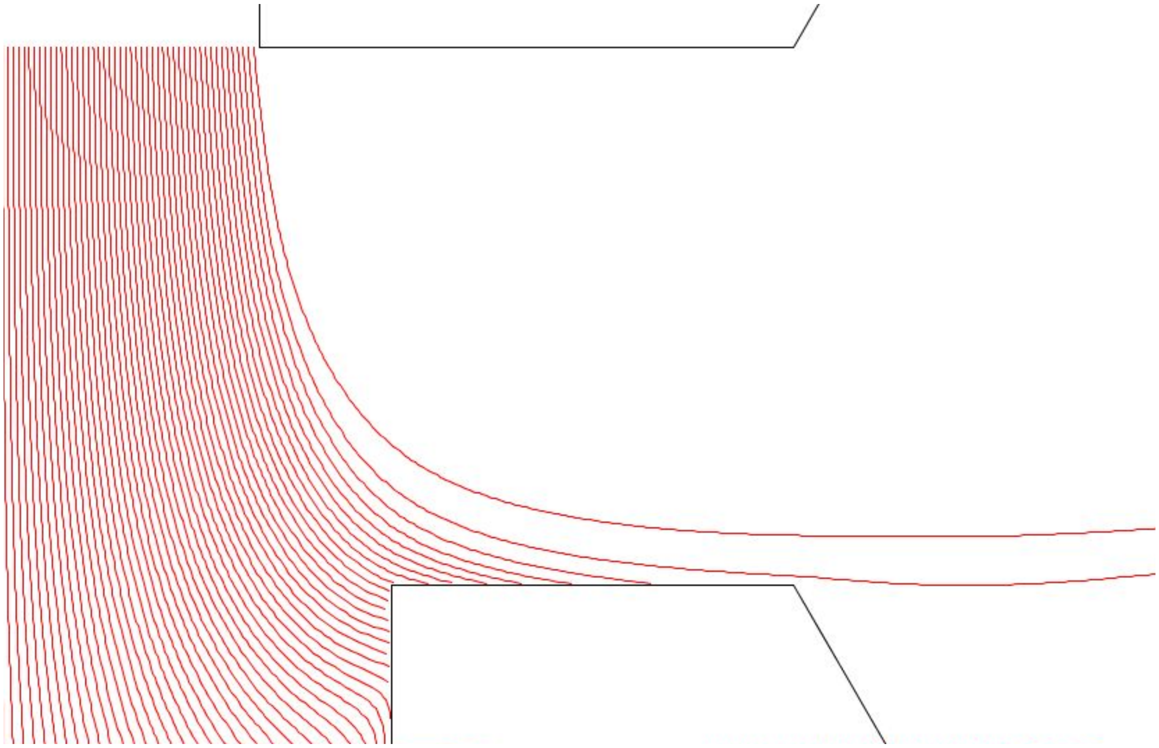

Figure 6.76: 0.40 μm particle paths from nozzle, simulation 5, drag law 3

6.2.5 Comparison with Experiment

The lack of reliable experimental data makes the validation of the numerical results a difficult task. Despite the existence of preliminary collection efficiency and wall loss data from the Sioutas' experiments [3], his claim of a 0.12 μm cutpoint was criticized within the aerosol community as being low [48], and his findings were never reproduced in a subsequent scientific publication. In fact, a recent study by Ding and Koutrakis [49] suggests quite different behavior for the efficiency and wall loss curves.

The following facts about virtual impactors were demonstrated in this research; across all the physically realistic computations:

1. The collection efficiency curve exhibits a steep slope at high enough Reynolds number conditions (transition to turbulent regimes).
2. Increasing the minor-to-total flow ratio enhances the collection efficiency and reduces the wall losses.
3. The wall losses curve shows a local maximum at a particle size corresponding approximately to the 50% cutpoint.
4. The peak value of the particle losses lies between 20% to 30% for a virtual impactor with similar geometrical and dynamical conditions [15,49].

To compare the model predictions to literature data, the discrete phase results from the compressible mean fluid velocity tracking calculation

(Figure 6.67; simulation 5) are plotted versus $St^{1/2}$ in Figure 6.77. This particular simulation was chosen since it possesses the most accurate numerical components (fine gird, 2nd order discretization), and employs a realistic model for the drag coefficient.

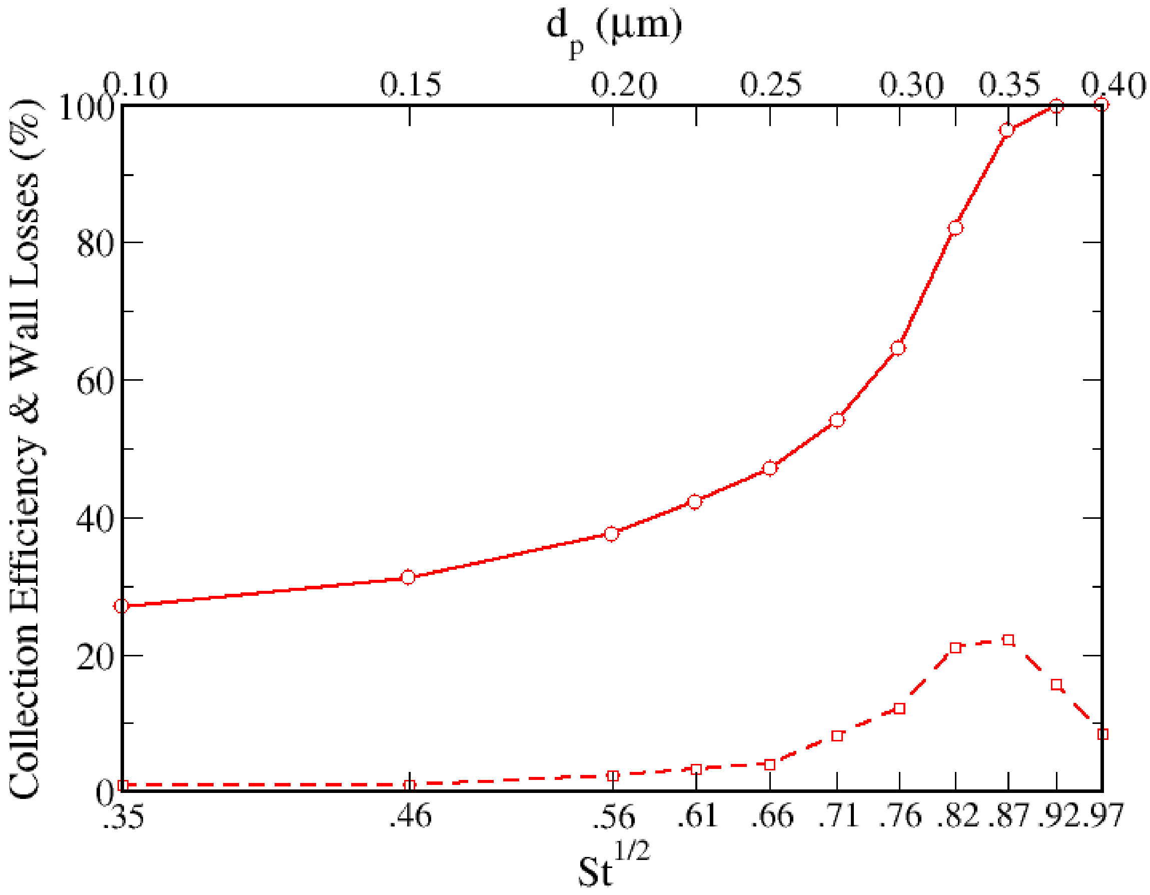

Figure 6.77: Collection efficiency (solid line), and wall loss (dashed line) vs. $St_{50}^{1/2}$, drag law 4, simulation 5

The above plot indicates that the 50% cutpoint Stokes number is between 0.66 and 0.71. The 50% cutpoint diameter, $d_{p\ 50}$, is between 0.250 and 0.275 μm. The $St_{50}^{1/2}$ is in good agreement with the reported values of

Ding and Koutrakis [49] who performed experiments on a geometrically equivalent impactor (W = 0.305 cm, W_1 = 0.427 cm, T = 0.330 cm, and S = 0.457 cm). Their values are 0.71 and 0.68 for nozzle Reynolds numbers of 6160 and 10,220 (based on hydraulic diameter) corresponding to a total flow rate of 30 LPM and 50 LPM, respectively. The minor-to-total flow ratio was 10%. Recall that the Reynolds number in simulation 5 is 9169, and Q_m/Q_T is 22%. Ding and Koutrakis also report a $St_{50}^{1/2}$equal to 0.47 at 20% flow separation but at a Reynolds number of 10,220. Granted that the simulation Reynolds number is less than this value, we expect the $St_{50}^{1/2}$ to be slightly higher. For completeness, the experimental data of Sioutas is shown in Figure 6.78, along with the numerical results of simulation 5 as presented in Figure 6.69. Aside from the obvious shift in the 50% cutpoint, it seems that the experimental curves do not capture the trends established in the literature; namely the steepness of the efficiency curve for larger particles, and the formation of a peak in the wall losses curve. On the other hand, the simulation results depict the proper shape for both curves, and quantitatively capture the upper bound of the wall losses peak. Further insight into the influence of the Reynolds number, and the shape of the efficiency curve is given in Figure 6.79, which shows the experimental data of Ding and Koutrakis. In addition, the typical behavior of the wall loss curve can be seen in Figure 6.80 at increasing values of the minor-to-total flow ratio.

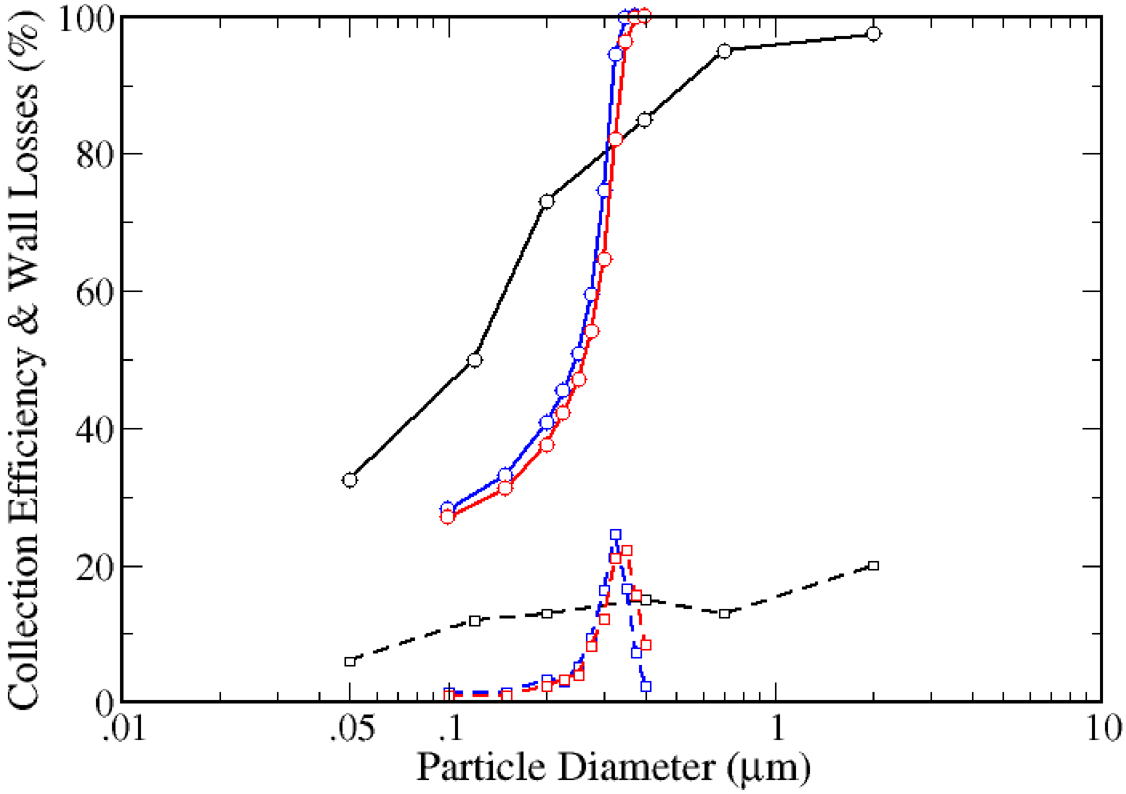

Figure 6.78: Collection efficiency (solid lines), and wall loss (dashed lines), Experiment [3] (black), drag law 4 (red), drag law 3 (blue), simulation 5

The experimental results of Figure 6.79 clearly show the expected steepness of the efficiency curve. As Re increases, the cutpoint size decreases, and the particle collection efficiency increases. The curves show that the 50% cutpoint particle diameter decreases from 5.8 μm at Re = 1500, to 1.8 μm at Re = 15,000. The impact of the Reynolds number on the square-root of the Stokes number, however, is minimal.

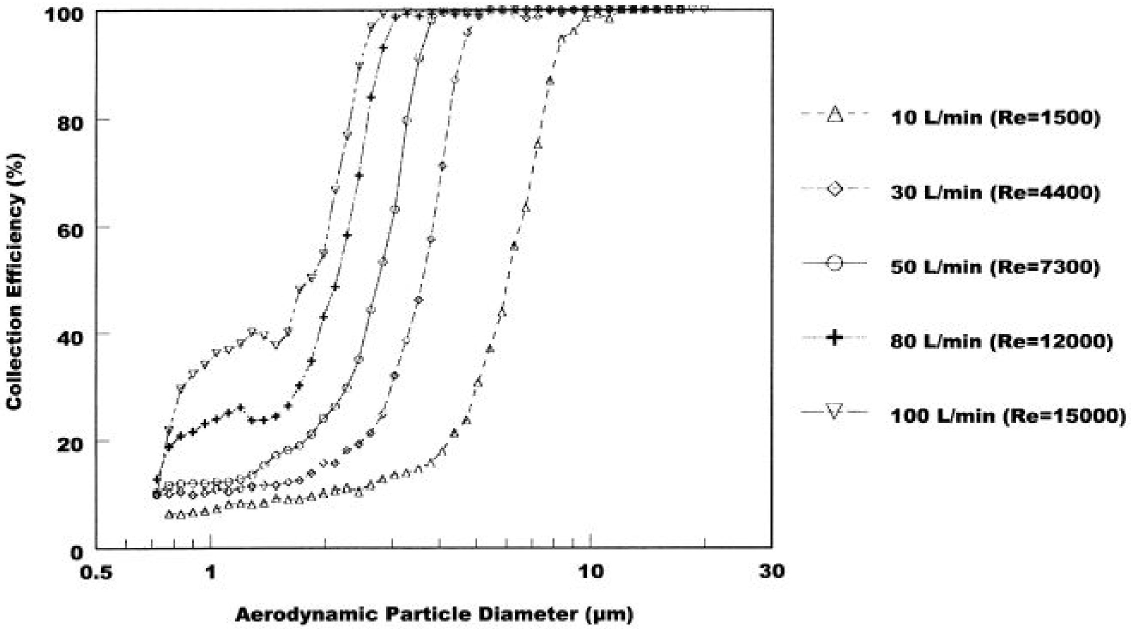

Figure 6.79: Effect of Re on the collection efficiency of a slit virtual impactor from Ding and Koutrakis [49]

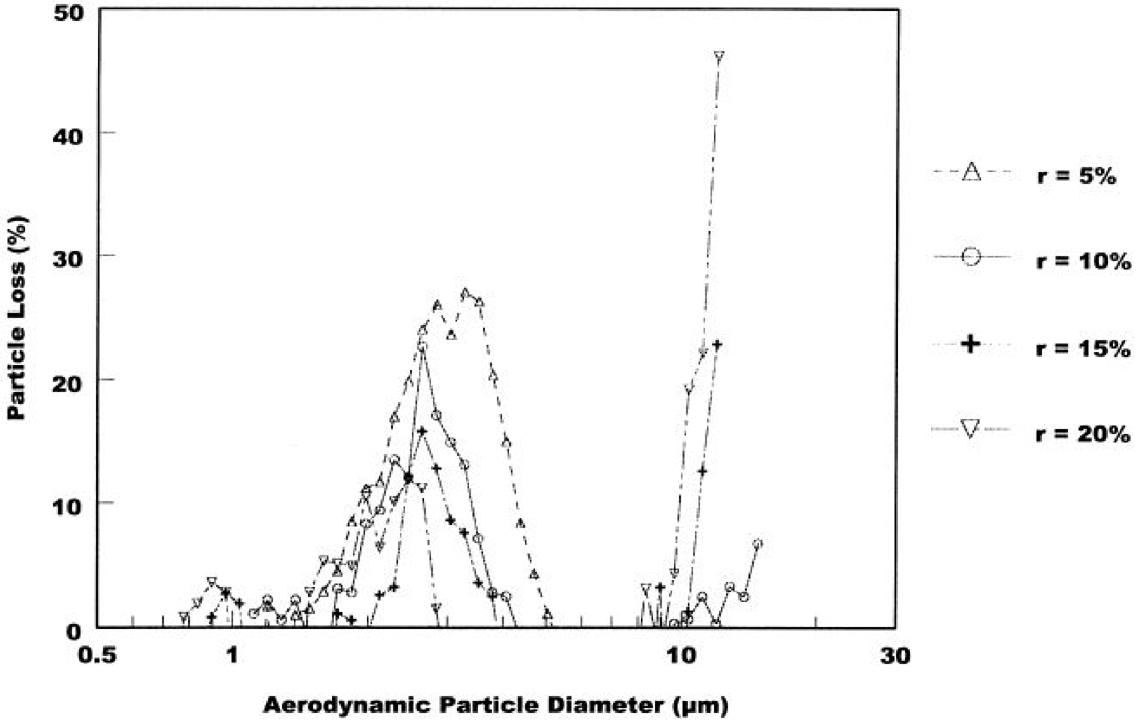

Figure 6.80: Effect of Q_m/Q_T (≡r) on the particle losses in a slit virtual impactor at Q_T = 50 LPM from Ding and Koutrakis [49]

This can be explained by the fact that the average nozzle velocity used in the calculation of the Stokes number is also increased. It should be noted that the length scale used in the calculation of the Reynolds number in the experiments [49] is based on the nozzle width W, whereas in the simulation we use 2W to be consistent with the hydraulic diameter definition. The hydraulic diameter of the slit studied by Ding and Koutrakis is approximately 1.4W.

With respect to wall loss, Figure 6.80 shows a decrease in the observed peak as the minor-to-total flow ratio is increased. For ratios between 10 to 20%, the peak value lies between 20 to 30%, and the curve exhibits a sharp decrease in the wall losses beyond the 50% cutpoint.

In summary, judging by the more recent and more admissible experimental results of Ding and Koutrakis [49] on a large cutoff virtual impactor, the data from Sioutas [3] can be criticized on two grounds. We already alluded to the steepness of the efficiency curve which is missing from the Sioutas plot. In addition, the asymptotic behavior towards 100% collection efficiency at increasing particle size, shown in the Ding and Koutrakis experiments and in the simulations, is not shown in the Sioutas experiment. Lastly, there is a paramount discrepancy in the particle loss data. The more recent experiments, the simulations, and the virtual impactor literature all agree on the appearance of a zenith in the losses curve.

6.2.6 Summary

The particle tracking results using the incompressible flow simulations revealed the inadequacy of the nonlinear model of the drag coefficient (Table 6.1) for the sub-micron particles of interest to this study. The model predicted relatively high cutpoint diameters, and low particle losses. In addition, the incompressible Reynolds stress model along with the Stokes-Cunningham drag law gave more realistic predictions for the efficiency and wall loss curves than the incompressible κ-ϵ model.

The stochastic particle tracking results exposed the incapacity of the isotropic assumption for the fluctuating fluid velocity components to predict realistic behaviors for the efficiency and wall loss curves. This is an indication of the non-isotropic character of the flow. On the other hand, the Reynolds stress model treatment of the fluctuating fluid velocity components resulted in minimal differences between the stochastic and mean fluid velocity tracking results. This is most likely caused by low levels of turbulent kinetic energy predictions.

A comparison between the results of the AeroTrack particle tracking algorithm and those of FLUENT™ revealed the need for a conservative selection of the integration and interpolation scheme to produce small enough time steps that can resolve the rapidly changing flow features.

Furthermore, it was determined that compressibility plays a role in the governing dynamics of the flow. This was shown by monitoring the relative Mach number along characteristic particle paths. For this reason, more realistic models for the drag coefficient were adopted (drag law 3 and 4 in Table 6.1), and consequently, the correct influence of the flow separation ratio on the efficiency and wall loss curves was predicted.

Lastly, a comparison between the improved numerical results and the experimental data in the literature was presented. The numerical results show that the 50% cutpoint diameter is higher than the reported experimental value for the same virtual impactor, under the same flow conditions. Moreover, the particle losses are in disagreement. A comparison with a more recent experiment, however, performed on a geometrically similar virtual impactor (comparable aspect ratios) highlights key similarities in the steepness of the efficiency curve, and the shape of the wall loss curve.

Chapter 7

Conclusions and Recommendations

This thesis presented the results of simulations to assess the performance of a low cutpoint, high sampling flow rate aerosol virtual impactor. The device has very small physical dimensions which makes it suitable as a portable “personal” aerosol sampler. Virtual impactors are ideal for collecting and concentrating particulate mass from the atmosphere, since they allow the particles to remain airborne and retain their physical properties. The virtual impactor studied in this research is capable of separating particles in the sub-micron diameter range, thus making it indispensable for human health studies concerned with the effects of inhalable fine particle matter ($d_p \leq 2.5\ \mu$m).

A two tier approach was undertaken to study the flow dynamics and performance of the virtual impactor. The first step was the prediction of the carrier phase fluid flow field influenced by turbulence. The second was the decoupled prediction of discrete spherical particle motion as governed by the fluid drag force. For the former task, a number of incompressible and compressible flow simulations with distinct turbulence models were employed. For the latter task, two particle tracking algorithms with varying degrees of accuracy were engaged, and a number of drag law models were exploited.

7.1 Summary

From the numerical analysis of Chapter 5, we have found that a significant number of iterations is required to dampen the numerical error across all the fluid flow simulations of Table 5.1. The highest iteration error was of the order 10^{-2}. It was found that a second order discretization scheme is more appropriate for this curved streamline flow since it reduced the iteration error by more than one half. Chapter 5 also dealt with the accuracy of the interpolation scheme implemented in the AeroTrack particle tracking code. The scheme was subjected to a number of interpolation tests to gauge its performance. The results indicate that the method is third order accurate, but computationally expensive, compared to second order accurate but fast linear interpolation.

The main findings of Chapter 6 lie in the comparison between the incompressible standard κ-ϵ turbulence model, the Reynolds stress model, and the compressible κ-ϵ model. First, the mean velocity field predictions for the incompressible κ-ϵ and RSM are quite distinct particularly in the virtual impaction region (Figure 5.2), which exhibits expansion and recirculation phenomena. The more robust Reynolds stress model predicts a recirculation vortex of a smaller length scale than that of the κ-ϵ model (Figure 6.6), and an expansion profile with a higher maximum velocity (Figure 6.10). This undoubtedly influences the particle motion, and thus the efficiency and wall loss curves. The compressibility effect, on the other hand, is observed as an increase in the flow velocities and pressure drop (from 0.3 to 0.4 bar across the nozzle). The mean flow structures, however, remained similar to those of the incompressible K-ϵ model (Section 6.1.2).

The motion of fluid elements in the incompressible Reynolds stress model simulation, and in the compressible K-ϵ model simulation showed that the residence time of fluid points entering the virtual impaction zone are comparable (Figure 6.38). Therefore, those two particular simulations were later used to study in detail the motion of discrete particles (Section 6.2.4).

7.2 Conclusions

The work in this thesis exposed the inadequacy of the stochastic K-ϵ model in capturing particle dispersion, simply because the turbulence isotropy assumption is too crude to produce any meaningful results. When the random walk technique was used with the Reynolds stress model however, consistent predictions for the efficiency and wall loss curves resulted. This was partly due to the non-isotropic nature of the normal stresses, and partly because the turbulent kinetic energy levels were much lower than those predicted by the K-ϵ model. The stochastic approach for modeling turbulent dispersion was not explored any further, and was not implemented in the developed particle tracking code, because of evidence in the literature which suggests that such schemes produce nonzero divergence velocity fields [50].

The characteristics of the physics involved in a particle-fluid motion were studied in this thesis. It was shown that particles with certain diameters (0.2 to 0.4 μm) experience a range of flow regimes that can significantly affect the mode of interaction between the particle and the fluid. Of particular interest is the magnitude of the slip velocity, which dictates the scales of the particle Reynolds and Mach numbers. It was found that the compressible form of the drag law, and the Stokes-Cunningham drag law with a pressure dependent slip correction factor, are more suitable for particle tracking calculations under the flow conditions of this virtual impactor.

The analysis of the performance of the virtual impactor, and the comparison with experimental data was done in the context of the compressible Ꝁ-ϵ model (simulation 5 in Table 5.1). This particular simulation, other than possessing the most accurate numerical components (fine grid, 2nd order discretization), gave the lowest 50% cutpoint diameter, approximately 0.25 μm (Figure 6.73). Despite the disagreement between this simulation's predictions and the reported experimental value of 0.12 μm (Figure 6.78), the simulation's efficiency and wall loss curves are in agreement with classical trends.

7.3 Recommendations

To eliminate any uncertainties about the turbulent flow field prediction, and to minimize modelling errors, a better approach than RANS, that will be pursued in the future is Large Eddy Simulation (LES). In LES, the three-dimensional time dependent details of the large scales of motion are directly resolved on the numerical grid by solving the filtered Navier-Stokes equations, while the small turbulent scales (sub-grid), which are typically believed to be independent of the overall flow geometry, are modeled. This approach has been proven to work well for predicting particle dispersion in one-way coupled flows [10,36].

Future work will investigate enhancements to the AeroTrack code, with particular focus on algorithms that reduce the computational cost, and perhaps parallelization. The algorithm proposed here, for applying the multi-variate scattered point interpolation scheme to a CFD simulation, requires further analysis of the most influencing parameters. This study was mainly concerned with accomplishing a working version rather than optimization.

Future work is needed to assess the validity of the classical particle equation of motion when the underlying dimensionless quantities (Re_p

and Ma_r) exceed the assumed limits. Furthermore, future work should be concerned with sharply resolving the flow structures near solid boundaries, and the inclusion of other dominant terms in the particle equation of motion, such as the lift force and Brownian diffusion.

Appendix A

Solution of the Particle ODE

The vector of unknowns in Equation (4.2.21) is taken as:

(A–1)

$$y = \left| \begin{array}{c} X_{1p} \\ U_{1p} \\ X_{2p} \\ U_{2p} \\ X_{3p} \\ U_{3p} \end{array} \right|$$

where (X_{1p}, X_{2p}, X_{3p}) and (U_{1p},U_{2p},U_{3p}) are the position and velocity vectors of a particle in the Cartesian coordinate system, respectively.

The system of ordinary differential equations becomes:

(A–2)

$$\frac{d\,y^{(1)}}{dt} = y^{(2)}$$

(A–3)

$$\frac{d\,y^{(2)}}{dt} = \frac{1}{\beta}(U_1 - y^{(2)})$$

(A–4)

$$\frac{d\,y^{(3)}}{dt} = y^{(4)}$$

(A–5)

$$\frac{d\,y^{(4)}}{dt} = \frac{1}{\beta}(U_2 - y^{(4)})$$

(A–6)

$$\frac{d\,y^{(5)}}{dt} = y^{(6)}$$

(A–7)

$$\frac{d\,y^{(6)}}{dt} = \frac{1}{\beta}(U_3 - y^{(6)})$$

Applying Equation (4.2.23) to each of the above differential equations, yields:

(A-8)

$$y_{n+1}^{(1)}-y_{n}^{(1)}-\frac{\Delta t}{2}\left[y_{n}^{(2)}+y_{n+1}^{(2)}\right]=0$$

(A-9)

$$y_{n+1}^{(2)}-y_{n}^{(2)}-\frac{\Delta t}{2}\left[\frac{1}{\beta}(U_1-y_{n}^{(2)})+\frac{1}{\beta}(U_1-y_{n+1}^{(2)})\right]=0$$

(A-10)

$$y_{n+1}^{(3)}-y_{n}^{(3)}-\frac{\Delta t}{2}\left[y_{n}^{(4)}+y_{n+1}^{(4)}\right]=0$$

(A-11)

$$y_{n+1}^{(4)}-y_{n}^{(4)}-\frac{\Delta t}{2}\left[\frac{1}{\beta}(U_2-y_{n}^{(4)})+\frac{1}{\beta}(U_2-y_{n+1}^{(4)})\right]=0$$

(A-12)

$$y_{n+1}^{(5)}-y_{n}^{(5)}-\frac{\Delta t}{2}\left[y_{n}^{(6)}+y_{n+1}^{(6)}\right]=0$$

(A-13)

$$y_{n+1}^{(6)}-y_{n}^{(6)}-\frac{\Delta t}{2}\left[\frac{1}{\beta}(U_3-y_{n}^{(6)})+\frac{1}{\beta}(U_3-y_{n+1}^{(6)})\right]=0$$

where the fluid velocity values U_1, U_2, and U_3 are only interpolated at the iteration step n. This is sufficiently accurate granted a small value for Δt is used. Therefore, the problem reduces to finding the root of the function:

(A-14)

$$F(y_{n+1})=0$$

Using the truncated Taylor series expansion, Newton's method iterates starting from an initial guess, y^{0}_{n+1}, taken to be the solution of the ODE from a Forward Euler scheme. The convergence criteria for Newton's method is met when either the L2-norm of the vector satisfies the tolerance condition: $|y_{n+1}| \leq \gamma |y^{0}_{n+1}|$ where $\gamma = 1.0e^{-6}$, or when the loop exceeds 32 iterations. The latter condition, it was observed, rarely occurred and the method converged in two iterations at each time step.

Finally, the Jacobian matrix was hard-coded, and can be easily shown to be as such:

$$J = \begin{vmatrix} 1 & -\Delta t/2 & 0 & 0 & 0 & 0 \\ 0 & 1+\Delta t/(2\beta) & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -\Delta t/2 & 0 & 0 \\ 0 & 0 & 0 & 1+\Delta t/(2\beta) & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -\Delta t/2 \\ 0 & 0 & 0 & 0 & 0 & 1+\Delta t/(2\beta) \end{vmatrix}$$

Appendix B

Solution of the Least Squares in 2D

The residual sum can be written as:

$$R^2 = \sum_{i=1, i \neq k}^{N} \omega_i(x_k, y_k) E_{ik}^2 \qquad \text{(B-1)}$$

where

$$E_{ik} = c_{k1}(x_i - x_k)^2 + c_{k2}(x_i - x_k)(y_i - y_k) + c_{k3}(y_i - y_k)^2 + c_{k4}(x_i - x_k) + c_{k5}(y_i - y_k) + f_k - f_i$$

The condition for R^2 to be a minimum is that its partial derivatives are zero (it is easily shown that the second derivatives are positive). Dropping the scripts on the summation sign, we write:

$$\frac{\partial R^2}{\partial c_{k1}} = 2 \sum \omega_i E_{ik} (x_i - x_k)^2 = 0 \qquad \text{(B-2)}$$

$$\frac{\partial R^2}{\partial c_{k2}} = 2 \sum \omega_i E_{ik} (x_i - x_k)(y_i - y_k) = 0 \qquad \text{(B-3)}$$

$$\frac{\partial R^2}{\partial c_{k3}} = 2 \sum \omega_i E_{ik} (y_i - y_k)^2 = 0 \qquad \text{(B-4)}$$

$$\frac{\partial R^2}{\partial c_{k4}} = 2 \sum \omega_i E_{ik} (x_i - x_k) = 0 \qquad \text{(B-5)}$$

$$\frac{\partial R^2}{\partial c_{k5}} = 2 \sum \omega_i E_{ik} (y_i - y_k) = 0 \qquad \text{(B-6)}$$

Substituting E_{ik} in Equations (B-2) to (B-6), we get:

$$\begin{aligned}&c_{k1}\sum \omega_i(x_i-x_k)^4+c_{k2}\sum \omega_i(x_i-x_k)^3(y_i-y_k)+c_{k3}\sum \omega_i(x_i-x_k)^2(y_i-y_k)^2\\&+c_{k4}\sum \omega_i(x_i-x_k)^3+c_{k5}\sum \omega_i(x_i-x_k)^2(y_i-y_k)=\sum \omega_i(f_i-f_k)(x_i-x_k)^2\end{aligned} \tag{B-7}$$

$$\begin{aligned}&c_{k1}\sum \omega_i(x_i-x_k)^3(y_i-y_k)+c_{k2}\sum \omega_i(x_i-x_k)^2(y_i-y_k)^2+c_{k3}\sum \omega_i(x_i-x_k)(y_i-y_k)^3\\&+c_{k4}\sum \omega_i(x_i-x_k)^2(y_i-y_k)+c_{k5}\sum \omega_i(x_i-x_k)(y_i-y_k)^2\\&=\sum \omega_i(f_i-f_k)(x_i-x_k)(y_i-y_k)\end{aligned} \tag{B-8}$$

$$\begin{aligned}&c_{k1}\sum \omega_i(x_i-x_k)^2(y_i-y_k)^2+c_{k2}\sum \omega_i(x_i-x_k)(y_i-y_k)^3+c_{k3}\sum \omega_i(y_i-y_k)^4\\&+c_{k4}\sum \omega_i(x_i-x_k)(y_i-y_k)^2+c_{k5}\sum \omega_i(y_i-y_k)^3=\sum \omega_i(f_i-f_k)(y_i-y_k)^2\end{aligned} \tag{B-9}$$

$$\begin{aligned}&c_{k1}\sum \omega_i(x_i-x_k)^3+c_{k2}\sum \omega_i(x_i-x_k)^2(y_i-y_k)+c_{k3}\sum \omega_i(x_i-x_k)(y_i-y_k)^2\\&+c_{k4}\sum \omega_i(x_i-x_k)^2+c_{k5}\sum \omega_i(x_i-x_k)(y_i-y_k)=\sum \omega_i(f_i-f_k)(x_i-x_k)\end{aligned} \tag{B-10}$$

$$\begin{aligned}&c_{k1}\sum \omega_i(x_i-x_k)^2(y_i-y_k)+c_{k2}\sum \omega_i(x_i-x_k)(y_i-y_k)^2+c_{k3}\sum \omega_i(y_i-y_k)^3\\&+c_{k4}\sum \omega_i(x_i-x_k)(y_i-y_k)+c_{k5}\sum \omega_i(y_i-y_k)^2=\sum \omega_i(f_i-f_k)(y_i-y_k)\end{aligned} \tag{B-11}$$

In matrix form:

$$Ac = b$$

where A is a 5x5 matrix whose rows are the coefficient summations for each of Equations (B-7) to (B-11), b is a column vector whose rows are the right hand sides of the equations, and c is the column vector:

$$c = \left|\begin{array}{c} c_{k1} \\ c_{k2} \\ c_{k3} \\ c_{k4} \\ c_{k5} \end{array}\right|$$

The solution to the linear system is obtained by Singular Value Decomposition using the LAPACK driver routine ***sgelss*** [45].

Appendix C

Solution of the Least Squares in 3D

The residual sum can be written as:

$$R^2 = \sum_{i=1, i \neq k}^{N} \omega_i(x_k, y_k, z_k) E_{ik}^2 \tag{C-1}$$

where

$$E_{ik} = c_{k1} dx^2 + c_{k2} dx\, dy + c_{k3} dy^2 + c_{k4} dx\, dz + c_{k5} dy\, dz + c_{k6} dz^2 + c_{k7} dx + c_{k8} dy + c_{k9} dz + f_k - f_i$$

such that $dx = x_i - x_k$, $dy = y_i - y_k$, $dz = z_i - z_k$

The condition for R^2 to be a minimum is that its partial derivatives are zero (it is easily shown that the second derivatives are positive).

Dropping the scripts on the summation sign, we write:

$$\frac{\partial R^2}{\partial c_{k1}} = 2 \sum \omega_i E_{ik} (x_i - x_k)^2 = 0 \tag{C-2}$$

$$\frac{\partial R^2}{\partial c_{k2}} = 2 \sum \omega_i E_{ik} (x_i - x_k)(y_i - y_k) = 0 \tag{C-3}$$

$$\frac{\partial R^2}{\partial c_{k3}} = 2 \sum \omega_i E_{ik} (y_i - y_k)^2 = 0 \tag{C-4}$$

$$\frac{\partial R^2}{\partial c_{k4}} = 2 \sum \omega_i E_{ik} (x_i - x_k)(z_i - z_k) = 0 \tag{C-5}$$

$$\frac{\partial R^2}{\partial c_{k5}} = 2 \sum \omega_i E_{ik} (y_i - y_k)(z_i - z_k) = 0 \tag{C-6}$$

$$\frac{\partial R^2}{\partial c_{k6}}=2\sum \omega_i E_{ik}(z_i-z_k)^2=0 \tag{C-7}$$

$$\frac{\partial R^2}{\partial c_{k7}}=2\sum \omega_i E_{ik}(x_i-x_k)=0 \tag{C-8}$$

$$\frac{\partial R^2}{\partial c_{k8}}=2\sum \omega_i E_{ik}(y_i-y_k)=0 \tag{C-9}$$

$$\frac{\partial R^2}{\partial c_{k9}}=2\sum \omega_i E_{ik}(z_i-z_k)=0 \tag{C-10}$$

Substituting E_{ik} in Equations (C-2) to (C-10), we get:

$$\begin{gathered} c_{k1}\sum \omega_i dx^4+c_{k2}\sum \omega_i dx^3 dy+c_{k3}\sum \omega_i dx^2 dy^2+c_{k4}\sum \omega_i dx^3 dz \\ +c_{k5}\sum \omega_i dx^2 dy\, dz+c_{k6}\sum \omega_i dx^2 dz^2+c_{k7}\sum \omega_i dx^3+c_{k8}\sum \omega_i dx^2 dy \\ +c_{k9}\sum \omega_i dx^2 dz=\sum \omega_i(f_i-f_k)dx^2 \end{gathered} \tag{C-11}$$

$$\begin{gathered} c_{k1}\sum \omega_i dx^3 dy+c_{k2}\sum \omega_i dx^2 dy^2+c_{k3}\sum \omega_i dx\, dy^3+c_{k4}\sum \omega_i dx^2 dy\, dz \\ +c_{k5}\sum \omega_i dx\, dy^2 dz+c_{k6}\sum \omega_i dx\, dy\, dz^2+c_{k7}\sum \omega_i dx^2 dy+c_{k8}\sum \omega_i dx\, dy^2 \\ +c_{k9}\sum \omega_i dx\, dy\, dz=\sum \omega_i(f_i-f_k)dx\, dy \end{gathered} \tag{C-12}$$

$$\begin{gathered} c_{k1}\sum \omega_i dx^2 dy^2+c_{k2}\sum \omega_i dx\, dy^3+c_{k3}\sum \omega_i dy^4+c_{k4}\sum \omega_i dx\, dy^2 dz \\ +c_{k5}\sum \omega_i dy^3 dz+c_{k6}\sum \omega_i dy^2 dz^2+c_{k7}\sum \omega_i dx\, dy^2+c_{k8}\sum \omega_i dy^3 \\ +c_{k9}\sum \omega_i dy^2 dz=\sum \omega_i(f_i-f_k)dy^2 \end{gathered} \tag{C-13}$$

$$\begin{gathered} c_{k1}\sum \omega_i dx^3 dz+c_{k2}\sum \omega_i dx^2 dy\, dz+c_{k3}\sum \omega_i dx\, dy^2 dz+c_{k4}\sum \omega_i dx^2 dz^2 \\ +c_{k5}\sum \omega_i dx\, dy\, dz^2+c_{k6}\sum \omega_i dx\, dz^3+c_{k7}\sum \omega_i dx^2 dz+c_{k8}\sum \omega_i dx\, dy\, dz \\ +c_{k9}\sum \omega_i dx\, dz^2=\sum \omega_i(f_i-f_k)dx\, dz \end{gathered} \tag{C-14}$$

$$\begin{gathered} c_{k1}\sum \omega_i dx^2 dy\, dz+c_{k2}\sum \omega_i dx\, dy^2 dz+c_{k3}\sum \omega_i dy^3 dz+c_{k4}\sum \omega_i dx\, dy\, dz^2 \\ +c_{k5}\sum \omega_i dy^2 dz^2+c_{k6}\sum \omega_i dy\, dz^3+c_{k7}\sum \omega_i dx\, dy\, dz+c_{k8}\sum \omega_i dy^2 dz \\ +c_{k9}\sum \omega_i dy\, dz^2=\sum \omega_i(f_i-f_k)dy\, dz \end{gathered} \tag{C-15}$$

$$c_{k1}\sum \omega_i dx^2 dz^2 + c_{k2}\sum \omega_i dx\,dy\,dz^2 + c_{k3}\sum \omega_i dy^2 dz^2 + c_{k4}\sum \omega_i dx\,dz^3 + c_{k5}\sum \omega_i dy\,dz^3 + c_{k6}\sum \omega_i dz^4 + c_{k7}\sum \omega_i dx\,dz^2 + c_{k8}\sum \omega_i dy\,dz^2 + c_{k9}\sum \omega_i dz^3 = \sum \omega_i (f_i - f_k) dz^2 \tag{C-16}$$

$$c_{k1}\sum \omega_i dx^3 + c_{k2}\sum \omega_i dx^2 dy + c_{k3}\sum \omega_i dx\,dy^2 + c_{k4}\sum \omega_i dx^2 dz + c_{k5}\sum \omega_i dx\,dy\,dz + c_{k6}\sum \omega_i dx\,dz^2 + c_{k7}\sum \omega_i dx^2 + c_{k8}\sum \omega_i dx\,dy + c_{k9}\sum \omega_i dx\,dz = \sum \omega_i (f_i - f_k) dx \tag{C-17}$$

$$c_{k1}\sum \omega_i dx^2 dy + c_{k2}\sum \omega_i dx\,dy^2 + c_{k3}\sum \omega_i dy^3 + c_{k4}\sum \omega_i dx\,dy\,dz + c_{k5}\sum \omega_i dy^2 dz + c_{k6}\sum \omega_i dy\,dz^2 + c_{k7}\sum \omega_i dx\,dy + c_{k8}\sum \omega_i dy^2 + c_{k9}\sum \omega_i dy\,dz = \sum \omega_i (f_i - f_k) dy \tag{C-18}$$

$$c_{k1}\sum \omega_i dx^2 dz + c_{k2}\sum \omega_i dx\,dy\,dz + c_{k3}\sum \omega_i dy^2 dz + c_{k4}\sum \omega_i dx\,dz^2 + c_{k5}\sum \omega_i dy\,dz^2 + c_{k6}\sum \omega_i dz^3 + c_{k7}\sum \omega_i dx\,dz + c_{k8}\sum \omega_i dy\,dz + c_{k9}\sum \omega_i dz^2 = \sum \omega_i (f_i - f_k) dz \tag{C-19}$$

In matrix form:

$$Ac = b$$

where A is a 9x9 matrix whose rows are the coefficient summations for each of Equations (C-11) to (C-19), b is a column vector whose rows are the right hand sides of the equations, and c is the column vector:

$$c = \begin{vmatrix} c_{k1} \\ c_{k2} \\ c_{k3} \\ c_{k4} \\ c_{k5} \\ c_{k6} \\ c_{k7} \\ c_{k8} \\ c_{k9} \end{vmatrix}$$

The solution to the linear system is obtained by Singular Value Decomposition using the LAPACK driver routine ***sgelss*** [45].

References

1. Crowe C. T., Sommerfeld M., and Tsuji Y., 1998, Multiphase Flows with Droplets and Particles, CRC Press.

2. Marple V. A., Rubow K. L., and Olson B. A., 1995, “Diesel Exhaust/Mine Dust Virtual Impactor Personal Aerosol Sampler: Design, Calibration and Field Evaluation”, Aerosol Science and Technology, **22**, 140-150.

3. Sioutas C., 1994, PhD. Dissertation, “Development of Virtual Impaction Technologies to Concentrate Fine Ambient Particles”, Harvard University, School of Public Health, Boston, MA.

4. Forney L. J., Ravenhall D. G., and Lee S. S., 1982, “Experimental and Theoretical Study of a Two-Dimensional Virtual Impactor”, Environmental Science and Technology, **16**, 492-497.

5. Barr E. B., Hoover M. D., Kanapilly G. M., Yeh H. C., and Rothenberg S. J., 1983, “Aerosol Concentrator: Design, Calibration and Use”, Aerosol Science and Technology, **2**, 437-442.

6. Marple V. A., and Chien C. M., 1980, “Virtual Impactors: A Theoretical Study”, Environmental Science and Technology, **14**, 976-984.

7. Durst F., Milojevic D., and Schonung B., 1984, “Eulerian and Lagrangian Predictions of Particulate Two-Phase Flows: A Numerical Study”, Applied Mathematical Modelling, **8**, 101-115.

8. Picart A., Berlemont A., and Gouesbet G., 1986, “Modelling and Predicting Turbulence Fields and the Dispersion of Discrete Particles Transported by Turbulent Flows”, International Journal of Multiphase Flow, **12**, 237-261.

9. Berlemont A., Desjonqueres P., and Gouesbet G., 1990, “Particle Lagrangian Simulation in Turbulent Flows”, International Journal of Multiphase Flow, **16**, 19-34.

10. Elghobashi S., 1994, “On Predicting Particle-Laden Turbulent Flows”, Applied Scientific Research, **52**, 309-329.

11. Crowe C. T., Troutt T. R., and Chung J. N., 1996, “Numerical Models for Two-Phase Turbulent Flows”, Annual Review of Fluid Mechanics, **28**, 11-43.

12. Stock D. E., 1996, “Particle Dispersion in Flowing Gases - 1994 Freeman Scholar Lecture”, Journal of Fluids Engineering, **118**, 4-17.

13. Kim I., Elghobashi S., and Sirignano W. A., 1998, “On The Equation of Spherical-Particle Motion: Effect of Reynolds and Acceleration Numbers”, Journal of Fluid Mechanics, **367**, 221-253.

14. Chen B. T., and Yeh H. C., 1987, “An Improved Virtual Impactor: Design and Performance”, Journal of Aerosol Science, **18**, 203-214.

15. Chen B. T., Yeh H. C., and Cheng Y. S., 1985, “A Novel Virtual Impactor: Calibration and Use”, Journal of Aerosol Science, **16**, 343-354.

16. McFarland A. R., Ortiz C. A., and Bertch Jr. R. W., 1978, “Particle Collection Characteristics of a Single-Stage Dichotomous Sampler”, Environmental Science and Technology, **12**, 679-682.

17. Rader D. J., and Marple V. A., 1985, “Effect of Ultra-Stokesian Drag and Particle Interception on Impaction Characteristics”, Aerosol Science and Technology, **4**, 141-156.

18. Asgharian B., and Godo M. N., 1997, “Transport and Deposition of Spherical Particles and Fibers in an Improved Virtual Impactor”, Aerosol Science and Technology, **27**, 499-506.

19. Masuda H., Hochrainer D., and Stober W., 1979, “An Improved Virtual Impactor for Particle Classification and Generation of Test Aerosols with Narrow Size Distributions”, Journal of Aerosol Science, **10**, 275-287.

20. FLUENT^{TM} 6.0 Reference Manual, 2001, FLUENT INC.

21. Panton R. L., 1996, Incompressible Flow, 2nd Edition, John Wiley & Sons.

22. Bernard P. S., and Wallace J. M., 2002, Turbulent Flow: Analysis, Measurement and Prediction, John Wiley & Sons.

23. Maxey M. R., and Riley J. J., 1983, “Equation of Motion for a Small Rigid Sphere in a Nonuniform Flow”, Physics of Fluids, **26**, 883-889.

24. McLaughlin J. B., 1994, “Numerical Computation of Particles-Turbulence Interaction”, International Journal of Multiphase Flow, **20** supplement, 211-232.

25. Ling W., Chung J. N., Troutt T. R., and Crowe C. T., 1998, “Direct Numerical Simulation of a Three-Dimensional Temporal Mixing Layer with Particle Dispersion”, Journal of Fluid Mechanics, **358**, 61-85.

26. Ma L., Ingham D. B., and Wen X., 2000, “Numerical Modelling of the Fluid and Particle Penetration Through Small Sampling Cyclones”, Journal of Aerosol Science, **31**, 1097-1119.

27. McLaughlin J. B., 1989, “Aerosol Particle Deposition in Numerically Simulated Channel Flow”, Physics of Fluids, **1**, 1211-1224.

28. Clift R., Grace J. R., and Weber M. E., 1978, Bubbles, Drops, and Particles, Academic Press.

29. Hinds W. C., 1982, Aerosol Technology: Properties, Behavior, and Measurement of Airborne Particles, John Wiley & Sons.

30. Crowe C. T., 1967, “Drag Coefficient of Particles in a Rocket Nozzle”, AIAA Journal, **5**, 1021-1022.

31. Crowe C. T., Babcock W. R., and Willoughby P. G., 1973, “Drag Coefficient for Particles in Rarefied, Low Mach-Number Flows”, Progress in Heat and Mass Transfer, **6**, 419-431.

32. Gosman A. D., and Ioannides E., 1981, “Aspects of Computer Simulation of Liquid-Fuelled Combustors”, AIAA Paper 81-0323.

33. Wang Q., and Squires K. D., 1996, “Large Eddy Simulation of Particle-Laden Turbulent Channel Flow”, Physics of Fluids, **8**, 1207-1223.

34. Burton T. M., and Eaton J. K., 2002, “High-Resolution Simulations of Particle-Eddy Interactions”, Powder Technology, **125**, 104-110.

35. Kontomaris K., Hanratty T. J., and McLaughlin J. B., 1992, “An Algorithm for Tracking Fluid Particles in a Spectral Simulation of Turbulent Channel Flow”, Journal of Computational Physics, **103**, 231-242.

36. Armenio V., Piomelli U., and Fiorotto V., 1999, “Effect of the Subgrid Scales on Particle Motion”, Physics of Fluids, **11**, 3030-3042.

37. Ahmed A. M., and Elghobashi S., 2001, “Direct Numerical Simulation of Particle Dispersion in Homogeneous Turbulent Shear Flows”, Physics of Fluids, **13**, 3346-3364.

38. Gasca M., and Sauer T., 2000, “Polynomial Interpolation in Several Variables”, Advances in Computational Mathematics, **12**, 377-410.

39. Derksen J. J., 2003, “Separation Performance Predictions of a Stairmand High-Efficiency Cyclone”, AIChE Journal, **49**, 1359-1371.

40. Derksen J. J., 2003, “Numerical Simulation of Solids Suspension in a Stirred Tank”, AIChE Journal, **49**, 2700-2714.

41. Shepard D., 1968, “A Two-Dimensional Interpolation Function for Irregularly Spaced Data”, Proceedings of the 23rd ACM National Conference, 517-524.

42. Kincaid D., and Cheney W., 1991, Numerical Analysis: Mathematics of Scientific Computing, Brooks/Cole Publishing Company.

43. Franke R., and Nielson G., 1980, “Smooth Interpolation of Large Sets of Scattered Data”, International Journal for Numerical Methods in Engineering, **15**, 1691-1704.

44. Renka R. J., 1988, “Multivariate Interpolation of Large Sets of Scattered Data”, ACM Transactions on Mathematical Software, **14**, 139-148.

45. INTEL® Math Kernel Library Reference Manual, 2003, developer.intel.com

46. Bentley J. L., Weide B. W., and Yao A. C., 1980, “Optimal Expected-Time Algorithms for Closest Point Problems”, ACM Transactions on Mathematical Software, **6**, 563-580.

47. Renka R. J., and Brown R., 1999, “Algorithm 792: Accuracy Tests of ACM Algorithms for Interpolation of Scattered Data in the Plane”, ACM Transactions on Mathematical Software, **25**, 78-94.

48. Dr. M. B. (Arun) Ranade, 2003, ***Personal Communication***.

49. Ding Y., and Koutrakis P., 2000, "Development of a Dichotomous Slit Nozzle Virtual Impactor", Journal of Aerosol Science, **31**, 1421-1431.

50. MacInnes J. M., and Bracco F. V., 1992, "Stochastic Particle Dispersion Modeling and the Tracer-Particle Limit", Physics of Fluids, **4**, 2809-2824.

51. Sioutas C., Koutrakis P., and Burton R. M., "A High-Volume Small Cutpoint Virtual Impactor For Separation of Atmospheric Particulate From Gaseous Pollutants", Particulate Science and Technology, **12**, 207-221.

52. Constantinides A., and Mostoufi N., 1999, Numerical Methods for Chemical Engineers with MATLAB Applications, Prentice Hall PTR.