COMPUTATIONAL FLUID DYNAMICS (CFD) SIMULATIONS OF AEROSOL IN A U-SHAPED STEAM GENERATOR TUBE

A Dissertation

by

PAMELA LONGMIRE

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2007

Major Subject: Interdisciplinary Engineering

COMPUTATIONAL FLUID DYNAMICS (CFD) SIMULATIONS OF AEROSOL IN A U-SHAPED STEAM GENERATOR TUBE

A Dissertation

by

PAMELA LONGMIRE

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Co-Chairs of Committee,	Yassin A. Hassan
	William H. Marlow
Committee Members,	Karan L. Watson
	Sherry J. Yennello
Head of Department,	Nagamangala K. Anand

May 2007

Major Subject: Interdisciplinary Engineering

ABSTRACT

Computational Fluid Dynamics (CFD) Simulations of Aerosol in a

U-Shaped Steam Generator Tube.

(May 2007)

Pamela Longmire, B.S., University of Cincinnati; M.S., University of Akron;

M.S., The Ohio State University

Co-Chairs of Advisory Committee: Dr. Yassin A. Hassan
Dr. William H. Marlow

To quantify primary side aerosol retention, an Eulerian/Lagrangian approach was used to investigate aerosol transport in a compressible, turbulent, adiabatic, internal, wall-bounded flow. The ARTIST experimental project (Phase I) served as the physical model replicated for numerical simulation. Realizable k-ε and standard k-ω turbulence models were selected from the computational fluid dynamics (CFD) code, FLUENT, to provide the Eulerian description of the gaseous phase.

Flow field simulation results exhibited: a) onset of weak secondary flow accelerated at bend entrance towards the inner wall; b) flow separation zone development on the convex wall that persisted from the point of onset; c) centrifugal force concentrated high velocity flow in the direction of the concave wall; d) formation of vortices throughout the flow domain resulted from rotational (Dean-type) flow; e) weakened secondary flow assisted the formation of twin vortices in the outflow cross section; and f) perturbations induced by the bend influenced flow recovery several pipe

diameters upstream of the bend. These observations were consistent with those of previous investigators.

The Lagrangian discrete random walk model, with and without turbulent dispersion, simulated the dispersed phase behavior, incorrectly. Accurate deposition predictions in wall-bounded flow require modification of the Eddy Impaction Model (EIM). Thus, to circumvent shortcomings of the EIM, the Lagrangian time scale was changed to a wall function and the root-mean-square (RMS) fluctuating velocities were modified to account for the strong anisotropic nature of flow in the immediate vicinity of the wall (boundary layer). Subsequent computed trajectories suggest a precision that ranges from 0.1% to 0.7%, statistical sampling error. The aerodynamic mass median diameter (AMMD) at the inlet (5.5 μm) was consistent with the ARTIST experimental findings. The geometric standard deviation (GSD) varied depending on the scenario evaluated but ranged from 1.61 to 3.2. At the outlet, the computed AMMD (1.9 μm) had GSD between 1.12 and 2.76. Decontamination factors (DF), computed based on deposition from trajectory calculations, were just over 3.5 for the bend and 4.4 at the outlet. Computed DFs were consistent with expert elicitation cited in NUREG-1150 for aerosol retention in steam generators.

DEDICATION

I dedicate this dissertation to the people that were most instrumental in making me the person I am, my parents and grandparents: *William and Thelma Cody*; *James E. Longmire*; *UL and A.M. Lauderdale*; *Johnnie and Minerva Longmire*; and *William and Ethelene Cody*. I am grateful my mother and grandmother are alive to share with me in this, my latest accomplishment. The others will always be with me in spirit as I celebrate the courage, endurance, and temperance they demonstrated during their stay here on this earth. I honor them here for they are the people that taught me "I am the master of my fate; I am the captain of my soul."

Out of the night that covers me,
Black as the Pit from pole to pole,
I thank whatever gods may be
For my unconquerable soul.

In the fell clutch of circumstance
I have not winced nor cried aloud.
Under the bludgeonings of chance
My head is bloody, but unbowed.

Beyond this place of wrath and tears
Looms but the Horror of the shade,
And yet the menace of the years
Finds, and shall find me, unafraid.

It matters not how strait the gate,
How charged with punishments the scroll,
I am the master of my fate:
I am the captain of my soul.

Invictus
William Ernest Henley, 1875

ACKNOWLEDGMENTS

Writing a dissertation can be a lonely and isolating experience, yet it is obviously not possible without the personal and practical support of numerous people. Thus, I extend my sincerest gratitude to one of Sandia National Laboratories' distinguished Senior Scientist and member of the Nuclear Regulatory Commission's Advisory Committee on Reactor Safeguards, Dana A. Powers. Without his guidance and encouragement, completion of the research and writing of this dissertation would not have been possible. I am particularly appreciative of his willingness to offer his time in both discussion of the various topics concerning this research effort and for providing expansive commentary on every draft of this document. His comments were insightful and provoked deeper contemplation. I am also grateful to Randy Gauntt (Manager, Modeling and Analysis Department) for all the support I received while researching and writing this dissertation and to the U.S. Nuclear Regulatory Commission, Office of Nuclear Regulatory Research, Division of Systems Analysis and Regulatory Effectiveness, Safety Margins and Systems Analysis Branch, who supported this research.

The members of my dissertation committee: Yassin Hassan (Professor and Associate Department Head, Nuclear Engineering), William (Bill) Marlow (Professor and Undergraduate Advisor), Karan Watson (Dean of Faculties and Associate Provost), and Sherry Yennello (Professor and Associate Dean), have generously given their time and encouragement. Both of my co-chairs have given so graciously of their expertise to better my work. I thank them for their contribution and their good-natured support.

TABLE OF CONTENTS

Page

ABSTRACT iii

DEDICATION v

ACKNOWLEDGMENTS vi

TABLE OF CONTENTS vii

LIST OF FIGURES x

LIST OF TABLES xii

NOMENCLATURE xiii

CHAPTER

I INTRODUCTION 1

II MATHEMATICAL MODELS 19

- Compressible Flow 19
- Governing Equations of Fluid Flow 21
  - Mass Conservation 24
  - Momentum Equation 26
  - Energy Equation 32
  - Equation of State 35
  - Auxiliary Conditions 36
- Modeling Approach 37
  - Eulerian Approach for the Continuous Phase 38
    - Turbulence Models 40
      - Standard k-ε Model 44
      - Realizable k-ε Model (rke) 48
      - Standard k-ω Model 49
  - Near-Wall Modeling 50
    - Law of the Walls 51
    - Non-Equilibrium Wall Function 54
    - Enhanced Wall Treatment 56
  - Lagrangian Approach for the Discrete Phase 58
    - Equation of Motion for Particles 59

CHAPTER Page

Integration of the Trajectory Equations 63
Default Discrete Random Walk Model (DRW) 64
Modification to the Dispersed Phase Model (DPM) 66
Lagrangian Time Scale & RMS Fluctuating Velocities 67
Spurious Drift 70

III SOLUTION ALGORITHM 74

Computational Domain 74
Turbulence Models 78
Compressible Flow Setup 80
Numeric Scheme 81
Continuous Phase (Flow Field) 84
Discrete Phase Model (DPM) 85

IV RESULTS AND DISCUSSION 88

ARTIST Experiment: Phase I, Single U-tube 88
Properties of the Working Fluid: Nitrogen 90
Characteristics of the Flow Field 91
Fully-Developed Flow 94
Turbulence Structures: Estimate of Scales 97
Flow Field Results 99
Velocity 99
Vorticity 114
Helicity 122
Turbulence Intensity 125
Summary of Flow Field Results 130
Particle Trajectory & Retention 132
Discrete Phase Model: Settings & Options 132
Comparison of Numerical Results with Correlations 135
Comparison of Numerical Results with Experiment 141
Decontamination Factor 143
Accuracy/Precision 146

V CONCLUSIONS 148

REFERENCES 152

APPENDIX A 158

APPENDIX B 185

Page

VITA……..……..…….207

LIST OF FIGURES

Page

Figure 1. Cut-Away View of a Typical Recirculating Steam Generator Taken From NUREG/CR-6365....3
Figure 2. Stress Corrosion Cracking, Tube Material: Inconel 4
Figure 3. ARTIST SG: Photo and Schematic Representation of the Test Section 8
Figure 4. Dean-Type Secondary Flow 13
Figure 5. Conservation Laws: Fluid Element....23
Figure 6. Fluid Element: Mass Flows at Boundaries 24
Figure 7. Fluid Element: Stress Components....27
Figure 8. Stress Components (*x*-Direction)....28
Figure 9. Heat Flux Vector....34
Figure 10. Boundary Conditions: Internal Flow 37
Figure 11. Turbulence Models 41
Figure 12. Boundary Layer 52
Figure 13. Flow Behavior in Near-Wall Regions 52
Figure 14. Near-Wall Modeling Approaches....53
Figure 15. Non-Equilibrium Wall Function: Turbulence Kinetic Energy Budget....56
Figure 16. Turbulence Model: Grid 77
Figure 17. Entrance Length and Boundary Layer Growth....96
Figure 18. Contours of Velocity Magnitude around the Pipe (m/s)....102
Figure 19. Velocity Magnitude, Inlet....104
Figure 20. Velocity Magnitude, Downstream (Inlet-Side) of Bend....104
Figure 21. Velocity Magnitude, Upstream (Outlet-Side) of Bend....105
Figure 22. Velocity Magnitude, Outlet 105
Figure 23. Plane Placement in the U-Bend 107
Figure 24. Velocity Magnitude, Bend....108
Figure 25. Velocity Contours Superimposed on Velocity Vectors at 0° 109

Page

Figure 26. Velocity Contours Superimposed on Velocity Vectors at 45° 110
Figure 27. Velocity Contours Superimposed on Velocity Vectors at 90° 111
Figure 28. Velocity Contours Superimposed on Velocity Vectors at 135° 112
Figure 29. Velocity Contours Superimposed on Velocity Vectors at 180° 113
Figure 30. Vorticity Magnitude, Downstream (Inlet-Side) of Bend 115
Figure 31. Vorticity Magnitude, Bend 116
Figure 32. Vorticity Magnitude, Upstream (Outlet-Side) of Bend 116
Figure 33. Velocity Contours Superimposed on Vorticity Vectors at 0° 117
Figure 34. Velocity Contours Superimposed on Vorticity Vectors at 45° 118
Figure 35. Velocity Contours Superimposed on Vorticity Vectors at 90° 119
Figure 36. Velocity Contours Superimposed on Vorticity Vectors at 135° 120
Figure 37. Velocity Contours Superimposed on Vorticity Vectors at 180° 121
Figure 38. Helicity, Bend 123
Figure 39. Helicity, Downstream (Inlet-Side) of Bend 124
Figure 40. Helicity, Upstream (Outlet-Side) of Bend 125
Figure 41. Turbulence Intensity, Inlet 127
Figure 42. Turbulence Intensity, Downstream (Inlet-Side) of Bend 127
Figure 43. Turbulence Intensity, Bend 128
Figure 44. Turbulence Intensity at Bend Outlet (180°) 129
Figure 45. Turbulence Intensity, Upstream (Outlet-Side) of Bend 129
Figure 46. Turbulence Intensity, Outlet 130
Figure 47. Contours of Static Pressure (Pa) on Bend Wall and Longitudinal Velocity (m/s) Profile 132
Figure 48. Deposition Efficiency Correlations 137
Figure 49. Deposition Efficiency: Computation versus Correlation 139
Figure 50. Pressure Drop Across 600 mm of Pipe Including the Bend 143
Figure 51. Accuracy as a Function of Number of Particles Sampled 147

LIST OF TABLES

Page

Table 1. Past SGTR Accidents at Pressurizer Water Reactors 4
Table 2. Essential Features of the Transient Sequence Used in the ARTIST Project..... 7
Table 3. Comparison of Parameters of Interest..... 15
Table 4. The Governing Equations of the Flow of a Compressible Newtonian Fluid.... 22
Table 5. RANS Turbulence Model Behavior and Usage 42
Table 6. Establishing Placement of First Grid Point..... 76
Table 7. Boundary Conditions for the Realizable k-ε Model..... 79
Table 8. Boundary Conditions for k-ω Model 79
Table 9. Dispersed Phase Model (DPM) Input 87
Table 10. Mean Values for Single U-Tube Test in ARTIST Experiment..... 90
Table 11. Property of N_2 (g) from NIST Standard Reference Database Software..... 92
Table 12. Model Selection Criteria 94
Table 13. Trends in Compressible Subsonic Pipe Flow of Perfect Gases[a] 94
Table 14. Equations for Computing Scales of Turbulence 98
Table 15. Computed Scales of Turbulence 98
Table 16. Flow Field Results..... 101
Table 17. Discrete Phase Model Panel Input 134
Table 18. Stokes Number for ARTIST 139
Table 19. Deposition Efficiency ($0.2 \leq Stk \leq 0.5$) 140
Table 20. Particle Retention in the Bend..... 142
Table 21. Decontamination Factors 144
Table 22. Computed AMMD, GSD, and Decontamination Factor..... 146

NOMENCLATURE

Greek Symbols

α	Damping Coefficient
α^*	Damping Coefficient (for turbulent viscosity, low-Re correction)
α_∞	Damping Coefficient (for turbulent viscosity, high-Re correction)
β	Volume Fraction
χ	Shape Factor
δ	Bend Curvature Ratio ($=R_b/r_p$)
δ_{ij}	Kronecker Delta
ε	Turbulent Dissipation Rate
ϕ	Scalar Quantity (Arbitrary)
γ	Ratio of Specific Heats
η	Transport Efficiency
ϑ	Velocity Scale (Large-Scale Turbulence)
φ	Vorticity
κ	von Karman Constant
λ	Mean Free Path
Λ	Viscosity (relating stresses to volumetric deformation)
θ	Bend Angle (radians)
μ	Dynamic Viscosity
μ_t	Turbulent Eddy Viscosity
ν	Kinematic Viscosity
ρ	Fluid Density
ρ_p	Particle Density
σ_ε	Prandtl Number (for turbulent dissipation rate)
σ_k	Prandtl Number (for turbulent kinetic energy)
τ_e	Eddy Lifetime
τ_{cross}	Eddy Crossing Time
τ_{ij}	Viscous Stress Component
τ_w	Wall Shear Stress
τ_+	Dimensionless Particle Relaxation Time
ω	Specific Turbulent Dissipation Rate
ξ	Turbulent Time Scale
ψ	Gaussian Random Number
ζ	Normally Distributed Random Number

Lower Case Symbols

c	Speed of Sound
c_p	Specific Heat at Constant Pressure
c_v	Specific Heat at Constant Volume
e_{ij}	Rate of Linear Deformation
d	Internal Pipe Diameter
d_g	Geometric Mean Diameter
d_p	Particle Diameter
k	Turbulent Kinetic Energy
k_B	Boltzmann's Constant
k_c	Molecular Heat Conductivity
l	Length Scale (Large-Scale Turbulence)
n	Local Coordinate Normal to the Wall
n_i	Number of Particles in Group i
p	Static Pressure
p_0	Stagnation Pressure
p_{op}	Operating Pressure
q	Heat Flux
r_p	Internal Pipe Radius
s	Surface Area of Sphere (relative to particle volume)
t	Time
$\overline{u}$	Velocity Vector
$\overline{u}_i$	Mean Component of Velocity
u_i'	Fluctuation Component of Velocity
u_p	Particle Velocity
u^*	Friction Velocity
u, v, w	Velocity Components (Fluid)
x, y, z	Cartesian Coordinates

Upper Case Symbols

AMG	Algebraic Multigrid
AAMG	Aggregative Algebraic Multigrid
BC	Boundary Condition
C_c	Cunningham Correction Factor
C_D	Drag Coefficient
De	Dean Number
D/Dt	Time Dependent Total Derivative
E	Energy
E_D	Deposition Efficiency
F_D	Drag Force
H	Helicity
I	Turbulence Intensity
Kn	Knudsen Number
M	Mach Number
M_w	Molecular Weight
N	Total Number of Particles
P	Penetration
Q	Volumetric Flow Rate
R	Universal Gas Constant
R_b	Bend Radius of Curvature
Re	Reynolds Number
S	Particle Surface Area
S_E, S_M,	Source Term (for energy & momentum equation, respectively)
T	Temperature
T_0	Stagnation Temperature
T_L	Lagrangian Integral Time
U_{ref}	Reference Velocity
V_t	Deposition Velocity

Acronyms & Abbreviations

AMMD	Aerodynamic Mass Median Diameter
ARTIST	AeRosol Trapping In a Steam generaTor
CFD	Computational Fluid Dynamics
DBA	Design Basis Accident
DF	Decontamination Factor
DNS	Direct Numerical Simulation
DPM	Dispersed Phase Model
DRW	Discrete Random Walk
EIM	Eddy Impaction Model
GSD	Geometric Standard Deviation
LES	Large Eddy Simulation
LES-NWM	Large Eddy Simulation Near Wall Modeled
LES-NWR	Large Eddy Simulation Near Wall Resolution
NPP	Nuclear Power Plant
NRC	Nuclear Regulatory Commission
NSSS	Nuclear Steam Supply System
ODSCC	Outside Diameter Stress Corrosion Cracking
PSI	Paul Scherrer Institute
PWR	Pressurized Water Reactor
PWSCC	Primary Water Stress Corrosion Cracking
RANS	Reynolds Averaged Navier-Stokes
RCS	Reactor Coolant System
rke	Realizable k-ε
RMS	Root-Mean-Square
RSM	Reynolds Stress Model
RSS	Reactor Safety Study
SG	Steam Generator
SGTR	Steam Generator Tube Rupture
skw	Standard k-ω
SRV	Safety Relief Valve
Stk	Stokes Number

CHAPTER I

INTRODUCTION

Pressurized water reactors (PWRs) use steam generators, very large components that convert water into steam from heat produced in the reactor core. These devices can weigh as much as 800 tons. Inside steam generators, thermally hot radioactive water is pumped through thousands of feet of tubing. Thus, a steam generator can contain anywhere from 3,000 to 16,000 tubes, each about three-quarters of an inch (~19 mm) in diameter which are under high pressure to prevent boiling. The thermally hot radioactive water (i.e., contains fission products) flowing inside of the tubes heats non-radioactive water on the outside of the tubes. This transfer of heat produces steam that turns the blades of turbines to generate electricity.

Two types of steam generators used in the conversion of water to steam in the PWR-type nuclear power plants (NPP) are: a) recirculating steam generators and b) once-through steam generators. The recirculating steam generator is of primary interest in this study. Figure 1, for the purpose of illustration, shows the U-shaped tubes in a cut-away view of a typical recirculating steam generator. In addition to other components of the nuclear steam supply system (NSSS), steam generators (SGs) are housed inside the reactor containment, a structure that normally acts as a final barrier to fission product release to the environment.

This dissertation follows the style and format of *Nuclear Technology*.

If a tube ruptures during reactor operations radioactivity from the primary coolant system, the system that pumps water through the reactor core, could escape directly to the atmosphere. Consideration of steam generator tube rupture (SGTR) accidents in the safety evaluation of PWRs is an area of active interest given that steam generator tube defects and failures occur with some frequency. Worldwide, there have been eleven SGTR events in PWRs. Table 1 list NPPs where SGTR events have occurred, the date, leakage rate of radioactive material and the cause of the SGTR. NUREG/CR-6365 reports the first SGTR event occurred in 1975 at Point Beach Unit 1, while the most recent event occurred in 2000 at Indian Point Unit 2.

Additionally, Table 1 lists the particular stressors that precipitate the degradation mechanisms (i.e., causes): fretting (wear), high cycle fatigue, outside diameter stress corrosion cracking (ODSCC), primary water stress corrosion cracking (PWSCC), and wastage. Inconel 600 mill-annealed,[1] a thin nickel alloy material, used in original SGs has proven susceptible to many forms of cracking, pitting, denting, and other types of degradation. Figure 2 shows intergranular cracking in Inconel. This type of stress corrosion cracking requires at least the presence of high applied or residual tensile stress or both (near the yield strength), corrosive environment (e.g., high temperature water), and susceptible tubing microstructure (e.g., alloy content and few intergranular carbides). All, if not most, of the aforementioned conditions (i.e., high stress, high temperature, and susceptible tubing microstructure) are present in an operating steam generator. Since 1983, Inconel 690 TT has often been specified as a replacement for

Inconel 600.[2] In 1993, the NRC found "no end in sight" to steam generator tube cracking problems at plants operating with original steam generators.[3]

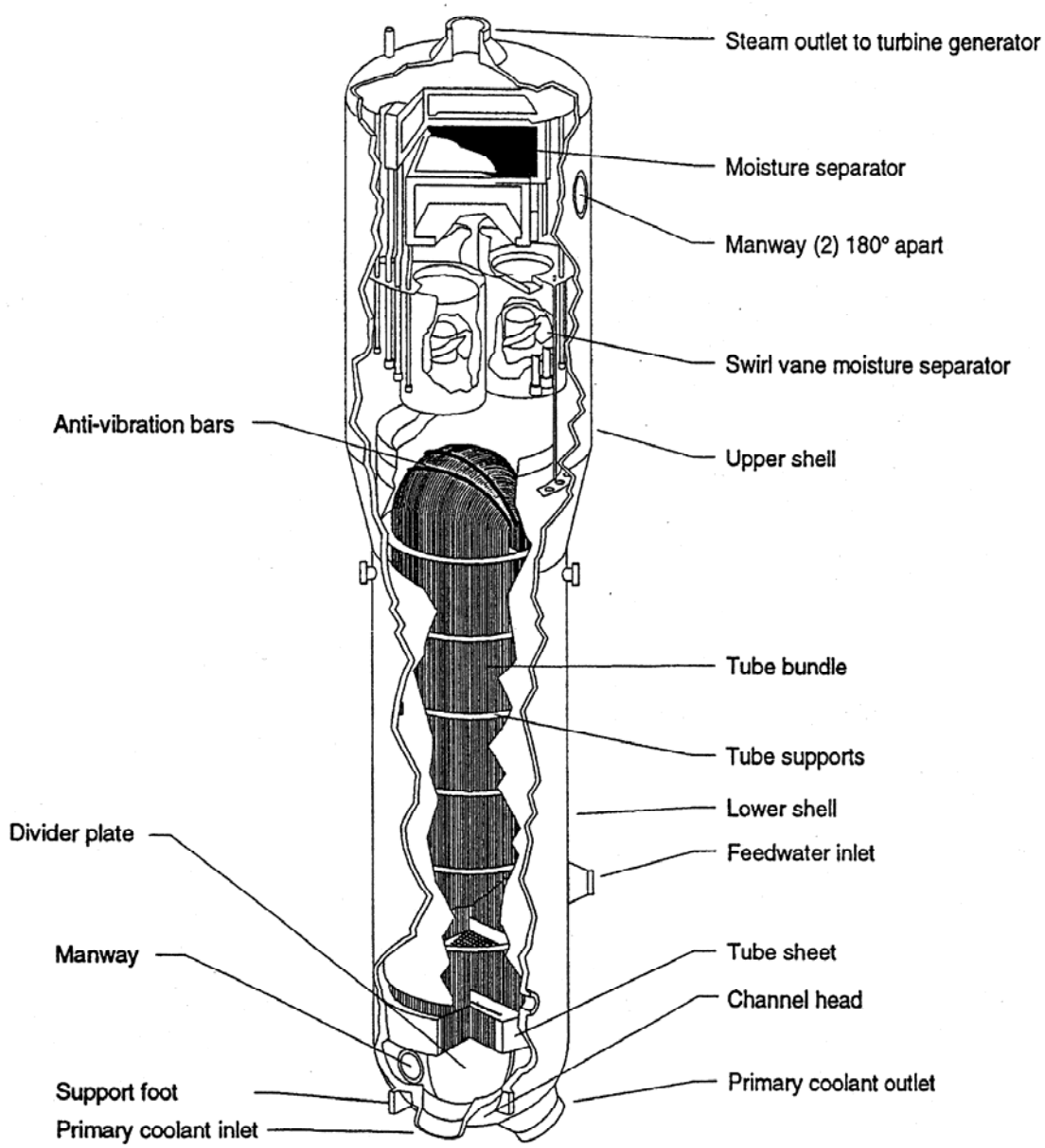


Figure 1. Cut-Away View of a Typical Recirculating Steam Generator Taken From NUREG/CR-6365[4]

Table 1. Past SGTR Accidents at Pressurizer Water Reactors

Plant	Date	Leak Rate*	Cause
Point Beach Unit 1	February 26, 1975	125	Wastage
Surry Unit 2	September 15, 1976	330	PWSCC in U-bend
Doel Unit 2	June 25, 1979	135	PWSCC in U-bend
Prairie Island 1	October 2, 1979	390	Loose parts
Ginna Unit 1	January 25, 1982	760	Loose parts and tube wear
Fort Calhoun	May 16, 1984	112	ODSCC at a crevice
North Anna Unit 1	July 15, 1987	637	High cycle fatigue in a U-bend
McGuire Unit 1	March 7, 1989	500	ODSCC in the free span
Mihama Unit 2	February 9, 1991	700	High cycle fatigue
Palo Verde Unit 2	March 14, 1993	240	ODSCC
Indian Point Unit 2	February 15, 2000	150	PWSCC in U-bend

*Leakage rate in gallons per minutes (gpm)
Legend for degradation mechanisms (i.e., causes column) identified below by stressor:
Fretting, Wear – flow induced vibration, aggressive chemicals
High cycle fatigue – high mean stress level and flow induced vibration, initiating defect (crack, dent, pit, etc.)
ODSCC – tensile stresses, impurity concentrations, and sensitive materials
PWSCC – temperature, residual tensile stresses, sensitive materials (low mill anneal temperature)
Wastage – phosphate chemistry, chloride concentration, resin leakage

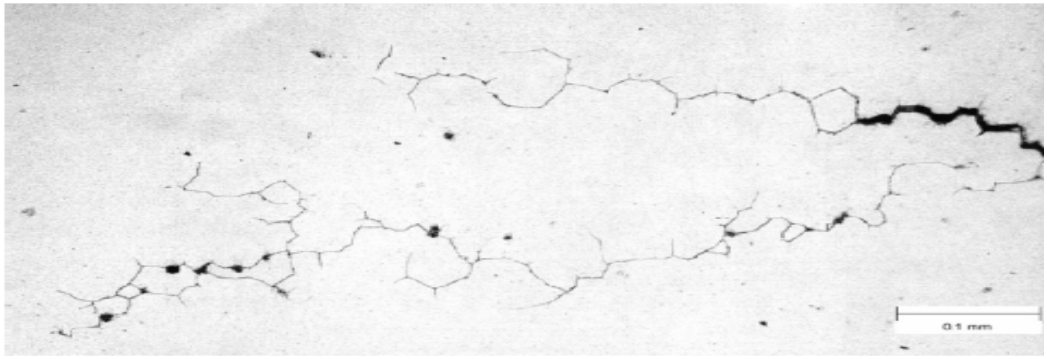


Figure 2. Stress Corrosion Cracking, Tube Material: Inconel

SGTRs are design basis accidents (DBA, a postulated accident that a nuclear facility must be designed and built to withstand without loss of systems, structures, and components necessary to assure public health and safety).[5] Probabilistic risk assessments (e.g., NUREG-1150[6] and NUREG-1560[7]) account for beyond DBA. In this instance, the types of accidents of interest are initiated by SGTRs accompanied by failures of additional systems or human errors. This class of reactor accident has the peculiar feature of being risk dominant, although the accidents are not frequency dominant. That is, the consequences of a SGTR accident progressing to core damage are severe, even though the expected frequencies are low. Severity of the accident consequences arises because gases laden with radioactive aerosol pass from the primary coolant system through the secondary system and out a safety relief valve (SRV), which is located outside of the reactor containment. Consequently, natural processes and engineered safety features within the containment do not have the opportunity to mitigate the potential release of radioactive material to the environment. "Containment bypass accident" is the name given to this type of accident sequence. Source terms (i.e., fractions defining the portion of the radionuclide inventory in the reactor at the start of an accident that is released to the environment) for bypass sequences (such as accidents initiated by SGTR) can be quite large. The magnitude of the source term from an SGTR accident depends on the integrity of the secondary system (i.e., steam generator tubes, steam turbine, condenser, and associated pipes, pumps, and heaters used to convert the heat energy of the reactor coolant system into mechanical energy for electrical

generation) and the containment. Releases may be quite small if both secondary systems and containment integrity are maintained.

It is undeniable that steam generator tube integrity is important to reactor safety because the tubes constitute one of the primary barriers between the radioactive and non-radioactive sides of PWR NPPs. It is a fact that steam generators constitute more than 50% of the surface area of the primary pressure boundary in a PWR. Inside the steam generator, the internal components: shroud, tube-sheet, support plates, bundle of tubes themselves, separator, dryer, etc. provide ample surface area for deposition (retention) of radioactive aerosols. In NUREG-1150, two factors account for aerosol retention on the secondary side of the steam generator. One factor represents the fraction released from the core that enters the secondary side of the steam generator; the second is the fraction entering the secondary side of the steam generator that is released to the environment through the safety relief valves (SRVs). For the SGTRs where the secondary system SRV sticks open, an expert panel determined the distributions for these two factors. Expert elicitation is invaluable; however, the capabilities exist today to obtain a more informed estimate of the contributions of these factors to the source term.

With this enormous amount of surface area available for aerosol removal, one questions how much reduction in the source term to attribute to aerosol retention in the secondary. In order to quantify secondary side aerosol retention subsequent to SGTR events the relevant mechanisms require proper identification, mathematical modeling, followed by validation against experimental data.

From 1998 to 2002, the fifth EURATOM Framework Program (EU-SGTR) conducted an experimental project to generate a comprehensive database on fission product retention in a steam generator and to verify and/or develop predictive models to support accident management interventions in SGTR sequences. The contribution of the Paul Scherrer Institute (PSI) to the EU-SGTR project was to investigate the integral effects in the bundle section of a model vertical steam generator. PSI conducted five tests in the framework of the EU-SGTR integral tests in a vertical steam generator bundle. These investigations dealt with aerosol retention in the tube bundle under dry, wet, and accident management conditions.

The ARTIST[8] (AeRosol Trapping In a Steam generaTor) experimental project being conducted at PSI is a continuation of EU-SGTR of sorts. The experimental investigation of aerosol retention is carried out in a scaled down PWR steam generator model of Framatome 33/19 design (see Figure 3, taken from Ref. 8). The primary objectives of the ARTIST experiments involve flow simulation and quantification of aerosol retention on the steam generator secondary side following a postulated SGTR event (see Table 2).

Table 2. Essential Features of the Transient Sequence Used in the ARTIST Project

1. Both primary and secondary sides of the faulted steam generator are completely empty before the core starts to heat-up and remains so throughout the rest of the sequence;
2. The intact steam generator is depressurized to ambient pressure by operator action before the onset of core degradation; and
3. Accumulation of hydrogen in the intact steam generator impedes the condensation of steam and limits the primary side depressurization such that choked flow occurs at the break.

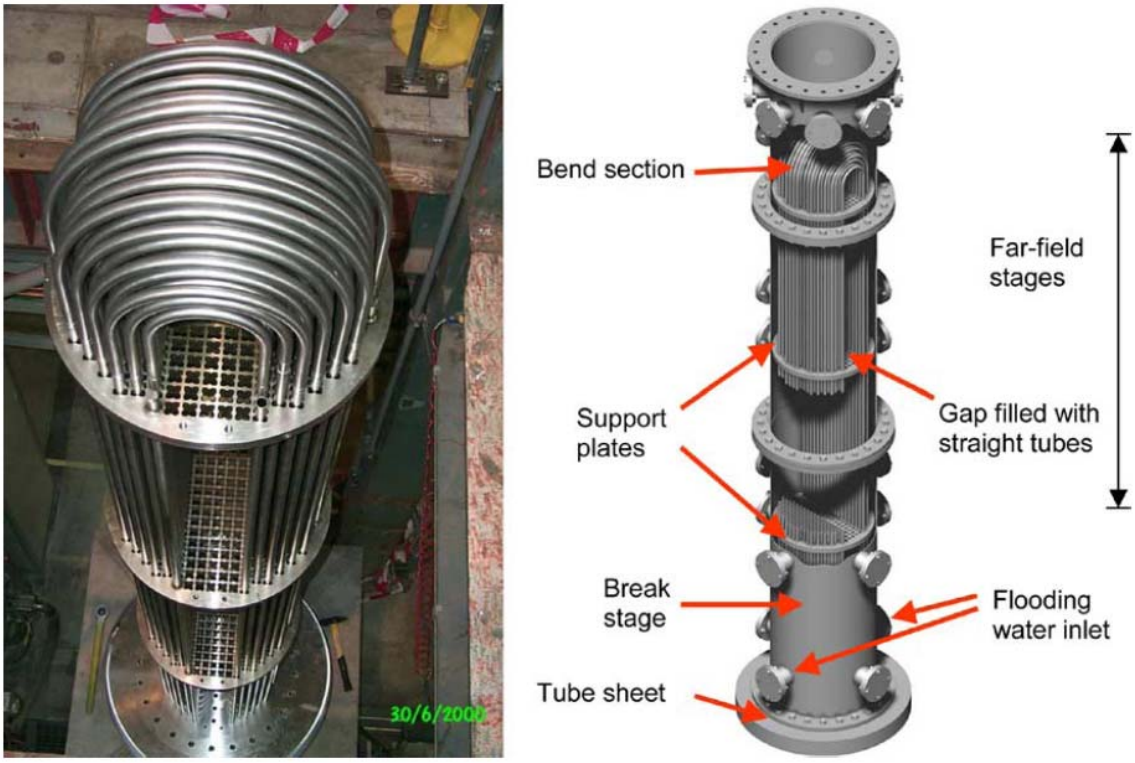


Figure 3. ARTIST SG: Photo and Schematic Representation of the Test Section

The seven project phases of the ARTIST experiment consist of quantification of aerosol retention in/on:

1) an intact steam generator tube;
2) the immediate vicinity of a ruptured tube, “break stage”;
3) the tube bundle several tube diameters away from a ruptured tube, “far-field”;
4) upper structures to include separators, dryers and support plates; and
5) the flooded bundle, an accident mitigation scheme whereby the tube bundle is covered with water.

An investigation of droplet retention in upper structures constitutes the sixth phase. The final phase, the integral test, will incorporate the cumulative knowledge gathered from

the other test phases. By its very nature and in many places, flows in the ARTIST experiment are turbulent, such as inside the intact as well as the broken tube, near the break, along tubes with significant temperature gradient with respect to the bulk fluid and in the separator and dryer.

This study has application to the first phase of the ARTIST project. Test conditions specified in ARTIST serve as the metric for this aerosol retention inside a single U-shaped tube investigation. Therefore, the primary objective of the research effort at hand is to simulate the aforementioned retention using computational fluid dynamics (CFD) tools.

The initial and perhaps the most crucial step in the study of aerosol deposition in the tube is flow field characterization. Proper representation of the primary flow is crucial for adequate prediction of particle deposition. Flow through the configuration under investigation in this study constitutes an internal wall-bounded flow. The presence of the wall introduces variation in the mean velocity at any cross section in the flow domain. The instantaneous velocity is equal to the mean velocity which is known (for instance from turbulence model predictions) plus a fluctuation, which is unknown. The fluctuating velocity component transports particles away from the mean flow and is responsible for particle deposition. However, in order to quantify particle deposition at the wall, the numerical approach must account for particle dispersion by the turbulence in the flow field. In major CFD codes, the Lagrangian Eddy Impaction Model (EIM) developed by Gosman and Ioannides[9] determines particle dispersion in random walk

models. In FLUENT, the discrete random walk (DRW) model facilitates this computation of particle trajectories.

In the EIM, the motion of a dispersed particulate phase in a turbulent primary flow is determined by particle interactions with a succession of eddies characterized by the scales, "eddy length" (distance traveled as a particle crosses an eddy) and "eddy lifetime" (time required for a particle to cross an eddy). One shortcoming in this exchange is the assumption that the fluctuation velocity sampled for the eddy remains constant throughout the lifetime of the eddy.[10] By allowing random time and length scales in the eddy interaction model, it is possible to describe real turbulent flows more accurately. Randomly sampled scales have been proposed by Kallio and Reeks[11] (used time scales sampled from an exponential probability distribution), Burnage and Moon[12] (used exponential distributions for time and length scales), and Wang and Stock[13] (used several different time scale distributions and developed a general method to find the Lagrangian integral time scale for a given distribution).

Another EIM shortfall and perhaps the most important is the isotropy assumption. Local turbulence kinetic energy values from the turbulence model are used to isotropically reconstruct fluctuating components of the instantaneous fluid velocity. In the case of wall-bounded flows, this assumption may be valid in the core region, far away from the walls. However, from both experimental[14] and numerical results[15] the assumption of isotropy does not hold in the near-wall region. Accurate prediction of fluid turbulence conveyance to the particles requires inclusion of anisotropy in velocity

fluctuations. Only then does the action of stochastic force (i.e., turbulent dispersion force) on particles in EIM account for particle dispersion due to fluid turbulence.

In short, the physical model based on discrete random walk for treatment of the particulate phase in standard CFD codes, such as FLUENT, CFX, and STAR-CD, requires modification. This model must cover in addition to the existing isotropic turbulence in the bulk flow, anisotropic turbulence in the boundary layer in different scales. As pointed out, the main limitation of the existing model is the consideration of only the average fluid quantities in the turbulence models. Particles in reality will interact with the instantaneous fluid motion within the boundary layer. As such, the particle-turbulence interaction phenomena in the boundary layer dictate deposition of particles on surfaces for inertial particles. As previously mentioned such phenomena are inadequately treated in standard CFD codes. Therefore, development and integration into the standard CFD code of an improved model for the particle tracking under conditions especially important for turbulent internal flow is necessary.

The CFD code used in this study is FLUENT. The reason for selection of FLUENT and not another CFD code is that the FLUENT code has become a workhorse for many ARTIST project participants. PSI entered into a framework agreement with FLUENT to develop for integration into the FLUENT code the improved model. As a participating ARTIST partner, the author provided recommendations on the requisite modifications to the EIM. ARTIST partners performing particle-tracking analysis for the project, upon request, received this part of the code (included in Appendix A).

Once equipped with the correct fluid flow field and particle trajectory calculations, evaluation of the penetration efficiency (ratio of particles completely traversing the system relative to initial particles released into the system) is possible. Penetration (P) is related to deposition (= 1 - P) because particles that do not negotiate through the system are assumed to deposit within the system. The literature search for established treatment of deposition of particles in turbulent tube flow yielded only three correlations for the prediction of deposition in bends. However, numerous experimental and numerical studies conducted describe flow characteristics in this geometry-type. In addition, the aforementioned correlations and studies involved turbulence levels substantially lower than (i.e., laminar to slightly turbulent) the conditions in the present study.

Many authors studied turbulent flow through square-duct bend geometry. Chang[16] conducted experiments and computations for water flowing into a square duct with a 180º bend and bend radius equal to 3.35 times the hydraulic diameter of the duct. Anwer[17] revealed that the imbalance between the radial pressure gradient and centrifugal forces sets up a Dean-type secondary motion (see Figure 4) in a curved pipe. Perturbation by and recovery from bend curvature were the main interest. The pipe Reynolds number and the bend curvature ratio ($\delta \approx R_b/r_p$ = 495.3 mm/38.1 mm) were 5.0 x 10^4 and 13, respectively (in this study, pipe Re ≈ 2.8 x 10^5 and $\delta \approx$ 82.3 mm/8.435 mm ≈ 9.76). Anwer used hot-wire techniques to measure the three components of mean velocity and the six components of the Reynolds stress tensor along the horizontal as

well as along the normal plane in streamwise locations extending from 18 diameters on either side of the U-bend (i.e., upstream and downstream of bend).

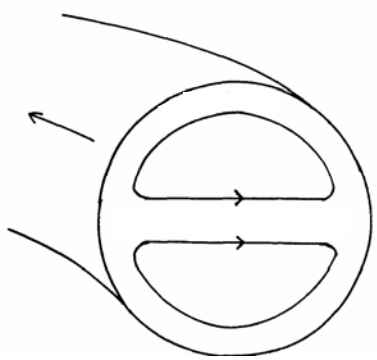


Figure 4. Dean-Type Secondary Flow

Lai[18] performed a detailed numerical study of turbulence-driven secondary flows in a circular-sectioned U-bend. They found three vortex pairs; namely the Dean-type vortex pair as a primary one, the center vortex pair near the pipe core and the separation vortex pair near the outer bend. Cheng and Farokhi[19] developed a turbulence model based on new algebraic formulations for the eddy viscosity incorporating the k-ε turbulence model. This model proposed to account for various effects of streamline curvature, anisotropy effects for instance, while the standard k-ε model assumes isotropic conditions of turbulence parameters. Sudo[20,21,22] performed experimental studies for a circular-sectioned 90º bend, a circular-sectioned 180º bend and a square-sectioned 90º bend. Sudo provided measurement data for the mean velocity, pressure distribution, turbulence intensity and Reynolds stresses using a hot wire anemometer.

In multiphase flow, the studies of gas-solid flow in 90º bends have attracted more attention than U-bends. This is understandable because most studies of gas-solid flows have focused on a vertical circulating fluidized bed and pneumatic conveying as regards transporting solid material.[23, 24, 25, 26, 27]

Deposition in straight pipes receives much attention in the literature and extensive research continues in the characterization of deposition in the straight pipe configuration. However, aerosol penetration through a 180º bend has received much less attention. Two correlations, empirically derived equations to calculate particle penetration through a 180º bend as a function of the dimensionless ratio of the stop distance of a particle to the characteristic dimension of the system (Stokes number, Stk) appear in the literature. Pui[28] and others provide a fit of experimental data of particle penetration measurements through 90º bends in 5.03-8.51 mm diameter tubes for flow Reynolds numbers at 6,000 and 10,000. Although based on behavior in a 90º bend, this correlation is frequently used to determine deposition in 180º bends. For the second correlation, McFarland[29] and others provide a fit to results from Lagrangian simulations for particle penetration through 45º, 90º, and 180º bends in 16 mm diameter tubes for flow Reynolds numbers at 3,200 and 19,800. The McFarland correlation includes in addition to Stokes number dependence, effects of bend angle (θ, e.g., 180°) and bend curvature ratio (δ) on particle penetration efficiency. A third penetration efficiency correlation for particles in turbulent flow through a bend advanced by Brockmann[30] is the result of data compilation. The Brockmann correlation is a function of Stokes number and angle of the bend in radians.

It may be inappropriate to directly apply either correlation to the experimental results from the ARTIST test series. The correlations were developed based on incompressible flow conditions. However, in the ARTIST test series gas density variability is significant consequently, compressibility effects cannot be ignored. Another reason why these correlations probably should not be directly applied becomes apparent with comparison of the range of validity of the correlations. Table 3 shows primary parameters of interest placed side by side for comparison. The supposition previously advanced will be tested to judge the predictive capability of the correlations using the numerical results generated in this study.

Table 3. Comparison of Parameters of Interest

	Pui, et al.[28]	McFarland, et al.[29]	ARTIST
Bend angle	90º	45º, 90º, & 180º	180º
Radius of curvature			82.3 mm
Reynolds number	6×10^3 - 1.0×10^4	$3.2 \times 10^3 - 1.98 \times 10^4$	~2.8×10^5
Internal tube diameter (mm)	5.03-8.51	16	16.87
Curvature ratio	10	2, 4, 10	9.76
Stokes number	0.03 – 1.46	0.07 – 1.2	2.609, inlet 0.4003, outlet
Penetration Efficiency	$\log P = -0.963 Stk$	$\ln P = \frac{4.61 + a\theta Stk}{1 + b\theta Stk + c\theta Stk^2 + d^2\theta Stk}$ $a = -0.9526 - 0.05686\delta$ $b = \frac{-0.297 - 0.0174\delta}{1 - 0.07\delta + 0.0171\delta^2}$ $c = -0.306 + \frac{1.895}{\sqrt{\delta}} - \frac{2.0}{\delta}$ $d = \frac{0.131 - 0.132\delta + 0.000383\delta^2}{1 - 0.129\delta + 0.0136\delta^2}$	-

Comparing the Reynolds number in row 3 of Table 3, it is observed that the level of turbulence present in the ARTIST test series exceed the upper bound for either correlation. Furthermore, it is counterintuitive given the range of validity of the three published correlations that either would be suitable for extrapolation. What these correlations provide however is a point at which to start development of a correlation (if necessary) that depicts the aerosol deposition behavior observed in the ARTIST tests.

Several options are available for consideration in the computational simulation of this turbulent compressible internal wall bounded flow. The ideal candidate for the characterization of the turbulent processes (e.g., flow and deposition) is direct numerical simulation (DNS) and a close second is the large eddy simulation (LES) method. DNS does not involve turbulence modeling of the flow but consists of the full solution of the standard transient Navier-Stokes equations. Therefore, the level of description and accuracy obtained by applying DNS cannot be equaled with other approaches (e.g., LES, RANS). However, according to Pope,[31] the drawback of DNS is of course its very large computational cost and the fact that this cost increases rapidly with the Reynolds number (Re). For DNS, the required number of grid points scales approximately as $Re^{9/4}$ and CPU time as Re^3. This means that for a Reynolds number of 2.8×10^5, approximately 4×10^{12} grid points requires 22×10^{15} seconds of CPU time on a modestly configured workstation.

Pope describes for LES of wall-bounded flows, approaches for near-wall resolution (LES-NWR) and near-wall modeled (LES-NWM). In the LES-NWR a sufficiently fine grid and filter resolves 80% of the energy everywhere. In LES-NWM,

the near-wall motions are not resolved; typically, a mean profile is applied. No further consideration of the latter is entertained since the interest in this study is to resolve the near-wall motions. Pope provides an exercise (p. 599) permitting calculation of the number of grid points for scenarios involving resolution of the near-wall region. The requisite number of grid points to resolve the continuous phase scales as $Re^{7/4}$ while the CPU time scales as $4 \times 10^3 Re^{7/4}$ ($\sim 16 \times 10^{12}$ seconds). We find that over a billion grid points are required for an unstructured grid.

Reynolds Averaged Navier-Stokes (RANS) 3D models solving two transport equations (e.g., turbulence kinetic energy and its dissipation rate) require 10^6 grid points and 10^9 seconds of CPU time. Including stress-tensor transport equations and viscous sublayer increases the number of grid points by a factor of $Re^{1/5}$ and CPU time by $4 \cdot Re^{1/5}$ (12×10^6 grid points and 49×10^9 seconds, respectively).

We see using the example ($Re = 2.8 \times 10^5$) that the number of grid points required to resolve the continuous phase [4×10^{12} (DNS), 4×10^9 (LES), and 10^6 (RANS)] and assuming each point (or node) requires 1000 bytes of memory for computation, corresponds to a petabyte (2^{50}), terabyte (2^{40}), and gigabyte (2^{30}), respectively. Thus, the amount of computation strongly depends on available computational resources. CPU times; 22×10^{15} seconds (DNS), 16×10^{12} seconds (LES), and 10^9 seconds (RANS); suggests the RANS approach is the least computationally intensive. Even so, the CPU time is prohibitively large. Fortunately, algorithms in CFD codes and multiple processors aid in its drastic reduction such that it is possible to perform RANS calculations on workstation-type computer systems. A

Dell Precision Workstation 650 with dual Intel® Xeon™ 2.8 GHz processor (4 GB RAM) running under Linux OS was used for the calculations in this study.

The available computational resources allowed solution of the flow field using two-equation turbulence models (k-ε and its variants or k-ω and its variants). Prediction of aerosol tracks is by solution of the particle's law of motion equation, standard treatment in a Lagrangian solution framework. In addition to drag and to a negligible degree, gravity, the particles are also subject to turbulent dispersion because of the high turbulence intensities. In situations like this, special treatment in the various boundary layers is required to capture such effects as flow deceleration in the laminar sublayer, lift force, and structure near the walls. Again, modification of the CFD code is required to include boundary layer effects in the particle tracking treatment.

This document contains a detailed description of the procedure followed to obtain the boundary layer particle behavior for aerosol-laden turbulent flow through the 180° bend. Data from the ARTIST experimental test series served as a source of comparison and to a varying degree, validation.

Following this introduction is a discussion in Chapter II of the Eulerian approach for modeling the fluid flow field, the Lagrangian approach for ascertaining particle trajectories, modifications to EIM, and other mathematical models used in the analysis. The third chapter contains the discussion of the solution algorithm (i.e., grid model and numerical scheme). The fourth chapter contains a discussion of results and the fifth, conclusions. Supplemental information appears in the appendices (e.g., Appendix A: source code, Appendix B: diagrams depicting flow characteristics, etc.).

CHAPTER II

MATHEMATICAL MODELS

Compressible flows create a unique set of flow physics for which the FLUENT code has special input requirements and solution techniques. Input and solution techniques will be touched on briefly in this chapter but a more complete discussion is in the next chapter. In this chapter the rationale for the treatment of the aerosol-laden gas flow under turbulent conditions is developed.

The conservation equations presented and assumptions advanced in this chapter are found in any standard gas dynamics, compressible fluid flow, or viscous flow textbook. Derivation of the conservation equations from first principles is included herein for completeness. Equations for the Eulerian description of the continuous phase and Lagrangian treatment of the dispersed phase although taken from the FLUENT User's Guide appears in any CFD text. These governing equations facilitate development of the modification to the Lagrangian Eddy Impaction model presented in this chapter.

Compressible Flow

Due to the high velocity and large pressure variations in the gas flow for this study, the expectation is that compressibility effects are significant. Theory tells us that when the flow velocity approaches or exceeds the speed of sound of the gas or when the pressure change in the system ($\Delta p/p$) is large, the variation of the gas density with pressure has a significant impact on the flow velocity, pressure, and temperature.

Generally, the magnitude of the Mach number (M) determines conditions under which compressibility effects influence the flow field properties. For instance, compressibility effects are negligible and the variation of the gas density with pressure can safely be ignored for Mach numbers much less than 1.0 ($M < 0.1$ or so). Subsonic flow has Mach number less than 1.0. In the transonic flow regime (M approaches 1.0) compressibility effects become important. When the Mach number exceeds 1.0, the flow is termed supersonic. The definition of the Mach number is the ratio of average velocity (u) to the speed of sound in the gas (c).

$$M \equiv \frac{u}{c} \tag{1}$$

The speed of sound in the gas is defined in terms of γ, the ratio of specific heats (c_p/c_v), universal gas constant (R) and temperature (T).

$$c = \sqrt{\gamma RT} \tag{2}$$

Flows studied in this analysis are by definition subsonic, perhaps transonic.

From fundamental gas dynamics, another way of characterizing compressibility is by the stagnation relations for the pressure (p_0) and temperature (T_0) of the flow. For a perfect gas (i.e., ideal gas law is applicable), the relationship between stagnation and static pressure (p) and temperature (T) exists as follows, assuming constant c_p. We define the specific heat at constant pressure (c_p) in terms of γ and R.

$$\frac{p_0}{p} = \left(1 + \frac{\gamma - 1}{2} M^2\right)^{\frac{\gamma}{\gamma - 1}} \tag{3}$$

$$\frac{T_0}{T} = 1 + \frac{\gamma - 1}{2} M^2 \quad \textbf{(4)}$$

$$c_p = \frac{\gamma R}{\gamma - 1} \quad \textbf{(5)}$$

These relationships describe the variation of the static pressure and temperature in flows as the velocity (Mach number) changes under isentropic conditions.

In FLUENT, the ideal gas law for compressible flow includes a fictitious term (operating pressure, p_{op}) in its definition.

$$p_{op} + p = \rho \left(\frac{R}{M_w} \right) T \quad \textbf{(6)}$$

The sum, operating pressure (p_{op}) and local static pressure (p) constitute the pressure term in the compressible form of the ideal gas expression. In the ideal gas expression, M_w is molecular weight of the gaseous species. In FLUENT, solution of the energy equation yields temperature, T.

Governing Equations of Fluid Flow

Table 4 lists the set of governing equations for compressible fluid flow. Presented in the following sections is their derivation from basic conservation principles. These conservation laws of physics governing fluid flow behavior include conservation of mass (continuity equation), Newton's second law (momentum equation), and the first law of thermodynamics (energy equation). Recall, Newton's second law states that the rate of change of momentum equals the sum of forces on the fluid particle. The first law

of thermodynamics states that the rate of change of energy is equal to the sum of the rate of heat addition and the rate of work done on a fluid particle. The thermodynamic equilibrium assumption and the Newtonian model of viscous stresses aid in mathematical system closure. Macroscopic properties, such as velocity, pressure, density, and temperature, along with their space and time derivatives describe of the behavior of the fluid.

It is consistent with theory to consider macroscopic properties as averages over large numbers of molecules. Such an assumption allows for any arbitrary point in a fluid to represent the smallest possible element of fluid such that macroscopic properties exist uninfluenced by individual molecules. In the formulation to follow, such a small element of fluid (with sides: δx, δy and δz) will be considered (Figure 5). The center of the element is located at position (x, y, z).

Table 4. The Governing Equations of the Flow of a Compressible Newtonian Fluid

Mass	$\frac{\partial \rho}{\partial t} + div\left(\rho \vec{u}\right) = 0$
x-momentum	$\frac{\partial (\rho u)}{\partial t} + div\left(\rho u \vec{u}\right) = -\frac{\partial p}{\partial x} + div(\mu \nabla u) + S_{Mx}$
y-momentum	$\frac{\partial (\rho v)}{\partial t} + div\left(\rho v \vec{u}\right) = -\frac{\partial p}{\partial y} + div(\mu \nabla v) + S_{My}$
z-momentum	$\frac{\partial (\rho w)}{\partial t} + div\left(\rho w \vec{u}\right) = -\frac{\partial p}{\partial z} + div(\mu \nabla w) + S_{Mz}$
Internal energy	$\frac{\partial (\rho i)}{\partial t} + div\left(\rho i \vec{u}\right) = -p div \vec{u} + div(k \nabla T) + \Phi + S_i$
Equations of state	$p = p(\rho, T)$ and $i = i(\rho, T)$ e.g., perfect gas $p = \rho RT$ and $i = c_v T$

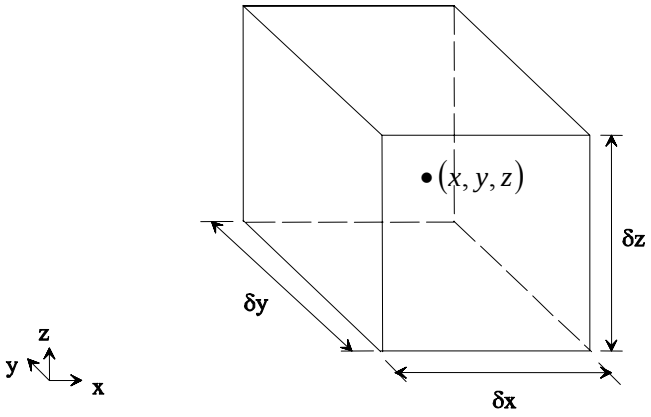


Figure 5. Conservation Laws: Fluid Element

A systematic account of changes in the mass, momentum, and energy of the fluid element due to fluid flow across its boundaries and, where appropriate, due to the action of sources inside the element, leads to the fluid flow equations. All fluid properties are functions of space and time (i.e., density, $\rho(x, y, z, t)$; pressure, $p(x, y, z, t)$; temperature, $T(x, y, z, t)$, and velocity vector, $\mathbf{u}(x, y, z, t)$).

The element under consideration is so small that fluid properties at the faces can be expressed accurately enough by means of the first two terms of a Taylor series expansion. For example, the pressure relative to the center of the element in the x-direction can be expressed as follows.

$$p - \frac{\partial p}{\partial x} \cdot \frac{1}{2} \delta x \quad \text{and} \quad p + \frac{\partial p}{\partial x} \cdot \frac{1}{2} \delta x \tag{7}$$

Mass Conservation

Transcribing the mass balance for the fluid element serves as the starting point in the derivation of the mass conservation equation. This is accomplished by equating the rate of increase of mass in the fluid element to the net rate of flow of mass into the fluid element. Figure 6 facilitates generation of the appropriate mathematical relations. The convention employed in the diagram is that flows directed into the element produce an increase of mass in the element. These flows have positive directional assignment. Thus, flows leaving the element receive a negative sign.

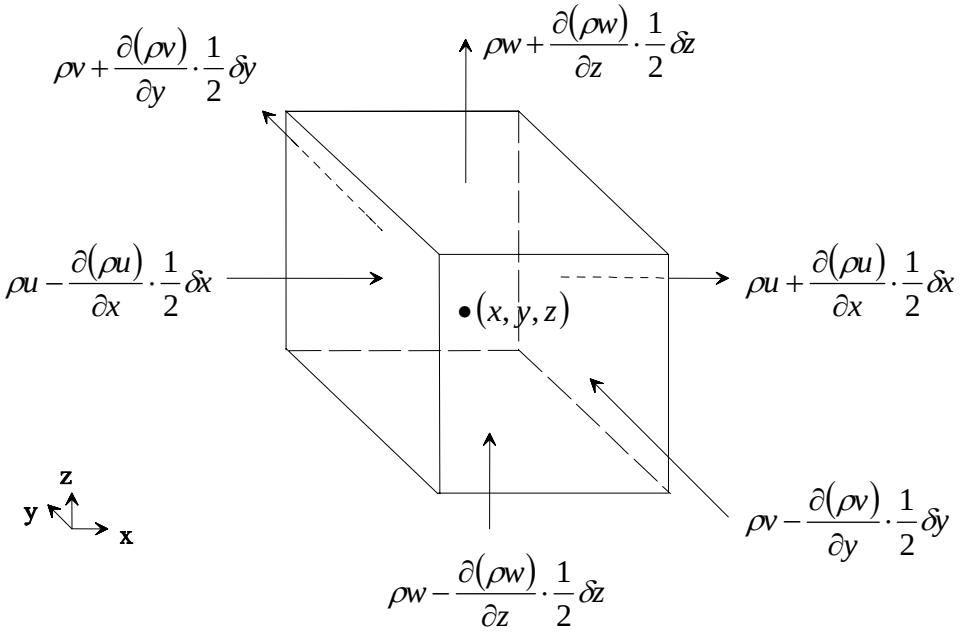


Figure 6. Fluid Element: Mass Flows at Boundaries

By definition, the rate of increase of mass in the fluid element is:

$$\frac{\partial}{\partial t}(\rho \delta x \delta y \delta z) = \frac{\partial \rho}{\partial t} \delta x \delta y \delta z \qquad \textbf{(8)}$$

The product of density, area and the velocity component normal to the face accounts for the mass flow rate across a face of the element. Equation 9, as illustrated in Figure 6, is a statement describing the net rate of flow of mass into the element across its boundaries.

$$\begin{aligned}&\left(\rho u-\frac{\partial(\rho u)}{\partial x}\cdot\frac{1}{2}\delta x\right)\delta y\,\delta z-\left(\rho u+\frac{\partial(\rho u)}{\partial x}\cdot\frac{1}{2}\delta x\right)\delta y\,\delta z\\&+\left(\rho v-\frac{\partial(\rho v)}{\partial y}\cdot\frac{1}{2}\delta y\right)\delta x\,\delta z-\left(\rho v+\frac{\partial(\rho v)}{\partial y}\cdot\frac{1}{2}\delta y\right)\delta x\,\delta z\\&+\left(\rho w-\frac{\partial(\rho w)}{\partial z}\cdot\frac{1}{2}\delta z\right)\delta x\,\delta y-\left(\rho w+\frac{\partial(\rho w)}{\partial z}\cdot\frac{1}{2}\delta z\right)\delta x\,\delta y\end{aligned}\qquad \textbf{(9)}$$

To obtain the unsteady, three-dimensional mass conservation or continuity equation at a point in a compressible fluid, one equates the rate of increase of mass inside the element to the net rate of flow of mass into the element across its faces. Arrangement of all terms of the resulting mass balance on the left hand side of the equal sign and dividing the expression by the element volume ($\delta x\,\delta y\,\delta z$) yields Equation 10.

$$\frac{\partial\rho}{\partial t}+\frac{\partial(\rho u)}{\partial x}+\frac{\partial(\rho v)}{\partial y}+\frac{\partial(\rho w)}{\partial z}=0 \qquad \textbf{(10)}$$

Equation 11 expresses Equation 10, the continuity equation, in a more compact form.

$$\frac{\partial\rho}{\partial t}+div\left(\rho\vec{u}\right)=0 \qquad \textbf{(11)}$$

The first term on the left hand side, the time rate of change of density (the Lagrangian time derivative of density) is the temporal rate of change of density observed by one who moves with the particle. The second (convective) term describes the net flow of mass out of the element across its boundaries. There sum is identically zero.

In the Eulerian representation of the continuity equation, $\partial\rho/\partial t + \vec{u}\cdot\nabla\rho = 0$, the first term on the right is the local rate of change; the second is the convective rate of change of density. The latter equals the speed of the fluid times the directional derivative of density in the direction of motion.

Momentum Equation

According to Newton's second law, the rate of change of momentum of a fluid particle equals the sum of the forces on the particle. Equation 12 shows using total derivatives, the rates of increase of *x*-, *y*- and *z*-momentum per unit volume of a fluid particle.

$$\rho\frac{Du}{Dt} = \frac{\partial(\rho u)}{\partial t} + div(\rho u\vec{u}) \qquad \rho\frac{Dv}{Dt} = \frac{\partial(\rho v)}{\partial t} + div(\rho v\vec{v}) \qquad \rho\frac{Dw}{Dt} = \frac{\partial(\rho w)}{\partial t} + div(\rho w\vec{w}) \tag{12}$$

The two types of forces typically recognized are surface (e.g., pressure and viscous) and body (e.g., gravity, centrifugal force, Coriolis force, electromagnetic force) forces. In the momentum equation, the common practice is to combine contributions due to the surface forces as separate terms and to include the effects of body forces as source terms. The pressure (a normal stress, denoted by *p*) and the nine viscous stress components (τ_{ij}) define the state of stress of a fluid element (shown in Figure 7). Subscripts in viscous stress components indicate that the stress component acts in the *j*-direction on a surface normal to the *i*-direction. The sign associated with pressure is opposite to that of the normal viscous stress. This is because the usual sign convention

takes a tensile stress to be the positive normal stress so that the pressure, which is by definition a compressive normal stress, has a negative sign.

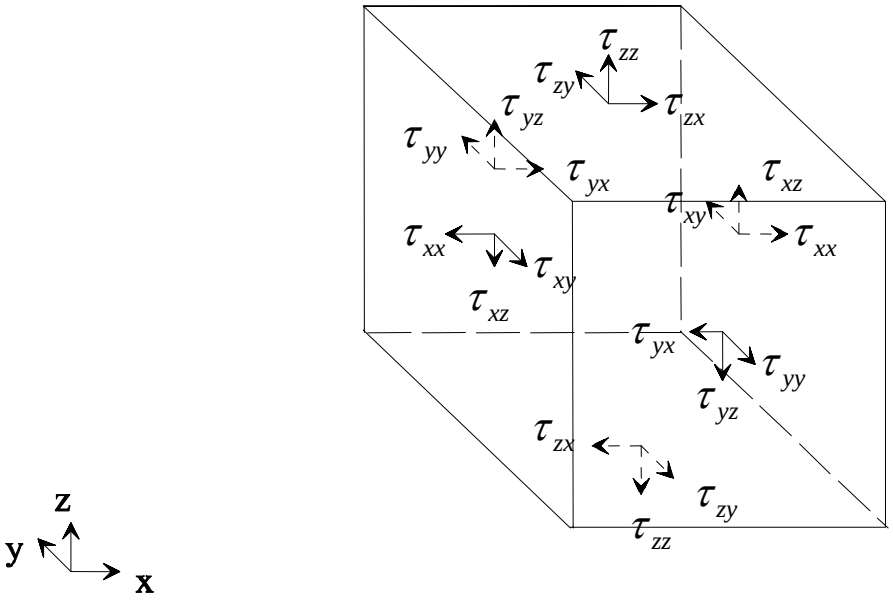


Figure 7. Fluid Element: Stress Components

By definition, the magnitude of a force resulting from a surface stress is the product of stress and area. Derivations of the y- and z-component of momentum follow the same method as that of the x-component of momentum. With that said, Figure 8 shows the x-components of the forces due to pressure p and stress components τ_{xx}, τ_{yx}, and τ_{zx}. The net force in the x-direction is the sum of the force components acting in that direction on the fluid element. The following expressions (Equations 13 through 15) describe the forces acting on left and right faces; back and front faces; and top and bottom faces.

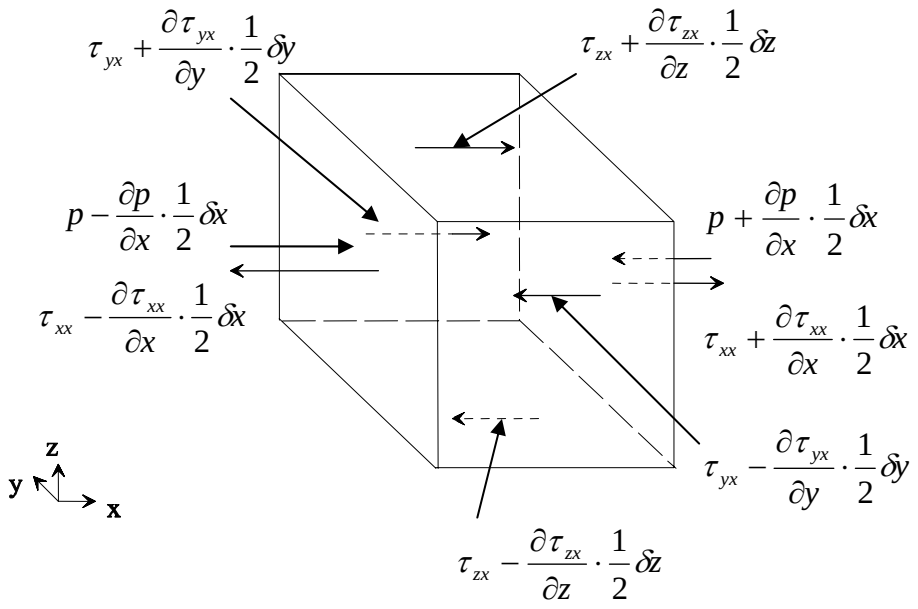


Figure 8. Stress Components (*x*-Direction)

$$\left(-\frac{\partial p}{\partial x}+\frac{\partial\tau_{xx}}{\partial x}\right)\delta x\delta y\delta z=\left[\left(p-\frac{\partial p}{\partial x}\cdot\frac{1}{2}\delta x\right)-\left(\tau_{xx}-\frac{\partial\tau_{xx}}{\partial x}\cdot\frac{1}{2}\delta x\right)\right]\delta y\delta z+\left[-\left(p+\frac{\partial p}{\partial x}\cdot\frac{1}{2}\delta x\right)+\left(\tau_{xx}+\frac{\partial\tau_{xx}}{\partial x}\cdot\frac{1}{2}\delta x\right)\right]\delta y\delta z \tag{13}$$

$$\frac{\partial\tau_{yx}}{\partial y}\delta x\delta y\delta z=-\left(\tau_{yx}-\frac{\partial\tau_{yx}}{\partial y}\cdot\frac{1}{2}\delta y\right)\delta x\delta z+\left(\tau_{yx}+\frac{\partial\tau_{yx}}{\partial y}\cdot\frac{1}{2}\delta y\right)\delta x\delta z \tag{14}$$

$$\frac{\partial\tau_{zx}}{\partial z}\delta x\delta y\delta z=-\left(\tau_{zx}-\frac{\partial\tau_{zx}}{\partial z}\cdot\frac{1}{2}\delta z\right)\delta x\delta y+\left(\tau_{zx}+\frac{\partial\tau_{zx}}{\partial z}\cdot\frac{1}{2}\delta z\right)\delta x\delta y \tag{15}$$

The summation of these terms divided by the volume ($\delta x\delta y\delta z$) yields the total force per unit volume on the fluid due to surface stresses.

$$\frac{\partial}{\partial x}\left(-p+\tau_{xx}\right)+\frac{\partial\tau_{yx}}{\partial y}+\frac{\partial\tau_{zx}}{\partial z} \tag{16}$$

Inclusion of body forces as a source (S_{Mx}) of *x*-momentum per unit volume per unit time and setting the rate of change of *x*-momentum of the fluid particle equal to the total force in the *x*-direction on the element due to surface stresses yields the *x*-momentum equation.

$$\rho\frac{Du}{Dt}=\frac{\partial}{\partial x}\left(-p+\tau_{xx}\right)+\frac{\partial\tau_{yx}}{\partial y}+\frac{\partial\tau_{zx}}{\partial z}+S_{Mx} \qquad \textbf{(17)}$$

The *y*- and *z*-component of the momentum equation have similar form (Equation 18 and 19, respectively).

$$\rho\frac{Dv}{Dt}=\frac{\partial\tau_{xy}}{\partial x}+\frac{\partial}{\partial y}\left(-p+\tau_{yy}\right)+\frac{\partial\tau_{zy}}{\partial z}+S_{My} \qquad \textbf{(18)}$$

$$\rho\frac{Dw}{Dt}=\frac{\partial\tau_{xz}}{\partial x}+\frac{\partial\tau_{yz}}{\partial y}+\frac{\partial}{\partial z}\left(-p+\tau_{zz}\right)+S_{Mz} \qquad \textbf{(19)}$$

The viscous stress components τ_{ij} in the momentum equations are unknowns. Therefore, a suitable model for the viscous stresses (τ_{ij}) is required. Expressing the viscous stresses as functions of the local deformation rate (or stain rate) is but one way to resolve the problem. The local rate of deformation is composed of the linear deformation rate and the volumetric deformation rate, in three-dimensional flows. In the following formulation, all gases are assumed isotropic.

For isotropic fluids[32] and three-dimensional flows, the rate of linear deformation (e_{ij}) of fluid element has nine components, six of which are independent. There are three linear elongating deformation components (Equation 20).

$$e_{xx}=\frac{\partial u}{\partial x} \qquad e_{yy}=\frac{\partial v}{\partial y} \qquad e_{zz}=\frac{\partial w}{\partial z} \qquad \textbf{(20)}$$

There are also six shearing linear deformation components (Equation 21).

$$e_{xy} = e_{yx} = \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \qquad e_{yz} = e_{zy} = \frac{1}{2}\left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z}\right) \qquad e_{xz} = e_{zx} = \frac{1}{2}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \tag{21}$$

The suffix interpretation is identical to that for stress components.

Viscous stresses are proportional to the rates of deformation in a Newtonian fluid. Newton's law of viscosity for compressible flows involves two proportionality constants. The proportionality constant, dynamic viscosity (μ) relates stresses to linear deformations the second proportionality constant viscosity (Λ) relates stresses to the volumetric deformation. Schlichting[33] (p. 57) gives using Stokes's hypothesis this viscosity as: $\Lambda = -\frac{2}{3}\mu$. The nine viscous stress components, of which six are independent, are

$$\tau_{xx} = 2\mu\frac{\partial u}{\partial x} + \Lambda div\vec{u} \qquad \tau_{yy} = 2\mu\frac{\partial v}{\partial y} + \Lambda div\vec{u} \qquad \tau_{zz} = 2\mu\frac{\partial w}{\partial z} + \Lambda div\vec{u} \tag{22}$$

$$\tau_{xy} = \tau_{yx} = \mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \qquad \tau_{yz} = \tau_{zy} = \mu\left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z}\right) \qquad \tau_{xz} = \tau_{zx} = \mu\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \tag{23}$$

Substitution of the shear stresses into the momentum equations yields the Navier-Stokes equations.

$$\rho\frac{Du}{Dt} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x}\left[2\mu\frac{\partial u}{\partial x} + \Lambda div\vec{u}\right] + \frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right] + \frac{\partial}{\partial z}\left[\mu\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right)\right] + S_{Mx} \tag{24}$$

$$\rho\frac{Dv}{Dt}=-\frac{\partial p}{\partial y}+\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right]+\frac{\partial}{\partial y}\left[2\mu\frac{\partial v}{\partial x}+\Lambda div\vec{u}\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right]+S_{My} \quad \textbf{(25)}$$

$$\rho\frac{Dw}{Dt}=-\frac{\partial p}{\partial z}+\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)\right]+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right]+\frac{\partial}{\partial z}\left[2\mu\frac{\partial w}{\partial z}+\Lambda div\vec{u}\right]+S_{Mz} \quad \textbf{(26)}$$

Often it is useful to rearrange the viscous stress terms as follows:

$$\begin{aligned}
&\frac{\partial}{\partial x}\left[2\mu\frac{\partial u}{\partial x}+\Lambda div\vec{u}\right]+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)\right]\\
&=\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial u}{\partial z}\right)\\
&+\left[\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial v}{\partial x}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial w}{\partial x}\right)\right]+\frac{\partial}{\partial x}\left(\Lambda div\vec{u}\right)\\
&=div\left(\mu\nabla u\right)+s_{Mx}
\end{aligned} \quad \textbf{(27)}$$

The two smaller contributions to the viscous stress terms in the momentum source are hidden in the source defined by $S_M = S_M + s_M$. The Navier-Stokes equations are shown below in the most useful form for the development of the finite volume method.

$$\rho\frac{Du}{Dt}=-\frac{\partial p}{\partial x}+div\left(\mu\nabla u\right)+S_{Mx} \quad \textbf{(28)}$$

$$\rho\frac{Dv}{Dt}=-\frac{\partial p}{\partial y}+div\left(\mu\nabla v\right)+S_{My} \quad \textbf{(29)}$$

$$\rho\frac{Dw}{Dt}=-\frac{\partial p}{\partial z}+div\left(\mu\nabla w\right)+S_{Mz} \quad \textbf{(30)}$$

Energy Equation

The first law of thermodynamics is a statement of conservation of energy. The mathematical statement of the first law of thermodynamics suggests that the rate of change of energy of a fluid particle is equal to the rate of work done on the particle plus the rate of heat addition to the fluid particle. Again, *y*- and *z*-components have similar form as that of the *x*-component. Thus, only derivation of the *x*-component is shown.

The product of the force and velocity component in the direction of the force equals the rate of work done on the fluid particle in the element by a surface force. Using the result from the momentum equation, Equation 31 gives the work done by these forces.

$$\begin{aligned}
&\left[\left(pu-\frac{\partial(pu)}{\partial x}\cdot\frac{1}{2}\delta x\right)-\left(\tau_{xx}u-\frac{\partial(\tau_{xx}u)}{\partial x}\cdot\frac{1}{2}\delta x\right)\right]\delta y\delta z\\
&-\left[\left(pu+\frac{\partial(pu)}{\partial x}\cdot\frac{1}{2}\delta x\right)+\left(\tau_{xx}u+\frac{\partial(\tau_{xx}u)}{\partial x}\cdot\frac{1}{2}\delta x\right)\right]\delta y\delta z\\
&+\left[-\left(\tau_{yx}u+\frac{\partial(\tau_{yx}u)}{\partial y}\cdot\frac{1}{2}\delta y\right)+\left(\tau_{yx}u+\frac{\partial(\tau_{yx}u)}{\partial y}\cdot\frac{1}{2}\delta y\right)\right]\delta x\delta z\\
&+\left[-\left(\tau_{zx}u+\frac{\partial(\tau_{zx}u)}{\partial z}\cdot\frac{1}{2}\delta z\right)+\left(\tau_{zx}u+\frac{\partial(\tau_{zx}u)}{\partial z}\cdot\frac{1}{2}\delta z\right)\right]\delta x\delta y
\end{aligned} \tag{31}$$

After some manipulation, the net rate of work done on the fluid particle by surface forces acting in the *x*-direction yield Equation 32. Equations 33 and 34 are the expressions for the additional rate of work done on the fluid particle due to the work done by the surface stress components in the *y*- and *z*-direction.

$$\left[\frac{\partial}{\partial x}\left[u\left(-p+\tau_{xx}\right)\right]+\frac{\partial}{\partial y}\left(u\tau_{yx}\right)+\frac{\partial}{\partial z}\left(u\tau_{zx}\right)\right]\delta x\delta y\delta z \tag{32}$$

$$\left[\frac{\partial}{\partial x}\left(v\tau_{xy}\right)+\frac{\partial}{\partial y}\left[v\left(-p+\tau_{yy}\right)\right]+\frac{\partial}{\partial z}\left(v\tau_{zy}\right)\right]\delta x\delta y\delta z \tag{33}$$

$$\left[\frac{\partial}{\partial x}\left(w\tau_{xz}\right)+\frac{\partial}{\partial y}\left(w\tau_{yz}\right)+\frac{\partial}{\partial z}\left[w\left(-p+\tau_{zz}\right)\right]\right]\delta x\delta y\delta z \tag{34}$$

Summing all the surface forces and dividing by the volume ($\delta x\delta y\delta z$) gives the total rate of work done per unit volume on the fluid particle. For compactness, collecting the terms containing pressure allows representation in vector form.

$$-\frac{\partial}{\partial x}(up)-\frac{\partial}{\partial y}(vp)-\frac{\partial}{\partial z}(wp)=-div\left(p\vec{u}\right) \tag{35}$$

This yields the following total rate of work done on the fluid particle by surfaces stresses.

$$\begin{aligned}&\frac{\partial\left(u\tau_{xx}\right)}{\partial x}+\frac{\partial\left(u\tau_{yx}\right)}{\partial y}+\frac{\partial\left(u\tau_{zx}\right)}{\partial z}\\&+\frac{\partial\left(v\tau_{xy}\right)}{\partial x}+\frac{\partial\left(v\tau_{yy}\right)}{\partial y}+\frac{\partial\left(v\tau_{zy}\right)}{\partial z}\\&+\frac{\partial\left(w\tau_{xz}\right)}{\partial x}+\frac{\partial\left(w\tau_{yz}\right)}{\partial y}+\frac{\partial\left(w\tau_{zz}\right)}{\partial z}\\&+\left[-div\left(p\vec{u}\right)\right]\end{aligned} \tag{36}$$

For the heat addition, consider the three components q_x, q_y, and q_z of the heat flux vector $\vec{q}$ shown in Figure 9. The difference between the rate of heat input across the left face and the rate of heat loss across the right face yields the net rate of heat transfer to the fluid particle due to heat flow in the *x*-direction (Equation 37). The net rates of heat

transfer to the fluid due to heat flows in the *y*- and *z*-direction have similar form (Equations 38 and 39, respectively).

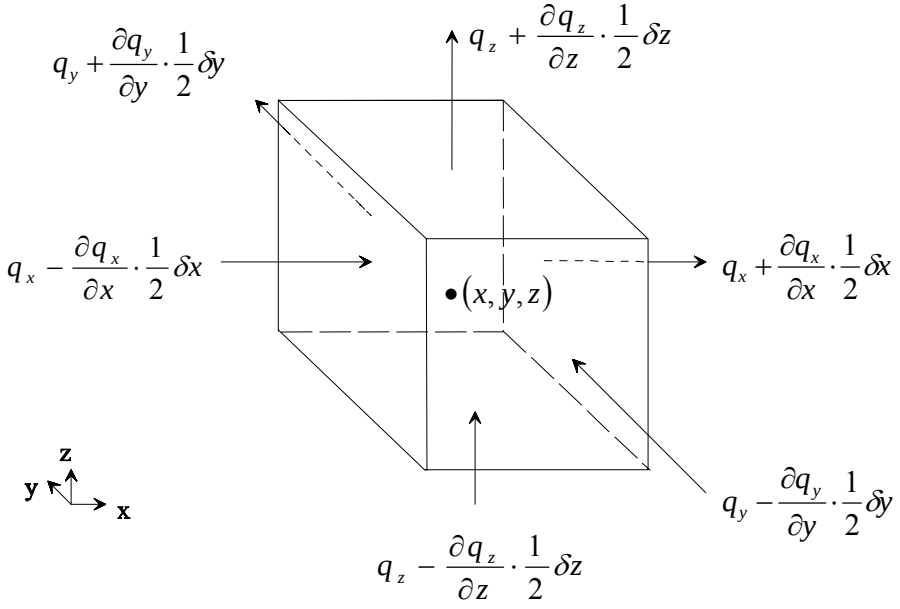


Figure 9. Heat Flux Vector

$$\left[\left(q_x - \frac{\partial q_x}{\partial x} \cdot \frac{1}{2}\delta x\right) - \left(q_x + \frac{\partial q_x}{\partial x} \cdot \frac{1}{2}\delta x\right)\right]\delta y \delta z = -\frac{\partial q_x}{\partial x}\delta x \delta y \delta z \qquad \textbf{(37)}$$

$$\left[\left(q_y - \frac{\partial q_y}{\partial y} \cdot \frac{1}{2}\delta y\right) - \left(q_y + \frac{\partial q_y}{\partial y} \cdot \frac{1}{2}\delta y\right)\right]\delta x \delta z = -\frac{\partial q_y}{\partial y}\delta x \delta y \delta z \qquad \textbf{(38)}$$

$$\left[\left(q_z - \frac{\partial q_z}{\partial z} \cdot \frac{1}{2}\delta z\right) - \left(q_z + \frac{\partial q_z}{\partial z} \cdot \frac{1}{2}\delta z\right)\right]\delta x \delta y = -\frac{\partial q_z}{\partial z}\delta x \delta y \delta z \qquad \textbf{(39)}$$

The sum of the x-, y, and z-component expressions divided by the volume *($\delta x \delta y \delta z$)* yields the total rate of heat added to the fluid particle per unit volume due to heat flow across its boundaries. For compactness, collecting the terms containing heat flux components allows representation in vector form (Equation 40).

$$-\frac{\partial}{\partial x}q_x - \frac{\partial}{\partial y}q_y - \frac{\partial}{\partial z}q_z = -div\vec{q} \qquad \textbf{(40)}$$

Application of Fourier's law of heat conduction relates the heat flux to the local temperature gradient Equation 41, [written in vector form: $\vec{q} = div(k_c \nabla T)$].

$$q_x = -k_c \frac{\partial T}{\partial x} \qquad q_y = -k_c \frac{\partial T}{\partial y} \qquad q_z = -k_c \frac{\partial T}{\partial z} \qquad \textbf{(41)}$$

The specific energy (E) of a fluid by definition is the sum of internal (thermal) energy (i_t), kinetic energy $\frac{1}{2}(u^2 + v^2 + w^2)$ and gravitational potential energy. Combining these into S_E, allows representation of the source of energy per unit volume per unit time. Equating the rate of change of energy of the fluid particle to the sum of the net rate of work done on the fluid particle, the net rate of heat addition to the fluid, and the rate of increase of energy due to sources yields the conservation of energy equation (Equation 42).

$$\begin{aligned}\rho\frac{DE}{Dt} = -div(p\vec{u}) + div(k\nabla T) + S_E + \frac{\partial(u\tau_{xx})}{\partial x} + \frac{\partial(u\tau_{yx})}{\partial y} + \frac{\partial(u\tau_{zx})}{\partial z} \\ + \frac{\partial(v\tau_{xy})}{\partial x} + \frac{\partial(v\tau_{yy})}{\partial y} + \frac{\partial(v\tau_{zy})}{\partial z} + \frac{\partial(w\tau_{xz})}{\partial x} + \frac{\partial(w\tau_{yz})}{\partial y} + \frac{\partial(w\tau_{zz})}{\partial z}\end{aligned} \qquad \textbf{(42)}$$

Equation of State

As displayed in Table 4, a system of five partial differential equations (mass conservation, x-, y-, z-momentum equations, and energy equation) describes the motion of a fluid in three dimensions. Four thermodynamic variables (ρ, p, E and T) represent the unknowns. The thermodynamic equilibrium assumption provides linkage for the

unknowns. Under this assumption and the use of just two state variables, description of the state of a substance is possible. Equations of state relate the other variables to the two state variables. For instance, using density (ρ) and temperature (T) as state variables, the equation of state relates pressure (p) as a function of density and temperature: $p = p(\rho, T) = \rho RT$. This is the well-known equation of state for a perfect gas.

In the flow of compressible fluids, equations of state provide the link for the energy equation. Equations of state also provide the link for mass conservation and momentum equations. This link arises due to density variations resulting from pressure and temperature variations in the flow field.

Auxiliary Conditions

Boundary conditions for a compressible viscous flow typically involve specifying no slip conditions at solid walls. For fluid boundaries, either temperature or heat flux are fixed. Specification at inlets includes density, magnitude of velocity vector components, and temperature as a function of position. The outflow condition universally used in the finite volume method is to specify pressure. Figure 10 illustrates the application of boundary conditions for a typical internal flow.

FLUENT, like most general purpose CFD codes also include inlet as well as outlet pressure boundary conditions for compressible flow. The pressures are set at fixed values with placement of sources and/or sinks of mass on the boundaries carry the

correct mass flow into and out of the solution zone across the constant pressure boundaries.

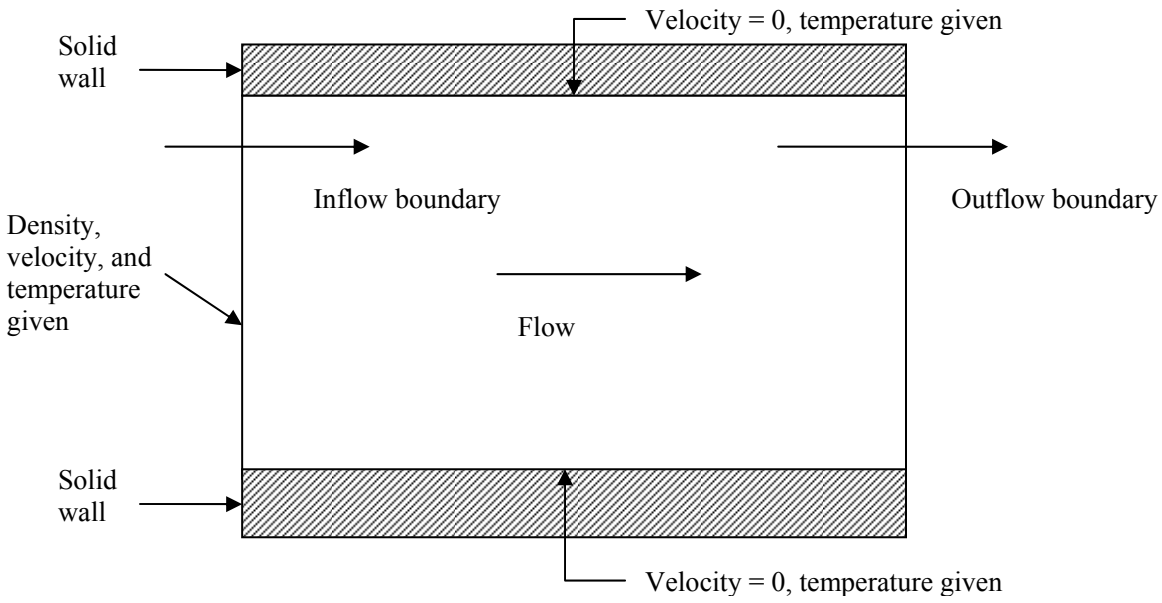


Figure 10. Boundary Conditions: Internal Flow

Modeling Approach

In this analysis, implementation of an Eulerian/Lagrangian approach afforded solution of our aerosol-laden gas (two-phase) flow problem. The Eulerian model describes gas (continuous phase) behavior while the Lagrangian model predicts particle trajectories (dispersed phase). Two turbulence models (realizable k-ε and standard k-ω) yielded results of the continuous phase turbulence phenomena. For our wall-bounded internal flow simulated with the realizable k-ε turbulence model, non-equilibrium wall functions link bulk flow conditions to flow behavior in the wall adjacent cells in the computational domain. Using the standard k-ω model, the two-zonal near-wall treatment provided this link. The three scenarios investigated for particle trajectory

predictions were simulations with and without turbulent dispersion activated in the discrete random walk (DRW) model. The third scenario involved use of the modified particle trajectory algorithm that aided in the prediction of aerosol deposition the pipe configuration. Particle sizes used in the particle trajectory simulations ranged from one to twelve microns (1 – 12 μm). For comparison with the ARTIST test data only particle sizes less than 8 microns were used to compute decontamination factors (DF).

Eulerian Approach for the Continuous Phase

Equation 43 represents the general continuity equation solved in FLUENT.

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0 \qquad \textbf{(43)}$$

The continuity equation equated to zero means the dispersed second phase adds no mass to the continuous phase and there are no user-defined sources. Equation 44, the general momentum equation solved in FLUENT differs slightly from the form presented previously.

$$\frac{\partial}{\partial t}(\rho \vec{u}) = -\nabla p + \nabla \cdot \mu \left[\left(\nabla \vec{u} + \nabla \vec{u}^{T} \right) - \frac{2}{3} \nabla \cdot \vec{u} I \right] - \nabla \cdot (\rho \vec{u} \vec{u}) + \rho \vec{g} + \vec{F} \qquad \textbf{(44)}$$

where

p	=	Static pressure
$\vec{u}$	=	Velocity vector
$\rho \vec{g}$	=	Gravitational body force
$\rho \vec{u} \vec{u}$	=	Reynolds stresses
$\vec{F}$	=	External body forces
μ	=	Molecular viscosity
I	=	Unit tensor

The second term on the right hand side of the momentum equation, the gradient of the stress tensor includes the effect of volume dilation. In a compressible fluid, the relative change in volume is equal to the negative of the relative rate of change in the local density, volumetric dilation. The general energy equation (Equation 45) solved in FLUENT also has a slightly different form than previously presented.

$$\frac{\partial}{\partial t}(\rho E)+\nabla\cdot\left(\vec{u}(\rho E+p)\right)=\nabla\cdot\left(k_{eff}\nabla T-\sum_{j}h_{j}\vec{J}_{j}+\bar{\bar{\tau}}_{eff}\cdot\vec{u}\right)+S_{h} \quad \textbf{(45)}$$

where
h = Sensible enthalpy
k_{eff} = Effective conductivity, $(k + k_t)$
$\vec{J}$ = Diffusion flux
S = Volumetric heat sources

The effective conductivity (k_{eff}) is the sum of thermal conductivity (k) and the turbulent thermal conductivity (k_t). The definition of turbulent thermal conductivity depends on the turbulence model used. The first three terms on the right hand side of Equation 45 represent energy transfer due to conduction, species diffusion, and viscous dissipation, respectively. The viscous dissipation terms in the energy equation describes the thermal energy created by viscous shear in the flow. Viscous heating is important when the Brinkman number, *Br* (Equation 46) approaches or exceeds unity. Qualitatively, the Brinkman number represents the ratio of dissipation effects to fluid conduction effects.[34] Compressible flows typically have Brinkman numbers of order unity or greater.

$$Br=\frac{\mu U_{e}^{2}}{k\Delta T} \quad \textbf{(46)}$$

Viscous dissipation terms are not automatically included in the energy equation when the segregated solver is used. Enabling viscous heating in the solver panel includes the viscous dissipation terms in the energy equation.

Turbulence Models

The modeling procedure involves computing the flow Reynolds number to determine if turbulence modeling is necessary. Assuming turbulence modeling is necessary; selection of the appropriate turbulence model requires engineering judgment. Key factors to consider in turbulence model selection include but are not limited to the available turbulence models and near wall treatment for solution of flow physics. Equally important are the available computational resources, accuracy required, and the turn around time.

Figure 11 summarizes the various types of turbulence models: Reynolds Averaged Navier-Stokes (RANS); Large Eddy Simulation (LES); and Direct Numeric Simulation (DNS). The bold outlined box indicates the models available in standard CFD codes (e.g., FLUENT, CFX., STAR-CD, FLOW-3D). Table 5 summarizes RANS turbulence models in terms of behavior and provides guidance for situations where their use is appropriate. The model deemed to provide superior performance for wall bounded boundary layer flows and most suitable for complex boundary layer flows under adverse pressure gradient and separation are the realizable k-ε and the standard k-ω turbulence models.

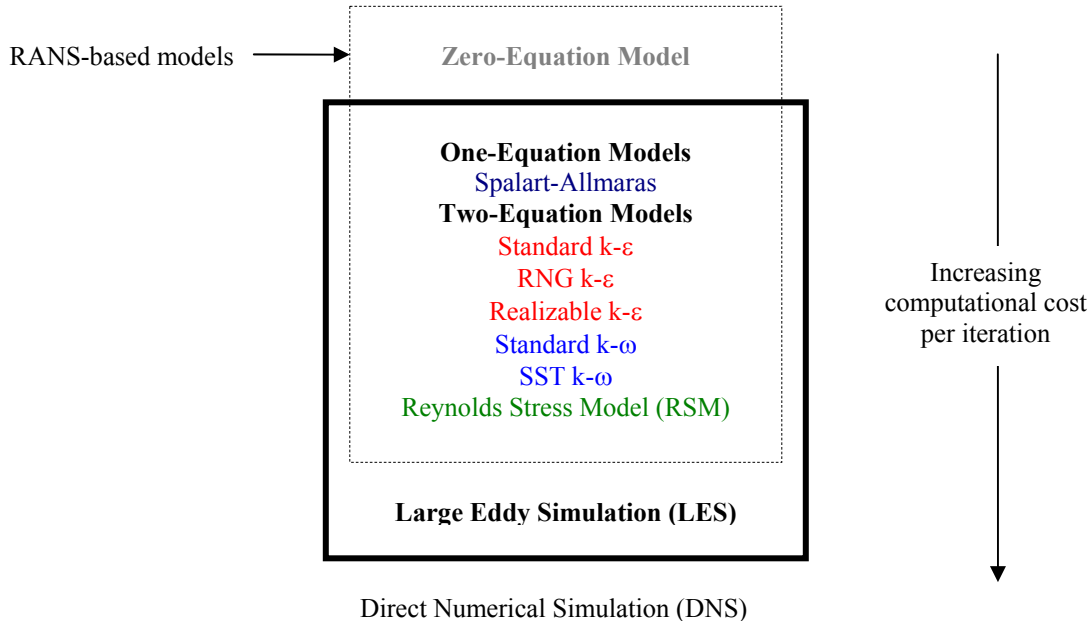


Figure 11. Turbulence Models

Generally, the solution procedure begins with modeling turbulence with the standard k-ε and changing to RNG k-ε, realizable k-ε, standard k-ω, or SST k-ω, if needed. For these turbulence models, the use of wall functions is recommended unless low-Re and/or complex near-wall physics are present.

Turbulent flow is characterized by fluctuating velocity fields, which are incorporated in the Reynolds stress tensor. For the k-ε and k-ω models, the Reynolds stresses $\left(-\rho_i \overline{u_i' u'}_i\right)$ employ the Boussinesq hypothesis[35] to relate the Reynolds stresses to the mean velocity gradients, while for RSM, the Reynolds stresses are solved in a transport equation for each of the terms in the Reynolds stress tensor. Detailed derivation of turbulence models can be found in other references.[36, 37, 38, 39]

Table 5. RANS Turbulence Model Behavior and Usage

Model	Behavior and Usage
Spalart-Allmaras	Economical for large meshes. Performs poorly for 3D flows, free shear flows, flows with strong separation. Suitable for mildly complex (quasi-2D) external/internal flows and boundary layer flows under pressure gradient (e.g., airfoils, wings, airplane fuselage, missiles, ship hulls).
Standard k-ε	Robust. Widely used despite the known limitations of the model. Performs poorly for complex flows involving severe ∇p, separation, strong streamline curvature. Suitable for initial iterations initial screening of alternative designs, and parametric studies.
RNG k-ε	Suitable for complex shear flows involving rapid strain, moderate swirl, vortices, and locally transitional flows (e.g., boundary layer separation, massive separation and vortex-shedding behind bluff bodies, stall in wide angle diffusers, room ventilation).
Realizable k-ε	Offers largely the same benefits and has similar applications as RNG. Unable to use multiple rotating reference frames. Possibly more accurate and easier to converge than RNG.
Standard k-ω	Superior performance for wall bounded boundary layer, free shear, and low Re flows. Suitable for complex boundary layer flows under adverse pressure gradient and separation (external aerodynamics and turbo-machinery). Can be used for transitional flows though tends to predict early transition). Separation is typically predicted to be excessive and early.
SST k-ω	Similar benefits as standard k-ω. Dependency on wall distance makes this less suitable for free shear flows.
RSM	Physically the most sound RANS model. Avoids isotropic eddy viscosity assumption. More CPU time and memory required. Tougher to converge due to close coupling of equations. Suitable for complex 3D flows with strong streamline curvature, strong swirl/rotation (e.g., curved duct, rotating flow passages, swirl combustors with very large inlet swirl, cyclones).

To render the Navier-Stokes equations tractable methods such as Reynolds-averaging (ensemble-averaging) or filtering are employed so that the small-scale turbulent fluctuations do not have to be directly simulated.

The Reynolds-averaged Navier-Stokes (RANS) equations govern the transport of the averaged flow quantities. In Reynolds averaging, the solution variables in the instantaneous (exact) Navier-Stokes equations are decomposed into the mean (ensemble-averaged or time-averaged) and fluctuating components. For the velocity components:

$$u_i = \bar{u}_i + u_i' \tag{47}$$

$\bar{u}_i$ and u_i' are the mean and fluctuating velocity components $(i = 1, 2, 3)$. Letting ϕ denote scalar quantities (e.g., pressure, energy, turbulence) facilitates expression of the instantaneous value of the scalar as the mean plus fluctuation value of the scalar (Equation 48).

$$\phi = \bar{\phi} + \phi' \tag{48}$$

Substituting expressions of this form for the flow variables into the instantaneous continuity and momentum equation and taking a time (or ensemble) average (an dropping the overbar on the mean velocity, $\bar{u}$) yields the ensemble-averaged momentum equations. The Reynolds-averaged Navier-Stokes (RANS) equations in Cartesian tensor form are as follows.

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}\left(\rho u_i\right) = 0 \tag{49}$$

$$\frac{\partial}{\partial t}\left(\rho u_i\right) + \frac{\partial}{\partial x_j}\left(\rho u_i u_j\right) = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial u_l}{\partial x_l}\right)\right] + \frac{\partial}{\partial x_j}\left(-\rho\overline{u_i' u_j'}\right) \tag{50}$$

This formulation of the RANS equations has the same general form as the instantaneous Navier-Stokes equations. The velocities and other solution variables are ensemble-averaged values. The Reynolds stresses $\left(-\rho\overline{u_i' u_j'}\right)$ appear as an additional term requiring modeling in order to close the ensemble-averaged momentum equation. For compressible flow, the continuity and momentum equations can be interpreted as Favre-averaged[35] Navier-Stokes equations, with the velocities representing mass-averaged values.

Standard k-ε Model

Derivation of the standard k-ε model assumes the flow is fully turbulent and the effects of molecular viscosity are negligible.[36] For locations near walls, the standard k-ε model, therefore, demands an additional model, which comprises the effects of molecular viscosity. In this situation, wall functions based on semi-empirical formulas and functions are usually employed.

The standard k-ε model[40] is a two-equation model, one equation for kinetic energy (k) and the other for dissipation rate (ε). The standard k-ε model uses the following transport equations for k and ε:

$$\frac{\partial(\rho k)}{\partial t} + div\left(\rho k \vec{U}\right) = div\left[\frac{\mu_t}{\sigma_k}\nabla k\right] + 2\mu_t E_{ij} \bullet E_{ij} - \rho\varepsilon \qquad \textbf{(51)}$$

$$\frac{\partial(\rho\varepsilon)}{\partial t} + div\left(\rho\varepsilon \vec{U}\right) = div\left[\frac{\mu_t}{\sigma_\varepsilon}\nabla\varepsilon\right] + C_{1\varepsilon}\frac{\varepsilon}{k}2\mu_t E_{ij} \bullet E_{ij} - C_{2\varepsilon}\rho\frac{\varepsilon^2}{k} \qquad \textbf{(52)}$$

The production term in the model k-equation is derived from the exact production term. The turbulent transport terms are represented using the gradient diffusion concept. Prandtl numbers σ_k and σ_ε connect the diffusivities of k and ε to the eddy viscosity (μ_t). The pressure term of the exact k-equation cannot be measured directly. Its effect is accounted for within the gradient diffusion term. Production and destruction of turbulent kinetic energy are always closely linked. The dissipation rate is large where production of turbulent kinetic energy is large. The model equation for the dissipation rate assumes that its production and destruction terms are proportional to the production and

destruction terms of the *k*-equation. Adoption of such forms ensures that ε increases rapidly if *k* increases rapidly and that it decreases sufficiently fast to avoid non-physical (negative) values of turbulent kinetic energy if *k* decreases. In the ε-equation, the factor ε/k in the production and destruction terms makes these terms dimensionally correct.

The equations contain five adjustable constants C_μ, σ_k, σ_ε, $C_{1\varepsilon}$, and $C_{2\varepsilon}$. The standard k-ε model employs values for the constants that are arrived at by comprehensive data fitting for a wide range of turbulent flows.

$$C_\mu = 0.09; \qquad \sigma_k = 1.00; \qquad \sigma_\varepsilon = 1.30; \qquad C_{1\varepsilon} = 1.44; \qquad C_{2\varepsilon} = 1.92 \qquad \textbf{(53)}$$

To compute the Reynolds stresses with the k-ε model an extended Boussinesq relationship is used.

$$-\rho\overline{u_i' u_j'} = \mu_t\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) - \frac{2}{3}\rho k\delta_{ij} = 2\mu_t E_{ij} - \frac{2}{3}\rho k\delta_{ij} \qquad \textbf{(54)}$$

The Kronecker delta (δ_{ij}) has its standard definition (i.e., $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$). The term serves to make the formula applicable to the normal Reynolds stresses for which $i = j$, and hence for $\tau_{xx} = -\rho\overline{u'^2}$, $\tau_{yy} = -\rho\overline{v'^2}$ and $\tau_{zz} = -\rho\overline{w'^2}$.

By taking into consideration that at high Reynolds numbers the rate at which large eddies extract energy from the mean flow is precisely matched to the rate of transfer of energy across the energy spectrum is small, dissipating eddies, the small eddy variable ε can be used in the large eddy scale (l) definition. Use is made of *k* and ε to define the velocity scale (ϑ) and length scale (l) representative of large-scale turbulence.

$$\vartheta = \sqrt{k} \tag{55}$$

$$l = \frac{\sqrt[3]{k}}{\varepsilon} \tag{56}$$

The eddy viscosity is represented as follows:

$$\mu_t = C\rho\vartheta l = \rho C_\mu \frac{k^2}{\varepsilon} \tag{57}$$

C_μ is a dimensionless constant.

The model equations for turbulent kinetic energy and its dissipation rate are elliptic by virtue of the gradient diffusion term. Their behavior gives rise to the need for the following boundary conditions.

- Inlet: distributions of k and ε must be given
- Outlet: $\partial k/\partial n = 0$ and $\partial \varepsilon/\partial n = 0$

  (n is the local coordinate normal to the wall)
- Solid walls: approach depends on Reynolds number

If no information is available at all, crude approximations for the inlet distributions for k and ε in internal flows can be obtained from the turbulence intensity I_i and a characteristic length L of the equipment (equivalent pipe radius) by means of the following simple formulations.

$$k = \frac{3}{2}\left(U_{ref} I_i\right)^2 \tag{58}$$

$$\varepsilon = C_\mu^{3/4} \frac{k^{3/2}}{l} \tag{59}$$

$$l = 0.07L \tag{60}$$

At high Reynolds number, the standard k-ε model[45] avoids the need to integrate the model equations right through to the wall by making use of the universal behavior of near wall flows. If y is the coordinate direction normal to the solid wall, the mean velocity at a point y_p with $30 < y_p^+ < 500$ satisfies the log-law and measurements of turbulent kinetic energy budgets indicate that the rate of turbulence production equals the rate of dissipation. Using these assumptions and the eddy viscosity formula it is possible to develop the following wall functions.

$$u^+ = \frac{U}{u_\tau} = \frac{1}{\kappa}\ln\left(Ey_p^+\right) \tag{61}$$

$$k = \frac{u_\tau^2}{\sqrt{C_\mu}} \tag{62}$$

$$\varepsilon = \frac{u_\tau^3}{\kappa y} \tag{63}$$

von Karman's constant $\kappa = 0.42$ and the wall roughness parameter $E = 9.8$ for smooth walls. Schlichting[33] gives values of E that are valid for rough walls.

In terms of assessment of performance, the advantages of use of the k-ε model is that it is a simple turbulence model for which only initial and/or boundary conditions need to be supplied; excellent performance for many industrially relevant flows; and is well established (the most widely validated turbulence model). However, the model has problems in swirling flows and flows with large, rapid, extra strains (e.g., highly curved boundary layers) since it does not contain a description of the subtle effects of streamline curvature on turbulence. Secondary flows in long non-circular ducts, which are driven

by anisotropic normal Reynolds stresses, also cannot be predicted owing to the deficiencies of the treatment of normal stresses within the k-ε model.

Realizable k-ε Model (rke)

There are two main differences between the realizable k-ε model and the standard k-ε model, the eddy-viscosity formulation, and the turbulence energy dissipation developed based on the dynamic equation of the mean square vorticity fluctuation. Equations 64 and 65 represent the conservation equation of turbulence kinetic energy and its specific dissipation, respectively. Constants are defined in Equations 66 and 67.

$$\frac{\partial}{\partial t}(\rho k)+\frac{\partial}{\partial x_i}(\rho k u_j)=\frac{\partial}{\partial x_i}\left[\left(\mu+\frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right]+G_k+G_b-\rho\varepsilon-Y_M+S_k \qquad \textbf{(64)}$$

$$\frac{\partial}{\partial t}(\rho \varepsilon)+\frac{\partial}{\partial x_i}(\rho \varepsilon u_j)=\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right]+\rho C_1 S_\varepsilon-\rho C_2\frac{\varepsilon^2}{k+\sqrt{\nu\varepsilon}}+C_{1\varepsilon}\frac{\varepsilon}{k}C_{3\varepsilon}G_b+S_\varepsilon \qquad \textbf{(65)}$$

$$C_1=\max\left(0.43,\frac{\eta}{\eta+5}\right) \qquad \eta=Sk\frac{k}{\varepsilon} \qquad \textbf{(66)}$$

$$C_{1\varepsilon}=1.44 \qquad C_2=1.9 \qquad \sigma_k=1.0 \qquad \sigma_\varepsilon=1.2 \qquad \textbf{(67)}$$

G_k = Generation of turbulence kinetic energy due to the mean velocity gradients
G_b = Generation of turbulence kinetic energy due to buoyancy
Y_M = Contributions of the fluctuating dilations in compressible turbulence to the overall dissipation rate
σ_k, σ_ε = Turbulent Prandtl numbers for k and ε
S_k, S_ε = User-defined source terms
S = Modulus of the mean rate-of-strain tensor

Shih[38] and colleagues provide detailed derivations for the closure equations.

The realizable k-ε model touts to perform well for flows involving rotation, boundary layers under strong adverse pressure gradients, separation, and recirculation. Non-equilibrium wall functions are recommended for use in flows involving separation, reattachment, and impingement where the mean flow and turbulence are subjected to severe pressure gradients and change rapidly.

Standard k-ω Model

Derivation of the standard k-ω model[39] is based on transport equations (Equations 68 and 69) for turbulence kinetic energy (k) and the specific dissipation rate (ω, ratio of ε to k).

$$\frac{\partial}{\partial t}(\rho k)+\frac{\partial}{\partial x_i}(\rho k u_i)=\frac{\partial}{\partial x_j}\left(\Gamma_k \frac{\partial k}{\partial x_j}\right)+G_k-Y_k+S_k \qquad \textbf{(68)}$$

$$\frac{\partial}{\partial t}(\rho \omega)+\frac{\partial}{\partial x_i}(\rho \omega u_i)=\frac{\partial}{\partial x_j}\left(\Gamma_\omega \frac{\partial \omega}{\partial x_j}\right)+G_\omega-Y_\omega+S_\omega \qquad \textbf{(69)}$$

G_k, G_ω = Turbulence kinetic energy or ω production due to mean velocity gradients
Γ_k, Γ_ω = Effective diffusivity of k $\left(=\mu+\mu_t/\sigma_k\right)$ or ω $\left(=\mu+\mu_t/\sigma_\omega\right)$
Y_k, Y_ω = Dissipation of k or ω
S_k, S_ω = User-defined source terms for k or ω
μ_t = Turbulent viscosity $\left(=\alpha^* \rho k/\omega\right)$

The model incorporates modifications for low-Reynolds number effects, compressibility, and shear flow spreading. The coefficient α^* (Equation 70) damps the turbulent viscosity causing a low-Reynolds number correction. Equation 71 shows the

damping coefficient constants. For the high-Reynolds number form of the *k*-ω model $\alpha^* = \alpha_\infty^* = 1$. This coefficient is also used in the production of ω. For the high-Reynolds number form of the *k*-ω model $\alpha = \alpha_\infty = 1$.

$$\alpha^* = \alpha_\infty^* \left(\frac{\alpha_0^* + \mathrm{Re}_t / R_k}{1 + \mathrm{Re}_t / R_k} \right) \tag{70}$$

$$\mathrm{Re}_t = \frac{\rho k}{\mu \omega} \qquad R_\omega = 2.95 \qquad R_k = 6 \;\; \alpha_0^* = \beta_i / 3 \qquad \beta_i = 0.072 \tag{71}$$

$$\alpha = \frac{\alpha_\infty}{\alpha^*} \left(\frac{\alpha_0 + \mathrm{Re}_t / R_\omega}{1 + \mathrm{Re}_t / R_\omega} \right) \tag{72}$$

Near-Wall Modeling

In the region near the wall, the gradient of quantities (e.g., velocity) is considerably large and requires a fine grid close to the wall to capture changes in the quantities. This causes the calculation to become more computationally expensive (i.e., time-consuming), requiring greater memory and faster processing, as well as expensive in terms of complexity of equations. A wall function, which is a collection of semi-empirical formulas and functions, provides a cheaper computational method by substituting the fine grid with a set of equations linking solution variables at near-wall cells and the corresponding quantities on the wall.

For complex flows where flow separation and reattachment occur, the standard wall function proposed by Launder and Spalding[40] becomes less reliable. The non-equilibrium wall function proposed by Kim and Choudhury[41] has proven to give better predictions because it accounts for the effects of pressure gradient and departure from

equilibrium. The k-ε models and the Reynolds stress model employ the non-equilibrium wall function model. For the k-ω models, accounting for the low-Reynolds number effects requires enabling translational flow, and then the near-wall grid has to be very fine to obtain better results for near wall modeling.

Law of the Walls

Walls are the main source of vorticity and turbulence. Thus, accurate near-wall modeling is of paramount importance. For instance, the successful prediction of pressure drop for internal flows depends on the fidelity of local wall shear predictions. The basis for modeled terms (e.g., diffusion, dissipation, and pressure-strain redistribution) is isotropic behavior. However, for internal wall-bounded flows the near-wall flow is anisotropic due to the presence of walls. Special near-wall treatments are necessary when equations cannot be integrated down to the wall. Flow behavior in the near-wall region as seen in the velocity profile (Figure 12) exhibits layer structure (identified from dimensional analysis). The flow domain is subdivided into three regions: laminar sublayer, buffer layer, and the turbulent core. The two-equation model k-ε and its variants are valid in the turbulent core region and through the log layer (buffer layer).

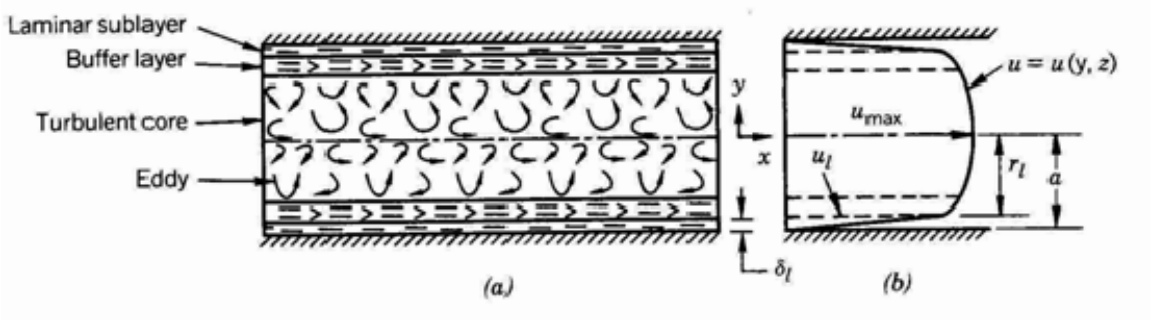


Figure 12. Boundary Layer

Viscous forces are dominate in the inner layer (sublayer), outer layer (turbulent core) depends upon mean flow, and the overlap layer (buffer) log-law applies. In the overlap, turbulent kinetic energy production and dissipation are nearly equal. As seen in Figure 13, dissipation is much greater than turbulent kinetic energy production in the sublayer.

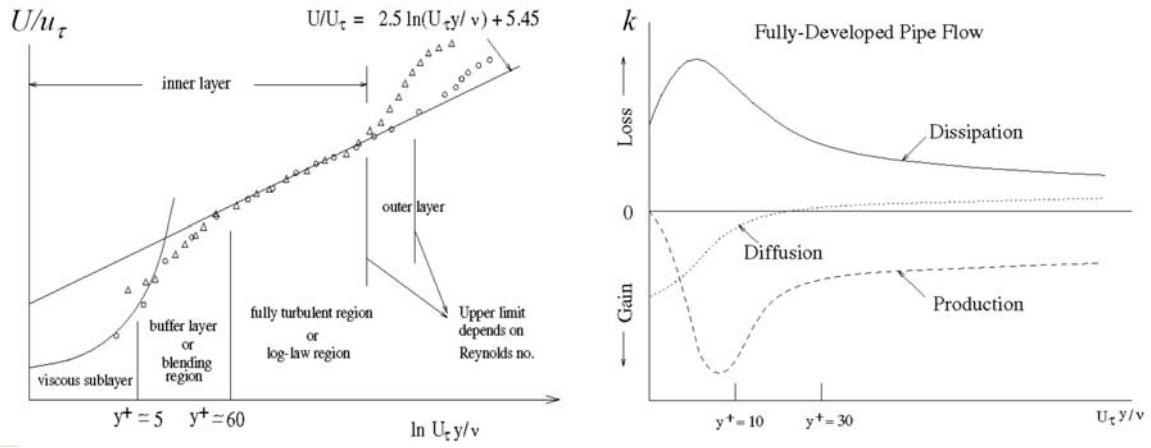
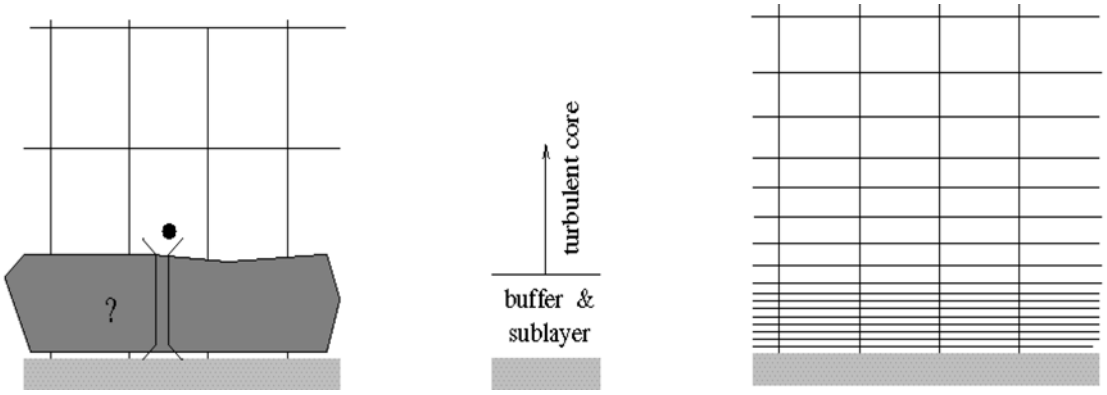


Figure 13. Flow Behavior in Near-Wall Regions

In the wall function approach (Figure 14 a), the viscosity-affected region is not resolved. Instead, the location of the wall-adjacent cell centroid is placed in the log layer (expressed in terms of wall unit, $y^+ \equiv \rho u_\tau y/\mu = 30$). The cell centroid information

provides the bridge for applicability of the various empirical formulations for the specific wall regions. Thus, wall functions use the law of the wall to supply boundary conditions for turbulent flows. The near-wall mesh can be relatively coarse.



(a) (b)

Figure 14. Near-Wall Modeling Approaches

Wall functions consist of wall laws for mean velocity and temperature, and formulas for turbulent quantities. The universal wall laws, Equation 73, (Lauder-Spalding wall functions[42]) results from dimensional analysis employing density, relaxation time, viscosity, shortest distance to the nearest wall, and turbulence kinetic energy (ρ, τ, μ, y, and k, respectively).

$$U^* = \begin{cases} y^* & y^* < y_\nu^* \\ \frac{1}{\kappa}\ln\left(Ey^*\right) & y^* > y_\nu^* \end{cases} \quad \textbf{(73)}$$

$$y^* = \frac{\rho C_\mu^{\frac{1}{4}} k_p^{\frac{1}{2}} y_p}{\mu} \quad \textbf{(74)}$$

$$U^* = \frac{U_p C_\mu^{\frac{1}{4}} k_p^{\frac{1}{2}}}{\tau_w / \rho} \quad \textbf{(75)}$$

κ = von Kármán constant (= 0.42)
E = Empirical constant (= 9.793)
U_p = Mean velocity of the fluid at point P
k_p = Turbulence kinetic energy at point P
y_p = Distance from point P to the wall
μ = Dynamic viscosity of the fluid

The logarithmic law for mean velocity is known to be valid for wall unit, (y^*) greater than about 30 to 60. In FLUENT, the log-law is employed when $y^* > 11.225$. On a coarse mesh and $y^* < 11.225$ at the wall-adjacent cells, FLUENT applies the laminar stress-strain relationship.

Formulas for turbulent kinetic energy (k) and its dissipation (ε) assumes local turbulent equilibrium (production = dissipation) prevails in the log layer. The turbulence kinetic energy transport equation applies when $\nabla k \cdot \hat{n} = 0$ at the surface.

$$\varepsilon = C_\mu^{\frac{3}{4}} k^{\frac{3}{2}} / (\kappa y) \quad \textbf{(76)}$$

Invalidation of the local equilibrium assumption renders wall functions less reliable. This assumption fails in the case of highly three-dimensional flow and when rapidly changing fluid properties occur near the wall.

Non-Equilibrium Wall Function

The non-equilibrium wall function takes into account the impact of mean flow and turbulence in complex flows involving separation and reattachment. The non-equilibrium wall function incorporates Launder and Spalding's log-law for mean

velocity sensitized to pressure gradient effects ($\widetilde{U}$) and a two-layer based concept adopted to compute the budget of turbulence kinetic energy ($\overline{G_k}, \bar{\varepsilon}$) in the wall-neighboring cells. The local equilibrium assumptions for the turbulence kinetic energy are relaxed in the wall neighboring cells.

$$\frac{\widetilde{U}C_\mu^{\frac{1}{4}}k^{\frac{1}{2}}}{\tau_w/\rho} = \frac{1}{\kappa}\ln\left(E\frac{\rho C_\mu^{\frac{1}{4}}k^{\frac{1}{2}}y}{\mu}\right) \tag{77}$$

$$\widetilde{U} = U - \frac{1}{2}\frac{dp}{dx}\left[\frac{y_\nu}{\rho\kappa^* k^{\frac{1}{2}}}\ln\left(\frac{y}{y_\nu}\right) + \frac{y - y_\nu}{\rho\kappa^* k^{\frac{1}{2}}} + \frac{y_\nu^2}{\mu}\right] \tag{78}$$

The physical viscous sublayer thickness (y_ν) is computed using a constant y_p^* (= 11.225) with the following expression.

$$y_\nu \equiv \frac{\mu y_p^*}{\rho C_\mu^{\frac{1}{4}}k^{\frac{1}{2}}} \tag{79}$$

As shown in Figure 15, the approach is to divide the flow domain into two regions: viscosity affected near-wall region and the fully turbulent core region. In Figure 15, y_n (= $2y_p$ for quadrilateral and hexahedral cells) represents the height of the cell.

Turbulence quantities are represented by profiles (Equations 80 - 82)[42] used to generate the cell-averaged production of turbulent kinetic energy ($\overline{G_k}$) and dissipation rate ($\bar{\varepsilon}$).

$$\tau_t = \begin{cases} 0 & y < y_v \\ \tau_w & y > y_v \end{cases} \qquad k = \begin{cases} (y/y_v)^2 k_p & y < y_v \\ k_p & y > y_v \end{cases} \qquad \varepsilon = \begin{cases} 2\nu k/y^2 & y < y_v \\ k^{\frac{3}{2}}/\left(\kappa C_\mu^{\frac{-3}{4}} y\right) & y > y_v \end{cases} \tag{80}$$

$$\overline{G_k} = \frac{1}{y_n}\int_0^{y_n} \tau_t \frac{\partial U}{\partial y} dy = \frac{1}{\kappa y_n} \frac{\tau_w^2}{\rho C_\mu^{\frac{1}{4}} k_p^{\frac{1}{2}}} \ln\left(\frac{y_n}{y}\right) \tag{81}$$

$$\bar{\varepsilon} = \frac{1}{y_n}\int_0^{y_n} \varepsilon dy = \frac{1}{y_n}\left[\frac{2\nu}{y_v} + \frac{k_p^{\frac{1}{2}}}{\kappa C_\mu^{\frac{1}{4}}} \ln\left(\frac{y_n}{y}\right)\right] k_p \tag{82}$$

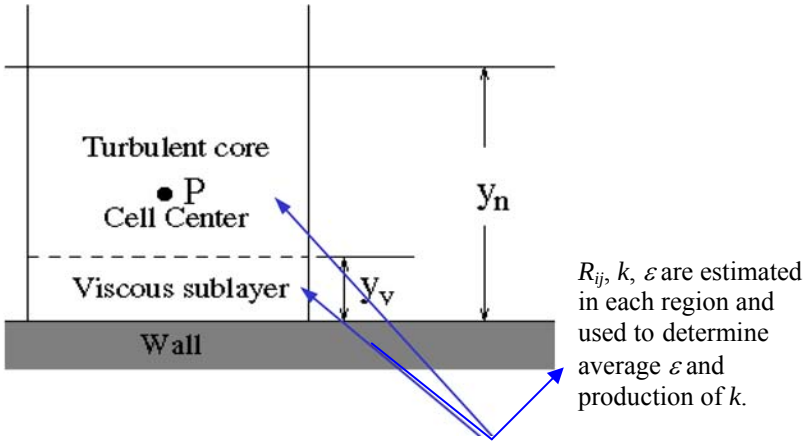


Figure 15. Non-Equilibrium Wall Function: Turbulence Kinetic Energy Budget

Enhanced Wall Treatment

In the two-layer zonal enhanced wall treatment approach, the near-wall region resolution extends down to the wall. Thus, this type of near-wall modeling method requires a relatively fine mesh (Figure 14 b). The initial wall-adjacent cell centroid location is assumed to be in the viscous sublayer ($y^+ = 1$). In addition, the recommendation is to place at least ten cells within the viscosity-affected near-wall region in order to resolve the mean velocity and turbulent quantities in this region.

Each region utilizes a different turbulence model based on whether Re_y $\left(\equiv \rho y\sqrt{k}/\mu\right)$ exceeds 200 (turbulent core) or below 200 (viscosity-affected region). Equation 83 shows the formulation used to compute the normal distance from the wall at the cell center (y) as defined in the FLUENT Users Guide.

$$y \equiv \min_{\vec{r}_w \in \Gamma_w} \left\| \vec{r} - \vec{r}_w \right\| \tag{83}$$

$\vec{r}$ = Position vector at the field point
$\vec{r}_w$ = Position vector on the wall boundary
Γ_w = Union of all the wall boundaries involved

The one-equation model of Wolfstein[43] for the viscosity-affected near-wall region describes flow while a high Reynolds number (e.g., k-ε) model resolves flow in the turbulent core. In the one-equation model, the turbulent kinetic energy (k) in the near-wall is the same as that for the high-Re model. However, estimation of the two-layer turbulent viscosity ($\mu_{t,2\ layer}$) and turbulence dissipation rate employ different length scale formulations, l_μ and l_ε, respectively.

$$\mu_t = \rho C_\mu l_\mu \sqrt{k} \tag{84}$$

$$l_\mu = \kappa C_\mu^{\frac{3}{4}} y \left(1 - \exp\left(\frac{-\mathrm{Re}_y}{70} \right) \right) \tag{85}$$

The algebraic expressions for dissipation rate and associated length scale[44] are as follows.

$$\varepsilon = \frac{k^{\frac{3}{2}}}{l_\varepsilon} \tag{86}$$

$$l_\varepsilon = \kappa C_\mu^{\frac{3}{4}} y \left(1 - \exp\left(\frac{-\mathrm{Re}_y}{2\kappa C_\mu^{\frac{3}{4}}}\right)\right) \tag{87}$$

The high Reynolds number turbulent viscosity (μ_t) applies in the outer region. The blending function relations the two-layer and high Reynolds number turbulent viscosity subsequently assisting in solution convergence when the solutions in the two zones do not match.

$$\begin{aligned}\mu_{t,\,\text{enhanced}} = {} & \frac{\mu_t}{2}\left[1 + \tanh\left(\frac{\mathrm{Re}_y - \mathrm{Re}_y^*}{\left|\Delta \mathrm{Re}_y\right| / \tanh(0.98)}\right)\right] \\ & + \mu_{t,\,2\,\text{layer}}\left(1 - \frac{1}{2}\left[1 + \tanh\left(\frac{\mathrm{Re}_y - \mathrm{Re}_y^*}{\left|\Delta \mathrm{Re}_y\right| / \tanh(0.98)}\right)\right]\right)\end{aligned} \tag{88}$$

Lagrangian Approach for the Discrete Phase

The Lagrangian model employs the solved Eulerian equations for the fluid phase and then integrates Lagrangian equations of motion for the dispersed phase, tracking individual particles through the flow field. The discrete phase formulation contains the assumption that the second phase is sufficiently dilute that particle-particle interactions and the effects of the particle volume fraction on the gas phase are negligible. These issues imply that the discrete phase must be present at a low volume fraction, usually less than 10-12%. Thus, a one-way coupled model assumes that particle motion is

influenced by the continuous fluid phase, but the fluid phase is unaffected by the presence of the particles.

Computation of the trajectory of a particle is by integration of the force balance on the particle. As shown in Equation 89, the force balance equates the particle inertia with the force acting on the particle. When the flow is turbulent, most CFD codes predict the trajectories of particles using the mean fluid phase velocity, u, in the trajectory equations. Optionally, the instantaneous value of the fluctuating gas flow velocity can be included (Equation 47), to predict the dispersion of the particles due to turbulence.

Equation of Motion for Particles

Prediction of the trajectory of a discrete phase particle is by integrating the force balance on the particle. Equation 89 is the particle force balance equation written for the x direction in Cartesian coordinates.

$$\frac{du_p}{dt} = F_D\left(u - u_p\right) + \frac{g_x\left(\rho_p - \rho\right)}{\rho_p} + F_x \qquad \textbf{(89)}$$

$F_D(u - u_p)$ is the drag force per unit particle mass, Equation 90 shows its definition.

$$F_D = \frac{18\mu}{\rho_p d_p^2}\frac{C_D \,\mathrm{Re}}{24} \qquad \textbf{(90)}$$

Here, u is the fluid phase velocity, u_p is the particle velocity, μ is the molecular viscosity of the fluid, ρ is the fluid density, ρ_p is the density of the particle, and d_p is the particle diameter. Re is the relative Reynolds number, which is defined according to Equation 91.

$$\mathrm{Re} \equiv \frac{\rho d_p \left| u_p - u \right|}{\mu} \tag{91}$$

The drag coefficient, C_D, can be taken from either

$$C_D = a_1 + \frac{a_2}{\mathrm{Re}} + \frac{a_3}{\mathrm{Re}^2} \tag{92}$$

where a_1, a_2, and a_3 are constants that apply for smooth spherical particles over several ranges of Re given by Morsi and Alexander,[45] or

$$C_D = \frac{24}{\mathrm{Re}}\left(1 + b_1 \mathrm{Re}^{b_2}\right) + \frac{b_3 \mathrm{Re}}{b_4 + \mathrm{Re}} \tag{93}$$

where

$$b_1 = \exp\left(2.3288 - 6.4581\chi + 2.4486\chi^2\right) \tag{94}$$

$$b_2 = 0.0964 + 0.5565\chi \tag{95}$$

$$b_3 = \exp\left(4.905 - 13.8944\chi + 18.4222\chi^2 - 10.2599\chi^3\right) \tag{96}$$

$$b_4 = \exp\left(1.4681 + 12.2584\chi - 20.7322\chi^2 + 15.8855\chi^3\right) \tag{97}$$

which is taken from Haider and Levenspiel.[46] The shape factor, χ, is defined as

$$\chi = \frac{s}{S} \tag{98}$$

where s is the surface area of a sphere having the same volume as the particle, and S is the actual surface area of the particle.

For sub-micron particles, a form of Stokes' drag law is available.[47] In this case, F_D is defined as

$$F_D = \frac{18\mu}{\rho_p d_p^2 C_c} \tag{99}$$

The factor C_c is the Cunningham correction to Stokes' drag law, which is computed from

$$C_c = 1 + \frac{2\lambda}{d_p}\left(1.257 + 0.4e^{\frac{1.1 d_p}{2\lambda}}\right) \quad \textbf{(100)}$$

where λ is the molecular mean free path, the average distance traveled by a gas molecule between collisions with other molecules. In FLUENT's chapter on "Modeling Species Transport and Finite-Rate Chemistry", the molecular mean free path is defined in terms of Boltzmann's constant (k_B = 1.38066 x 10^{-23} J/K), temperature, pressure, and gas accommodation coefficient (σ): $\lambda = k_B T / \left(\sqrt{2}\pi\sigma^2 P\right)$. The gas accommodation coefficient is calculated as mass fraction weighted average of each gas species in the system, $\sigma = \sum_{i=1}^{N_g} Y_i \sigma_i$. The summation involves the mass fraction of the gas species (Y_i) and the Lennard-Jones characteristic length of the gas species (σ_i). Hirschfelder[48] (p. 209) gives the value for the Lennard-Jones characteristic length for nitrogen in angstroms as 3.698 Å.

A high-Mach-number drag law is also available. This drag law is similar to the spherical law

$$C_D = a_1 + \frac{a_2}{\mathrm{Re}} + \frac{a_3}{\mathrm{Re}^2} \quad \textbf{(101)}$$

with corrections[49] to account for a particle Mach number greater than 0.4 at a particle Reynolds number greater than 20.

Equation 89, particle force balance equation permits incorporation of additional forces (F_x) that can be important under special circumstances. The first of these is the

"virtual mass" force, the force required to accelerate the fluid surrounding the particle. This force can be written as

$$F_x = \frac{1}{2}\frac{\rho}{\rho_p}\frac{d}{dt}\left(u - u_p\right) \tag{102}$$

Virtual mass force is important when the fluid density exceeds particle density ($\rho > \rho_p$). The virtual mass force is unimportant in this analysis but is included for completeness. Forces arising due to the pressure gradient in the fluid can be included via Equation 103.

$$F_x = \left(\frac{\rho}{\rho_p}\right) u_p \frac{\partial u}{\partial x} \tag{103}$$

For sub-micron particles, the effects of Brownian motion can optionally be included in the additional force term. The components of the Brownian force are modeled as a Gaussian white noise process with spectral intensity,[50] $S_{n,ij}$ given by

$$S_{n,ij} = S_0 \delta_{ij} \tag{104}$$

where δ_{ij} is the Kronecker delta function, and

$$S_0 = \frac{216 \nu k_B T}{\pi^2 \rho d_p^5 \left(\frac{\rho_p}{\rho}\right)^2 C_c} \tag{105}$$

T is the absolute temperature of the fluid, ν is the kinematic viscosity, and k_B is Boltzmann's constant. Amplitudes of the Brownian force components are of the form

$$F_{b_i} = \zeta_i \sqrt{\frac{\pi S_0}{\Delta t}} \tag{106}$$

where ζ_i are zero-mean, unit-variance-independent Gaussian random numbers. The amplitudes of the Brownian force components are evaluated at each time step. The

energy equation must be enabled in order for the Brownian force to take effect. Brownian force is intended only for non-turbulent models and its description was included here again for completeness.

Saffman's lift force, or lift due to shear, can also be included in the additional force term as an option. The lift force used is from Li and Ahmadi[50] and is a generalization of the expression provided by Saffman.[51]

$$\vec{F} = \frac{2K\rho d_{ij}\sqrt{\nu}}{\rho_p d_p \left(d_{lk} d_{kl}\right)^{\frac{1}{4}}}\left(\vec{v} - \vec{v}_p\right) \qquad \textbf{(107)}$$

where $K = 2.594$ and d_{ij} is the deformation tensor. The deformation tensor[52] is defined as $d_{ij} = 0.5\left(s_{ij} + s_{ji}\right)$ where s_{ij} is the mean shear rate in the j-direction for the i-component of the mean velocity of the carrier phase $\left(s_{ij} \equiv \partial\langle u_i \rangle / \partial x_i\right)$. The criterion that must be satisfied for this form of the lift force to be valid is that the particle Reynolds number based on the particle-fluid velocity difference must be smaller than the square root of the particle Reynolds number based on the shear field. This restriction is valid for submicron particles. Therefore, FLUENT recommends its use only for submicron particles.

Integration of the Trajectory Equations

The trajectory equations, and any auxiliary equations describing heat or mass transfer to or from the particle, are solved by stepwise integration over discrete time steps. Integration in time of the particle force-balance equation yields the velocity of the particle (u_p) at each point along the trajectory, with the trajectory itself predicted by

$$\frac{dx}{dt} = u_p \tag{108}$$

Equations similar to the particle force-balance equation and (*dx*/*dt*) are solved in each coordinate direction to predict the trajectories of the discrete phase. Assuming the term containing the body force remains constant over each small time interval, and linearizing any other forces acting on the particle, the trajectory equation can be rewritten in simplified form as

$$\frac{du_p}{dt} = \frac{1}{\tau_p}\left(u - u_p\right) \tag{109}$$

where τ_p is the particle relaxation time. FLUENT uses a trapezoidal scheme for integrating the trajectory equation.

$$\frac{u_p^{n+1} - u_p^n}{\Delta t} = \frac{1}{\tau}\left(u^* - u^{n+1}\right) \tag{110}$$

where *n* represents the iteration number and

$$u^* = \frac{1}{2}\left(u^n + u^{n+1}\right) \tag{111}$$

$$u^{n+1} = u^n + \Delta t u_p^n \cdot \nabla u^n \tag{112}$$

$\frac{dx}{dt} = u_p$, and $\frac{du_p}{dt} = \frac{1}{\tau_p}\left(u - u_p\right)$ are solved simultaneously to determine the velocity and position of the particle at any given time.

Default Discrete Random Walk Model (DRW)

The simulation of interactions of a succession of discrete stylized fluid phase turbulent eddies is modeled in the default Discrete Random Walk (DRW) model. A

Gaussian distributed random velocity fluctuation (u', v', and w') and a time scale (τ_e) characterizes each eddy. The values of the Gaussian distributed random velocity fluctuations that prevail during the lifetime of the turbulent eddy are sampled by assuming that they obey a Gaussian probability distribution, so that $u' = \zeta\sqrt{\overline{u'^2}}$. Zeta ($\zeta$) represents a normally distributed random number. $\sqrt{\overline{u'^2}}$ represents the local root-mean-square (RMS) of the velocity fluctuation. Values of the RMS fluctuating components are obtained from employing the isotropy assumption and making use of the known turbulence kinetic energy. The turbulence quantities are known because solution of the continuous phase is necessary (the Eulerian solution) prior to the use of the Lagrangian particle trajectory calculation. Values valid for the k-ε, k-ω, and their variants are $\sqrt{\overline{u'^2}} = \sqrt{\overline{v'^2}} = \sqrt{\overline{w'^2}} = \sqrt{2k/3}$. For the Reynolds stress model (RSM) derivation of the velocity fluctuations anisotropy of the stresses is included. The value of the RMS fluctuating components in this case are $u' = \zeta\sqrt{\overline{u'^2}}$, $v' = \zeta\sqrt{\overline{v'^2}}$, and $w' = \zeta\sqrt{\overline{w'^2}}$.

The characteristic lifetime of the eddy (τ_e) is either a constant or a random variation about T_L (fluid Lagrangian integral time). The uniform random number, r, is defined on the semi-open interval [0, 1). Shown below for the two-equation turbulence models are the constant and random formulation for the characteristic eddy lifetime (time spent in turbulent motion along the particle path).

k-ε model and variants: $\tau_e = 2T_L = 2C_L \frac{k}{\varepsilon}$ or

$$\tau_e = -T_L \log(r) = -C_L \frac{k}{\varepsilon} \log(r)$$

k-ω model and variants: $\tau_e = 2T_L = 2C_L \omega^{-1}$ or $\tau_e = -T_L \log(r) = -\frac{C_L}{\omega} \log(r)$

The particle eddy crossing time, τ_{cross} (Equation 113) is a function of particle relaxation time $\left(\tau = \rho_p d_p^2 / 18 \rho_g \nu_g\right)$, eddy length scale ($L_e$), and the magnitude of the relative velocity $\left(\left|u_g - u_p\right|\right)$.

$$\tau_{cross} = -\tau \ln\left[1 - \left(\frac{L_e}{\tau \left|u_g - u_p\right|}\right)\right] \quad \textbf{(113)}$$

The particle interacts with the fluid phase eddy over the interaction time (i.e., the smaller of the eddy lifetime and the eddy crossing time). When the eddy lifetime is reached, a new value of the instantaneous velocity is obtained by applying a new value of ζ in $u' = \zeta \sqrt{\overline{u'^2}}$. During one interaction time, the particle velocity is kept constant and its trajectory is calculated. In the next interaction time, the particle is at the new location, and the local fluid-phase turbulent fluctuations are used to calculate the current particle velocity and trajectory.

Modification to the Dispersed Phase Model (DPM)

The user defined function (UDF) developed for use in FLUENT to modify the stochastic DPM model is described herein. The UDF is intended for use in post-

processing particle tracking on steady state flow fields computed from either *k-ε* and its variants, k-ω and its variants, or RSM turbulence model.

Lagrangian Time Scale & RMS Fluctuating Velocities

Initially, the default stochastic model is employed as long as the particle y^+ is greater than 200. Where, y^+ is defined as:

$$y^+ = \frac{y \cdot u^*}{\upsilon} \tag{114}$$

In this equation, y is the distance of the particle to the nearest wall, ν is the kinematic viscosity and u^* is the friction velocity. The definition for friction velocity shown below is the square root of the ratio of wall shear stress to fluid density.

$$u^* = \sqrt{\frac{\tau_w}{\rho}} \tag{115}$$

If y^+ is less than 200, then the Lagrangian time scale (T_L^+) is changed to a wall function.[53]

$$T_L^+ = \begin{cases} 7.122 + 0.5731 \cdot y^+ - 0.00129 \cdot \left(y^+\right)^2 & 5.0 < y^+ < 200 \\ 10.0 & y^+ \le 5.0 \end{cases} \tag{116}$$

$$T_L^+ = T_L \cdot \frac{(u^*)^2}{\nu} \tag{117}$$

The values of the RMS fluctuating velocities are modified to account for the strong anisotropic nature of the flow in the boundary layer. The correction factors come from curve fits of DNS results.[54]

streamwise direction $\vec{I}$: $$\sqrt{\overline{U'^2}} = \frac{0.40 \cdot y^+}{1 + 0.0239(y^+)^{1.496}} \cdot u^* \quad \textbf{(118)}$$

normal direction to wall $\vec{J}$: $$\sqrt{\overline{V'^2}} = \frac{0.0116 \cdot (y^+)^2}{1 + 0.203 \cdot y^+ + 0.00140(y^+)^{2.421}} \cdot u^* \quad \textbf{(119)}$$

spanwise direction $\vec{K} = \vec{I} \times \vec{J}$: $$\sqrt{\overline{W'^2}} = \frac{0.19 \cdot y^+}{1 + 0.0361(y^+)^{1.322}} \cdot u^* \quad \textbf{(120)}$$

The streamwise, normal, and spanwise unit vectors in the particle local coordinate system ($\vec{I}$, $\vec{J}$, $\vec{K}$) are determined as follows:

- One obtains the mean fluid velocity vector $\vec{u}$ at a distance $y^+ = 200$ from the wall. The unit vector $\vec{I}$ in the streamwise direction is equal to $\frac{\vec{u}}{\|\vec{u}\|}$
- $\vec{J}$ is the unit vector normal to the face of the nearest wall
- The spanwise unit vector is just $\vec{K} = \vec{I} \times \vec{J}$

The RMS components of velocity in the local coordinate system $\left(\vec{I}, \vec{J}, \vec{K}\right)$ are as follows:

- The streamwise RMS velocity $\sqrt{\overline{U'^2}}$ ($\vec{I}$ component)
- The normal RMS velocity $\sqrt{\overline{V'^2}}$ ($\vec{J}$ component)
- The spanwise RMS velocity $\sqrt{\overline{W'^2}}$ ($\vec{K}$ component)

The fluctuating velocity in the local coordinate system $\left(\vec{I},\vec{J},\vec{K}\right)$during the eddy lifetime is computed by randomizing the components:

$$U' = \psi_1 \sqrt{\overline{U'^2}} \tag{121}$$

$$V' = \psi_2 \sqrt{\overline{V'^2}} \tag{122}$$

$$W' = \psi_3 \sqrt{\overline{W'^2}} \tag{123}$$

The ψ's are Gaussian random numbers with 0 mean and standard deviation 1. Finally, the fluctuating velocity vector $\vec{U}'$ is transformed back into the computational coordinate frame x-y-z ($\vec{i}$, $\vec{j}$, $\vec{k}$) to become $\begin{pmatrix} u' \\ v' \\ w' \end{pmatrix}$ by writing:

$$\vec{U}' = U'\vec{I} + V'\vec{J} + W'\vec{K} = \vec{u}' = u'\vec{i} + v'\vec{j} + w'\vec{k} \tag{124}$$

The $\begin{pmatrix} u' \\ v' \\ w' \end{pmatrix}$ now replace the non-isotropy of the stresses included in the derivation of the velocity fluctuations for the lifetime of the eddy as given in Equations 125 through 127 below.

$$u' = \zeta \sqrt{\overline{u'^2}} \tag{125}$$

$$v' = \zeta \sqrt{\overline{v'^2}} \tag{126}$$

$$w' = \zeta \sqrt{\overline{w'^2}} \tag{127}$$

Spurious Drift

Random walk models, whether discrete (DRW) or continuous (CRW), are known to introduce spurious drift.[55, 56, 57] This drift artificially pushes small (e.g., tracer) particles from high turbulence regions (e.g. bulk) into low turbulence regions (e.g. walls) within a single eddy lifetime. Thus, a particle receives a large fluctuation velocity at the beginning of an eddy may cause the particle to be directed toward the wall. This particle then potentially continues to move outward toward the boundary layer edge under the effect of the fluctuation until the eddy lifetime ends. For the particle to move beyond the boundary layer edge, the sampled transverse fluctuation velocity must exceed the magnitude of the mean transverse fluid velocity. This action tends to move the particle back into the boundary layer. When the particle deposits out in the free stream and a new eddy begins, the particle moves back toward the boundary layer at the mean fluid velocity. Thus, particles may be ejected by turbulent motion from the boundary layer, but cannot be assisted back into the boundary layer by turbulence[55]. This behavior contributes to the observed over-prediction of deposition of small particles if walls are present.[55, 56, 57] Small particles should remain fully mixed in the limit of small fluid particles, and not deposit on surfaces.

Attempts have been made to correct for this drift with some success, in particular the recent model of Bocksell and Loth.[57] These authors proposed modifications that essentially remove the spurious drift if CRW is used, and reduce the drift significantly in the case of DRW.

For the DRW, Bocksell and Loth[57] propose that during the particle trajectory integration inside an eddy life, the turbulent fluid velocity be corrected as follows (vector form):

$$U'_{new} = \psi\sqrt{\overline{U'^2}} + \overline{U}' \tag{128}$$

where ψ is a random number with zero mean and unit standard deviation (held constant during an eddy lifetime). The first term on the right hand side is held constant during the eddy lifetime. However, the correction velocity term, second term of the RHS, is updated at each time step during the eddy in the following manner.

$$\overline{U}'(t+\Delta t) = \overline{U}'(t) + \Delta t \cdot \overline{U'}\frac{\partial \overline{U'}}{\partial x_j} \tag{129}$$

There are two possibilities on how to proceed depending on the particle location: isotropic turbulence (particle in the bulk) and anisotropic turbulence (particle in the boundary layer).

For particles in the bulk, the correction for any velocity component (e.g., *V*) is of the form:

$$V'_{new} = \psi_2\sqrt{\overline{V'^2}} + \overline{V}' \tag{130}$$

$$\overline{V}'(t+\Delta t) = \overline{V}'(t) + \Delta t\frac{1}{3}\frac{\partial k}{\partial y} \tag{131}$$

In the context of the default DRW model in FLUENT, correction implementation is straightforward. The only requirement is addition of a correction for the fluid

turbulent velocity, $\overline{V}'(t+\Delta t)$, at each time step inside the eddy. This involves the computation of the gradient of *k*. However, the gradient of *k* is already available through the macro for cell gradients C_D_G(c, t), the turbulent kinetic energy dissipation rate gradient vector.

For anisotropic turbulence, in the context of the improved treatment inside boundary layers (our UDF), we already keep track of the particle distance to the nearest wall (local coordinates) and the correction would be, for example for the wall normal velocity:

$$V'_{new} = \psi_2 \sqrt{\overline{V'^2}} + \overline{V}' \tag{132}$$

with

$$\overline{V}'(t+\Delta t) = \overline{V}'(t) + \Delta t \cdot \overline{V}'(t) \frac{\partial \overline{V'(t)}}{\partial y} \tag{133}$$

where y is the local normal coordinate axis.

$$\overline{V}'(t) = \sqrt{\overline{V'^2}} = \frac{0.0116 \cdot (y^+)^2}{1 + 0.203 \cdot y^+ + 0.00140 (y^+)^{2.421}} \cdot u^* \tag{134}$$

The derivative in the second term is just:

$$\frac{\partial \overline{V}'}{\partial y} = \frac{\partial \overline{V}'}{\partial y^+} \frac{\partial y^+}{\partial y} = \frac{\partial \overline{V}'}{\partial y^+} \frac{u^*}{\upsilon} \tag{135}$$

The term $\frac{\partial \overline{V}'}{\partial y^+}$ could be calculated analytically or numerically. The same procedure applies to other components.

As a first attempt to see the approximate effect of this correction on the UDF results, one could define a mean correction during the eddy[56] and correct just once during the whole lifetime of the eddy τ_e. Equation (132) would become:

$$V'_{new} = \lambda_2 \sqrt{\overline{V'^2}} + \frac{\tau_e}{2} \cdot \overline{V}'(t) \frac{\partial \overline{V'}}{\partial y} \qquad \textbf{(136)}$$

The same type of formulation can be extended to the other components of velocity.

CHAPTER III

SOLUTION ALGORITHM

Outlined in this chapter are the steps taken to setup and solve the problem (i.e., wall-bounded turbulent flow of aerosol-laden compressible gas) for the steady-state flow field solution and particle trajectories. The initial step included preprocessing, which consisted of:

1) creating the solid geometry,
2) creating the mesh (geometry divided into smaller elements) with the attached boundary layer (elements adjacent to the solid wall),
3) definition of boundaries (e.g., inlet, outlet, and walls),
4) specification of the fluid region, and
5) exporting the mesh for use in the solver.

Next the mesh was imported into the solver (i.e., FLUENT) where its internal algorithms created the computational domain (i.e., control volumes). The fluid flow field was computed to obtain a converged solution (Eulerian) then particle trajectories were computed (Lagrangian approach). Each of the aforementioned topics are addressed in the following paragraphs.

Computational Domain

The solid geometry used in this study was created from primitives provided in GAMBIT (Geometry and Mesh Building Intelligent Toolkit), FLUENT's preprocessing software package. For use with the realizable k-ε turbulence model, the entire U-tube

configuration (18.948 m overall length; 16.87 mm ID; and 82.3 mm U-bend radius of curvature) was created and meshed. However, the computational domain (i.e., mesh) generated for use with the standard k-ω turbulence model consisted of straight sections of 200 diameters in length and the bend. The "tmerge" utility in the FLUENT package allowed for assemblage of both straight sections connecting the bend.

The reason for the different model setup is that the two turbulence models require different resolution in the region adjacent to the wall in order to obtain sufficient information to characterize gradients. A relatively coarse mesh near the wall is required for the realizable k-ε model while the standard k-ω model requires a very fine mesh. To control the mesh density and, thereby, to control the amount of information available in the computational model in specific regions of interest, the boundary layer feature in GAMBIT was used. The spacing of mesh node rows in regions immediately adjacent to edges and/or faces requires the user to define the boundary layer. Thus, the required user-specified information includes specification of the boundary layer algorithm; height of the first row of mesh elements; growth factor; and the total number of rows. The growth factor specifies the height of each succeeding row of elements. The total number of rows defines the depth of the boundary layer. Additional user provided information include specification of which edge or face to attach the boundary layer and which face or volume defines the direction of the boundary layer.

Since the Reynolds number is less than 5×10^5, the boundary layer is laminar and the velocity distribution in the boundary layer is parabolic.[58] With this as the basis, equation 137 (pg 40, Ref. 33), gives an estimate of $0.03l$ for the boundary layer

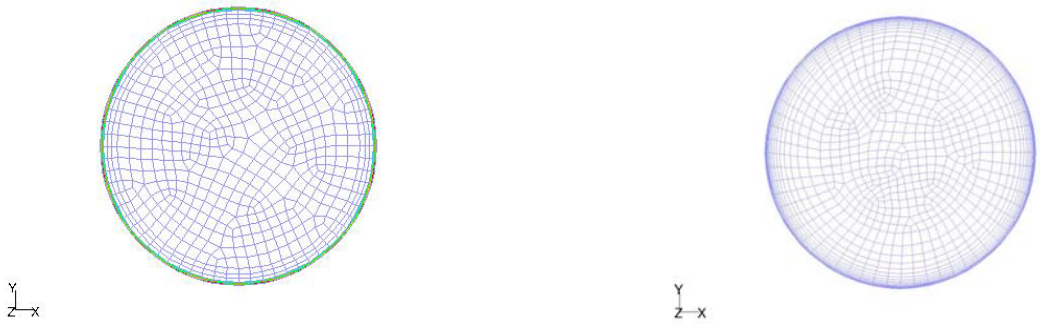
thickness. Table 6 shows an alternate approach to determining the location of the first cell centroid.

For the realizable k-ε model, using a 1.2 growth factor, four rows of cells starting at 0.1886 mm gave a depth of 1.0124 mm. For the standard k-ω model, using a 1.3 growth factor, 25 rows of cells starting at 1.47707 μm gave a depth of 3.46935 mm. Figure 16 shows the sixty-eight (68) node placement on the edge of the inlet face.

$$\frac{\delta}{l} = 0.37\left(\frac{U_\infty l}{\nu}\right)^{-1/5} = 0.37\left(\mathrm{Re}_l\right)^{-1/5} \qquad \textbf{(137)}$$

Table 6. Establishing Placement of First Grid Point

- Ability for near wall treatments to accurately predict near wall flows depends on placement of wall adjacent cell centroids (cell size).
  - For standard wall functions and non-equilibrium wall functions, centroid should be located in log-layer: $y_p^+ \approx$ 30-300
  - For best results using enhanced wall treatment, centroid should be located in laminar sublayer: $y_p^+ \approx 1$
    - This near-wall treatment can accommodate cells placed in the log-layer.
- To determine actual size of wall adjacent cells, recall that:
  - $y_p^+ \equiv y_p u_\tau / \nu \Rightarrow \boxed{y_p \equiv y_p^+ \nu / u_\tau}$
  - $u_\tau \equiv \sqrt{\tau_w / \rho} = U_0 \sqrt{\overline{c_f}/2}$
  - Skin friction coefficient can be estimated from empirical correlation;[59]
    - Pipe flow: $\overline{c_f}/2 \approx 0.0359\,\mathrm{Re}_D^{-0.2}$
- Use post-processing to confirm near-wall mesh resolution.



(a) Realizable k-ε (b) Standard k-ω

Figure 16. Turbulence Model: Grid

Mesh topologies in Figure 16 illustrates that for the given cross section, two zones are present. A structured mesh applied in the circumferential part, where wall effects require an accurate representation. An unstructured mesh employed in the bulk region where velocity gradients are less important. The discretization in the circumferential direction is uniform, but the radial distribution was chosen in order to increase the number of elementary volumes close to walls. With the boundary layer in place and face mesh attached, all geometries were meshed (three-dimensionally) with elementary hexahedral volumes. The Cooper-type element, a volume meshing type in GAMBIT, used an algorithm to sweep the mesh node patterns of specified 'source' faces through the volume. The hexagonal shape and Cooper-type elements were employed to obtain better convergence and accuracy.

The realizable k-ε model has 1.5 x 10^6 hexahedral cells. For the k-ω model, each straight section contains 8.2x10^5 (814,464) hexahedral cells while the bend has 7.3x10^5 (732,800) hexahedral cells for a total of ~2.4 x 10^6 cells. These mesh configurations were exported for use in the solver, FLUENT.

Turbulence Models

The first of the two turbulence models, the realizable k-ε option was selected from the viscous model panel. Non-equilibrium wall functions were enabled for near wall treatment. For the turbulence specification method, (boundary conditions panel) values for intensity and hydraulic diameter were input. With intensity and hydraulic diameter set as the turbulence specification method FLUENT computes turbulent kinetic energy and turbulent dissipation rate. Table 7 lists the estimation of turbulent kinetic energy and turbulent dissipation rate manually input in the initialization panel. In the table, turbulence length scale, *l*, is a physical quantity related to the size of the largest "energy containing" eddies in turbulent flows. The size of the pipe restricts the value of the turbulence length scale in fully developed pipe flow since the turbulent eddies cannot be larger than the pipe. So, in the expression for the turbulence length scale, $l = 0.07L$, *L* is the relevant dimension of the pipe (i.e., diameter). The factor, 0.07, is based on the maximum value of the mixing length in fully developed turbulent pipe flow. In the FLUENT documentation it is recommended that the user create using the custom field function calculator panel the quantities, the ratio of production of kinetic energy (k) to its dissipation $\left(G_k/\rho\varepsilon\right)$; ratio of the mean flow to turbulent time scale $\xi \equiv \left(Sk/\varepsilon\right)$; and Reynolds stresses derived from the Boussinesq formula $\left(e.g.,-\overline{uv} = \nu_t\, \partial u/\partial y\right)$.

Table 7. Boundary Conditions for the Realizable k-ε Model

	Formula	Inlet	Outlet
Turbulence intensity (%)	$I = 0.16\left(\mathrm{Re}_{D_H}\right)^{-\frac{1}{8}}$	3.344	3.328
Turbulent kinetic energy (m^2/s^2)	$k = 1.5\left(u_{avg} I\right)^2$	8.456	97.27
Turbulent dissipation rate (m^2/s^3)	$\varepsilon = C_\mu^{\frac{3}{4}} k^{\frac{3}{2}} / l$	3421.6	133492.3

The two-equation, standard k-ω model was the second turbulence model used in this analysis. In its setup, FLUENT sets the near-wall treatment automatically. Enabling the transitional flow option in the viscous model panel permits usage of the low-Reynolds number variants for correction to the turbulent viscosity. Thus, the damping coefficient (default setting = 1) damps the turbulent viscosity permitting a low-Reynolds-number correction. Enabling this option requires adherence to mesh guidelines for the enhanced wall treatment (i.e., $y^+ \approx 1$). Inclusion of the viscous dissipation terms in the energy equation, which describe the thermal energy created by viscous shear in the flow, requires enabling viscous heating in the viscous model panel. Table 8 lists the boundary conditions for the turbulent kinetic energy (k) and specific dissipation rate (ω), the latter estimated from the turbulence length scale ($l = 0.07L$).

Table 8. Boundary Conditions for k-ω Model

	Formula	Inlet	Outlet
Turbulence intensity (%)	$I = 0.16\left(\mathrm{Re}_{D_H}\right)^{-\frac{1}{8}}$	3.344	3.328
Turbulent kinetic energy (m^2/s^2)	$k = 1.5\left(u_{avg} I\right)^2$	8.456	97.27
Specific dissipation rate (1/s)	$\omega = \sqrt{k} / \left(C_\mu^{\frac{1}{4}} l\right)$	4495.9	15248.3

Compressible Flow Setup

The important parameters in the setup of a compressible flow calculation are operating pressure (p_{op}), ideal gas model for computing density variations (activates the energy equation), viscous dissipation terms in the energy equation (requires enabling viscous heating in solver panel), and boundary conditions.

The recommended setting for operating pressure depends on the density relationship and the Mach number (> 0.1). To avoid numerical round-off effects on pressure calculations in low-Mach number flow, operating pressure is set to either zero or mean flow pressure. Operating pressure set to zero sets absolute pressure (p_{abs}) equivalent to gauge pressure (p_{gauge}) by definition (i.e., in FLUENT $p_{abs} = p_{op} + p_{gauge}$). Pressure (total or static) is set relative to the operating pressure (i.e., gauge pressure). Zero operating pressure, was used in the flow problem.

Enabling the energy equation allows coupling between the flow velocity and the static temperature. Viscous dissipation terms in the energy equation are not automatically computed when the segregated solver is used. However, by enabling viscous heating in the solver panel these terms are computed.

For inlet flow, either pressure inlet or mass flow inlet is available. Total (stagnation) temperature is required input with either boundary condition selection. Total pressure (and static pressure if flow is supersonic) is required input for pressure inlets. The mass flow inlet boundary condition was selected in the current simulation. The required input was inlet mass flow and total (stagnation) temperature. The only boundary condition used in regions where flow exits the system is pressure outlet and its

required input is static pressure. In post-processing compressible flow, variables of interest are total temperature, total pressure, and Mach number. Monitors were set to track the behavior of these parameters throughout the calculation.

Numeric Scheme

Execution of the solver required as an initial step, solution algorithm selection. The function of the solver is to provide a vehicle for solution of the continuity, momentum, and energy equations. The options available in FLUENT for solution of the governing equations are segregated or coupled solver. The segregated solver solves the governing equations sequentially whereas the coupled solver performs simultaneous solution of the governing equations. In both the segregated and coupled solution methods, the discrete, non-linear governing equations are linearized to produce a system of equations for the dependent variables ("unknowns", e.g., velocities, pressure, temperature, and conserved scalars) in every computational cell. An updated flow-field solution is obtained when the resultant linear system is solved. Depending on the solution algorithm, either an implicit or explicit linearization technique is implemented. Implicit linearization involves computation of the unknown value in each cell by using a relation that includes both existing and unknown values from neighboring cells. Whereas in explicit linearization, only existing values are used to compute the unknown variable in each cell. The implicit option is available in both solution algorithms but there is no explicit option in the segregated solver. The two numerical methods use different linearization approaches but discretization processes are similar.

Discretization schemes are control-volume based techniques whereby integration of the governing equations about each control volume yields discrete equations that conserve each quantity (e.g., mass, momentum, energy). The discrete values of scalar quantities are stored at the cell center. Convective terms, required at faces, are obtained by interpolation of cell center values upstream (i.e., upwind) of the face, relative to the direction of normal velocity. Face values for diffusive terms are central differenced. The selection of discretization scheme depends on the physics models (e.g., compressible flow, turbulence models) used in the simulation.

Since the discretized equations along with the initial and boundary conditions were solved to obtain a numerical solution using the segregated solution method, only discretization schemes for the implicit segregated solver will be presented in what follows. Discretization requires scheme selection for momentum (pressure interpolation), continuity (density interpolation), pressure-velocity coupling, and time-advancement.

In general, the available upwind schemes for converting the governing equations to algebraic equations (i.e., discretization) include first-order upwind, second-order upwind, power-law, QUICK (quadratic upwind), and central-differencing schemes. The available pressure interpolation schemes include the body-force-weighted scheme, "PRESTO!" (PREssure Staggering Option), and second-order scheme. The SIMPLE (Semi-Implicit Method for Pressure-Linked Equations) family of algorithms are available for the discretization of the continuity equation. For compressible flow calculations FLUENT applies upwind interpolation of density at cell faces and the

available schemes include first-order upwind, second-order upwind, and QUICK. For steady-state calculations, the available pressure-velocity coupling algorithms are SIMPLE, SIMPLEC, and PISO (Pressure Implicit with Split Operator). The default, iterative time-advancement scheme advanced the solutions for a given time step until the convergence criteria was met.

The recommendation offered in the FLUENT documentation is "PRESTO!" for flows in strongly curved domains and second-order pressure interpolation scheme for compressible flows. Second-order pressure interpolation was used for the realizable k-ε calculations, while "PRESTO!" was used with the standard k-ω turbulence model. For pressure-velocity coupling, the phase-coupled SIMPLE algorithm[60] was used. A second-order upwind discretization scheme was used for the turbulent kinetic and turbulent dissipation energy. These schemes ensured, in general, satisfactory accuracy, stability, and convergence.

For this single-phase flow calculation, the steady-state solution strategy was employed. The convergence criterion was based on the residual value of the calculated variables, i.e. mass, velocity components, turbulent kinetic energies, and turbulent dissipation energies. The default threshold values (0.001) were accepted initially because FLUENT considers these values small enough to produce accurate results. Subsequent to solution convergence with the default, thresholds were reduced an order of magnitude and the calculation was again run to convergence. This action was repeated until solution convergence for 10^{-7} was realized. This procedure ensured convergence independence in that the solution did not change between 10^{-5} and 10^{-7}.

Other solution strategies include reduction of under relaxation factors of momentum, turbulence kinetic energy, and turbulence dissipation energy to bring the non-linear equation close to the linear equation. The best strategy was to initialize the solution with accurate initial guesses for user-specified parameters.

Continuous Phase (Flow Field)

At the start of the calculation fluid properties are updated based on the initialized solution. In subsequent calculations, fluid properties are updated based on the current solution. In order to update the velocity field, pressure and face mass fluxes are used to solve the components of the momentum equations. These velocities must satisfy the continuity equation locally. If continuity is not met, a Poisson type equation for the pressure correction is derived from the linearized momentum equations and the continuity equation. This pressure correction equation is then solved to obtain the necessary corrections to the pressure and velocity fields and the face mass fluxes such that continuity is satisfied. Next, equations for scalars (e.g., turbulence and energy) are solved using the previously updated values of the other variables. A check for convergence of the equation set is made. These steps are continued until the convergence criteria are met.

Under relaxation controls the update of computed variables (i.e., pressure, density, body forces, momentum, turbulence kinetic energy, turbulence dissipation rate, turbulent viscosity, and energy) at each iteration. Default under relaxation values should be used unless other models require different input. To accelerate convergence, parameters for the algebraic multigrid (AMG) solver could be modified. Default values

were used. Therefore, cycle type for pressure equation (V-cycle), and flexible for x-, y-, z-momentum, turbulence kinetic energy, turbulence dissipation rate, and energy are retained along with the termination and restriction values. Default values for flexible cycle parameters (i.e., sweeps = 2; maximum fine relaxations = 30); and maximum coarse relaxations (= 50) were retained. Aggregative AMG (AAMG), the default solver, was used with no specified stabilization method.

The approach for determining solution convergence using the default convergence criterion was said to be appropriate for most situations. With the default criterion, scaled residuals decrease to 10^{-3} for all equation except energy (10^{-6}). In the present study residuals decreased to 10^{-7} for all parameters. The alternate approach advanced in the FLUENT User's Guide involves requiring the normalized un-scaled residuals to drop by three orders of magnitude. However, this criterion alone may not be enough, so integrated quantities (e.g., drag) should be monitored and used to determine solution convergence.

Discrete Phase Model (DPM)

The low volume fraction of the second phase allowed use of the Lagrangian DPM in the quantification of aerosol retention in the tube. Based on interparticle spacing and low particulate loading, coupling between the phases is one-way. In this analysis, inert particles were injected into the continuous phase from a surface. The surface itself would generate a non-uniform distribution of points. Therefore, in order to generate a uniform spatial distribution of particle streams released from the surface, a bounded plane surface with equally spaced data points was created. Using the Plane

Surface panel, bounded plane with sample points was selected to create several injection planes. For example, the coordinates selected for locating the plane at the inlet were: (x_0 = -.008435, y_0 = -.008435, z_0 = 0); (x_1 = -0.008435, y_1 = 0.008435, z_1 = 0); and (x_2 = 0.008435, y_2 = 0.008435, z_2 = 0). Fourteen (14) points on edge 1 connected the origin to the first point (i.e., x_0, y_0, z_0) and twenty-eight (28) points on edge 2 connected the first point to the second point (i.e., x_1, y_1, z_1). The plane contained 336 points (336 0D facets).

Table 9 lists input for the dispersed phase model used for both turbulence models. The execution of the user-defined function (UDF) modifying the stochastic model require different input. Its implementation requires compiling the UDF and building the associated libraries. Drag law, scalar update, DPM time step, the number of scalars allotted for information storage for each particle, along with modification of the wall unit threshold are but a few of the several changes. Another difference is that for numerics, accuracy control is disabled and trapezoidal tracking scheme is used.

Table 9. Dispersed Phase Model (DPM) Input

Step length scale	Default
Maximum number of steps	7.59E+06
Drag law	Spherical
Saffman lift force	Enabled
Accuracy control	Enabled
Tolerance	Default
Maximum refinements	Default
Automated tracking scheme	Enabled
Higher order scheme	Runge-Kutta
Lower order scheme	Implicit
Injection name	User-defined
Injection type	Surface
Diameter distribution	Uniform
Released from	Bound plane (equally spaced nodes)
Particle type	Inert
Particle material type	Titanium
Velocity of particle (x)	Continuous flow field result
Velocity of particle (y)	Continuous flow field result
Velocity of particle (z)	Continuous flow field result
Temperature (K)	Continuous flow field result
Mass flow rate (kg/s)	0.065
Turbulent dispersion	Stochastic tracking model
Number of tries	2
Effect of turbulent velocity fluctuations on particle trajectories	DRW: Discrete random walk model
Random eddy lifetime	Enabled
Integral time scale constant	0.5
Boundary conditions, inlet/outlet	Escape
Boundary conditions, wall	Trap
Boundary conditions, wall (inlet)	Reflect; $e_n = e_t = 0$ (inelastic collision)

CHAPTER IV

RESULTS AND DISCUSSION

Experimental data from Phase I of the ARTIST project provided the bounding conditions used for simulation setup. The chapter begins with a detailed discussion of the ARTIST (Phase I) experimental conditions, which permits transition into the discussion of flow field characteristics. The latter lays bear the rational for physical model selection from the suite of models available in the CFD code (FLUENT). Presented next are the numerical results of flow field calculations. Parameters used in the flow field evaluation include velocity, vorticity, helicity, and turbulence intensity. A brief explanation of vorticity, helicity, and turbulence intensity precedes presentation of the results. Particle deposition behavior is examined. Presented herein is a comparison of correlations from the open literature against the computationally derived particle deposition results. In addition, the numerical experiment is compared against the physical experiment. Lastly, the decontamination factor (DF) is computed which is indicative of the amount of aerosol retention in the U-tube configuration.

ARTIST Experiment: Phase I, Single U-tube

Table 10 lists the mean values of the thermal hydraulic variables, aerosol size distribution, and decontamination factor (DF) from the ARTIST Phase I experiment. The overall length of the U-shaped tube under consideration was 18.948 m with an internal diameter of 0.01687 m. The curvature ratio (δ, defined as the ratio of bend curvature (R) to internal tube radius) is 9.76. Velocities in the experiment varied from

71 m/s at the inlet to 242 m/s at the outlet. There was nearly a fourfold difference between inlet and outlet pressure (0.36 and 0.1067 MPa, respectively).

The working fluid used in the experiment conducted at near room temperature was nitrogen. The total volume of gas was approximately 0.04586 m^3. The ratio of the mass flow rate to volumetric flow rate of nitrogen gives an indication of the density at the boundaries (inlet 4.1 kg/m^3 and outlet 1.2 kg/m^3).

Fluidized bed generators fed a total of 391.5 grams of TiO_2 (ρ_p= 3.95 g/cc), of which approximately 199 grams collected in the primary system, mostly in the mixing chamber. 192.2 grams reached the ARTIST test section, with 62.75 grams collected on test tube walls. The volume of aerosol ($192.2/\rho_p$) is approximately 48.7 cm^3 (just under four tablespoons). The aerosol concentrations in the inlet and outlet lines of the ARTIST facility were 0.456 g/m^3 and 0.095 g/m^3, respectively. Volume fraction of aerosol (0.11%) and gas phase (99.89%) is indicative of a dilute flow. The material density ratio ($\varphi \equiv \rho_p/\rho_c$), is nearly 1000. The particulate loading ($\beta \equiv (\alpha_p\rho_p)/(\alpha_c\rho_c)$), ratio of the product of volume fraction and density for the dispersed and continuous phase, $\beta \approx$ 1.026. The mechanics of a dispersed phase flow also depends on the average distance between the particles. By estimating this distance, it can be determined whether the particles are in contact or isolated. The measure of interparticle spacing expressed as L/D [$= (\pi/6)(1+\kappa)/\kappa)^{1/3}$], using κ ($= \beta/\varphi$), the average distance between individual particles of the dispersed phase is 7.91. Thus, individual particles can be treated as isolated and neighboring particles have little influence on each other.[61]

The aerosol size distribution at the inlet was broad with an inlet aerodynamic mass median diameter (AMMD) between 4.8 and 5.5 μm and a geometric standard deviation (GSD) between 2.3 and 3.4. At the outlet, the AMMD was 1.05 μm, with a GSD between 2.6 and 2.7. A mean DF due to aerosol retention in the U-tube of 1.4 was reported for the 5.5-hour test.

Table 10. Mean Values for Single U-Tube Test in ARTIST Experiment

Variable	Unit	Inlet	Outlet
Mass flow rate	kg/h	234	234
Volumetric flow rate	m^3/h	57	195
Pressure	bar_a	3.6	1.067
Gas temperature	°C	24.1	22.9
Aerosol concentration	g/m^3	0.456	0.095
Aerosol AMMD	μm	4.8 – 5.5	1.0 - 1.1
Aerosol GSD	-	2.3 – 3.4	2.6 - 2.7
DF	-	0.9 - 2.2 (mean 1.4)	

Properties of the Working Fluid: Nitrogen

Table 11 lists the thermodynamic and transport properties of nitrogen obtained from the NIST standard reference database software.[62] It is observed that under the pressure and temperature conditions of the experiment, the density of nitrogen computed using the NIST software is 4.1 kg/m^3 at the inlet and at the outlet, 1.2 kg/m^3. This change in density is significant and as such, compressibility effects cannot be neglected.

The Mach number ($M = U/c$), the ratio of the flow velocity (U) to the speed of sound (c) is less than unity for adiabatic flow and less than $1/\gamma$ for isothermal flow. γ is

the ratio of specific heat at constant pressure to that at constant volume (c_p/c_v). The speed of sound for nitrogen at 0.36 MPa (3.6 bar) and 297.25 K (24.1 °C) computed using the NIST software is 352 m/s and at 296.05 K (22.9 °C) and 0.1067 MPa (1.067 bar) is 351 m/s. The inlet velocity, 71 m/s, corresponds to a Mach number of approximately 0.2. The outlet velocity, 242 m/s, corresponds to a Mach number of approximately 0.7. The ratio of specific heat is approximately 1.4 ($1/\gamma \approx 0.7$). Based on the Mach numbers, the flow is subsonic and best modeled as adiabatic.

Characteristics of the Flow Field

For straight pipes, the Reynolds number (Re) gives an indication of the flow type (i.e., laminar/transition/turbulent). The Reynolds number is the ratio of inertial (resistance to change or motion) forces, as described by Newton's second law of motion, to viscous forces. If the Reynolds number is high, inertial forces dominate and turbulent flow exists. If it is low, viscous forces prevail, and laminar flow results. The critical Reynolds number for transition to turbulent flow is $Re \approx 2300$. Using velocities from the ARTIST experiment the Reynolds number at the inlet and outlet is 274598 and 285681, respectively. Thus for a *Re* of ~2.8×10^5, flow is turbulent.

Table 11. Property of N_2 (g) from NIST Standard Reference Database Software

Property	Units	Inlet	Outlet
Temperature	°C [K]	24.1 [297.25]	22.9 [296.05]
Pressure	Bar [Pa]	3.6 [360000]	1.067 [106700]
Density	kg/m^3	4.083382	1.214588
Specific volume	m^3/kg	0.2448950	0.8233248
Internal energy	J/kg	219586.5	219219.3
Enthalpy	J/kg	307748.7	307068.0
Entropy	J/K-kg	6454.063	6812.552
c_v	J/K-kg	743.7835	743.1240
c_p	J/K-kg	1045.624	1041.433
c_p/c_v	-	1.405818	1.401425
Sound speed	m/s	351.9311	350.8357
Compressibility factor	-	0.9992888	0.9997719
Joule-Thom.	K/bar	0.2143425	0.2181484
Quality	mol/mol	Superheated	Superheated
2^{nd} Virial Coef.	m^3/kg	-0.0001815412	-0.0001899576
3^{rd} Virial Coef.	$(m^3/kg)^2$	0.000001805119	0.000001808679
Helmholtz	J/kg	-1698884	-1797637
Gibbs	J/kg	-1610721	-1709788
Heat of Vapor.	J/kg	not calculated	not calculated
Iso. Compress.	1/bar	0.2779672	0.9374185
Expansivity	1/K	0.00339496	0.003387128
Specific Heat Input	J/kg	307993	307467.8
Fugacity	Bar	3.597387	1.066755
Dielec. Con.	-	1.001919	1.000571
Viscosity	μPa-s	17.8113	17.71576
Thermal conductivity	mW/(m-K)	25.75766	25.59539
Thermal diffusivity	cm^2/s	0.06032689	0.2023492
Prandtl	-	0.7230441	0.7208244
Kinematic viscosity	cm^2/s	0.04361900	0.1458583
Surface Tension	N/m	not defined	not defined
$dp/d\rho$	$Bar/(kg/m^3)$	0.8810212	0.8782895
$d^2p/d\rho^2$	$Bar/(kg/m^3)^2$	-0.0002813096	-0.0003222457
dp/dT	Bar/K	0.01221352	0.003613251
$d\rho/dT$	$(kg/m^3)/K$	-0.01386292	-0.004113964
$d\rho/dp$	$(kg/m^3)/bar$	1.135046	1.138577

For curved pipes, the magnitude of the Dean number (*De*) indicates the characteristics of the flow much like the Reynolds number does for internal flow in a

straight pipe. The literature is peppered with an assortment of definitions for the Dean number. The working definition used herein for the Dean number {square root [(centrifugal force)(inertial force)]/(viscous force)} is taken to be proportional to the Reynolds number. Using the parameters, tube diameter (d_t), tube length (L), gas viscosity (μ), radius of curvature of the bend (R), density (ρ), and velocity (U), the Dean number is defined in the following form.

$$De = \frac{d_t U \rho}{\mu} \sqrt{\frac{L}{2R}} \qquad \textbf{(138)}$$

Defining the Dean number in this manner (138) illustrates the Reynolds number (Re) and bend curvature (δ) dependence. Thus, the simple expression characterizing flow through bends of circular cross section is $De = \mathrm{Re}/\sqrt{\delta}$. Using the Reynolds number previously computed the Dean number for the ARTIST experiment range from 87897 to 91444. Dean numbers greater than or equal to 370 constitute turbulent flow.

Table 12 summarizes the values characterizing the fluid condition for the ARTIST experiment under consideration in this analysis.

The preceding analysis indicates the flow is subsonic, compressible, turbulent, and adiabatic (i.e., there is no heat transfer). Table 13 lists for flow in a pipe of constant cross section, the trends with increasing distance as gas flows through a pipe. These trends are consistent with the conditions in this analysis: static and total pressure, density, and temperature decrease. Velocity, Mach and Reynolds number increase.

Table 12. Model Selection Criteria

Parameter	Formula	Inlet	Outlet
Gas constant [$N \cdot m/K \cdot kg$]	$R = c_p - c_v$	302	298
Specific heat ratio [-]	$\gamma = c_p / c_v$	1.41	1.4
Sound speed [m/s]	$c = \sqrt{\gamma RT}$	355	352
Mach number [-]	$M = U/c$	0.2	0.7
Density [kg/m^3]	$\rho = p/R \cdot T$	4.01	1.21
Total pressure [Pa]	$p_T = p\left(1 + \frac{\gamma - 1}{2} M^2\right)^{\frac{\gamma}{\gamma - 1}}$	370215	148328
Total temperature [K]	$T_T = T\left(1 + \frac{\gamma - 1}{2} M^2\right)$	299.7	325.3
Pressure ratio [-]	p/p_T	0.97	0.72
Dynamic pressure [Pa]	$q = p_T - p$	10215	41628
Compressibility factor [-]	$Z = p/(\rho RT)$	~ 1	~ 1
Reynolds number [-]	$\mathrm{Re} = \rho U d_t / \mu$	274598	285681
Dean number [-]	$De = \mathrm{Re}/\sqrt{\delta}$	87897	91444

Table 13. Trends in Compressible Subsonic Pipe Flow of Perfect Gases[a]

	ADIABATIC[b]
Static pressure	Decreases
Total pressure[c]	Decreases
Velocity	Increases
Density	Decreases
Temperature	Decreases
Mach number	Increases[d]
Reynolds number	Increases[e]
Stagnation temperature[f]	Constant

a. Uniform pipe or duct. Trends are with increasing distance as gas flows down the pipe.
b. The Mach number at the entrance is less than 1.0.
c. Static pressure plus dynamic pressure, $p + \frac{1}{2}\rho U^2$.
d. Tends toward the limit of 1 for long pipes.
e. Kinematic viscosity decreases with temperature.
f. Isentropic stagnation temperature.

Fully-Developed Flow

Flow in enclosed conduits like in the bend and straight sections in this investigation is classified as internal flow. For internal flow, in addition to determining

whether the flow regime is laminar or turbulent, one must determine whether the flow is in an entrance region or a fully developed flow region. For flow in a circular pipe, where the fluid enters the pipe with a uniform velocity, the fluid makes contact with the surface; viscous effect becomes important and a boundary layer begins to develop. However, when a fluid leaves a reservoir and first enters a pipe, the flow cannot truly be considered internal even though it is flowing inside of a pipe. In this region, all the principles of external boundary layer flow apply. In a circular pipe, the boundary layer simply has not grown large enough to interact, so there is no difference between flow bounded on all sides and flow bounded on one side. External flow guidelines apply until the flow becomes fully developed.

Any flow bounded by a surface will develop a region that is adjacent to the surface, in which the flow properties are different from the free stream. Friction is the primary reason for the existence of this boundary layer. As seen in Figure 17, at the pipe entrance the boundary layer is very thin. Except in the immediate vicinity of the wall, inviscid flow exists. Further downstream, the boundary layer starts thickening. The development of the boundary layer occurs at the expense of shrinking the inviscid flow region and concludes with boundary layer merger at the centerline. Boundary layer growth to the point where they meet in the middle of the pipe represents a new type of flow, fully developed flow. The distance from the entrance at which this condition occurs is termed the entrance length. Following this merger, viscous effects extend over the entire cross-section and the velocity profile no longer changes with increasing length. Therefore, in a velocity boundary layer, the flow velocity is zero at the wall because of

the assumed no-slip condition. The velocity increases as you move away from the wall until it eventually becomes a constant and is equal to the free stream value. Thus, before becoming fully developed, internal flows have velocity profiles that evolve in the streamwise direction. After a certain distance into the conduit however, the time averaged velocity profile becomes steady and uniform.

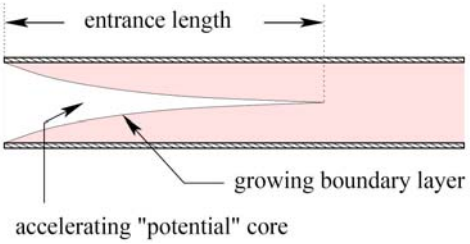


Figure 17. Entrance Length and Boundary Layer Growth

There is no precise dividing line between the potential flow region where friction is negligible and the boundary layer, but it is customary to define the boundary layer as that region where the fluid velocity parallel to the surface is less than 99% of the free stream velocity, described by potential flow theory. The laminar-transition-turbulence sequence occurs in all flows, if the surface is long enough, regardless of whether the free stream flow is laminar or turbulent. However, as the degree of turbulence in the free stream increases, the transition from laminar to turbulent flow in the boundary layer occurs earlier, that is, closer to the leading edge.

For the circular pipe, the velocity profile is parabolic for laminar flow. The profile is flatter for turbulent flow. Profile flattening in turbulent flow is due to turbulent mixing in the radial direction. The extent of the entry region depends on whether flow is

laminar or turbulent. The flow regime based on the Reynolds number in this investigation is turbulent.

The entrance length ($z_{fd,h}$) for turbulent flow range from 10 and 60 times the hydraulic diameter (D_H). Fully developed flow is assumed if ($z_{fd,h}/D_H$) exceeds 10.

$$10 \le \frac{z_{fd,h}}{D_H} \le 60 \qquad \textbf{(139)}$$

Before making an allowance for the entrance length, the boundary layer growth at high Reynolds number must be considered. When the Reynolds number is high enough to induce turbulence, an entrance region is followed by fully-developed flow. However, the structure of the entrance region is somewhat more complicated. There is a laminar boundary layer near the entrance, which transitions to a turbulent boundary layer eventually. The transition to turbulence takes place approximately at a longitudinal Reynolds number Re = 3 x 10^5. Rule of thumb: turbulent flow is fully-developed within about 25 to 40 pipe diameters, though some details of the turbulent flow field may not appear for 80 diameters or more. Even in the fully developed region, the flow immediately next to the wall is still laminar. This is the laminar sublayer. In this problem, since the Reynolds number is slightly less than 3 x 10^5, 200 diameters were used to assure fully developed flow was attained in the k-ω calculations.

Turbulence Structures: Estimate of Scales

The length scale of turbulence is important for determining the appropriate grid spacing in the computational domain. Structures called eddies dominate the flow. Eddies can be any size, ranging from the entire width of the pipe to microscopic

structures. These eddies rotate and mix the flow in such a way that it is almost impossible to solve analytically. Equations for estimating the scales of turbulence[63] are given in Table 14.

Table 14. Equations for Computing Scales of Turbulence

Eddies	Size, length scale	Frequency, time scale
Largest Eddies	$l_L = d_t/2$	$f_L = 2u^*/d_t = 0.4\left(\overline{U}/d_t\right)\mathrm{Re}^{\frac{1}{8}}$
Energy Containing Eddies	$l_e = 0.05 d_t\,\mathrm{Re}^{\frac{1}{8}}$	$f_e = u^*/l_e = 20\left(u^*/d_t\right)\mathrm{Re}^{\frac{1}{8}} \approx 4\left(\overline{U}/d_t\right)$
Most Dissipative Eddies	$l_d = 20 d_t\,\mathrm{Re}^{-0.78}$	$f_d = \left(6u^{*2}/\nu\right)/\mathrm{Re}^{0.44} = 0.02\left(\overline{U}/d_t\right)\mathrm{Re}^{0.56}$
Kolmogorov	$l_K = 4 d_t\,\mathrm{Re}^{-0.78}$	$f_K = \left(17u^{*2}/\nu\right)/\mathrm{Re}^{0.44} = 0.06\left(\overline{U}/d_t\right)\mathrm{Re}^{0.56}$

For the pipe analyzed in this investigation (i.e., 0.01687 m internal diameter), nitrogen flow velocities range between 71-242 m/s (Re ~ 2.8 x 10^5), the size of the largest eddies in the core region have a frequency in the range 8.055 – 28.16 kHz and is about 50% of the pipe diameter. The "energy containing" eddies have a frequency between 16.835 – 58.566 kHz. The "most dissipative" eddies have a frequency range between 94 kHz – 333 MHz. At the smallest eddies (of the order of the Kolmogorov scale) energy is dissipated into heat. Table 15 lists eddy size and associated frequency computed from the experiment's inlet and outlet data.

Table 15. Computed Scales of Turbulence

	Eddy size & frequency at inlet		Eddy size & frequency at outlet	
Eddies	Size (m)	Frequency (Hz)	Size (m)	Frequency (Hz)
Largest Eddies	8.44E-03	8055	8.44E-03	28160
Energy Containing Eddies	4.04E-03	16835	4.06E-03	58566
Most Dissipative Eddies	1.93E-05	93507	1.87E-05	332587
Kolmogorov	3.86E-06	280521	3.75E-06	997761

Flow Field Results

Table 16 shows results from both calculations; realizable k-ε and standard k-ω. In these calculations, constant mass flow was maintained at the inlet, the code held constant temperature at the value specified at the boundary. The code adjusted the velocity to conserve mass flow. Velocities and pressures compare favorably with the Phase I experiment of the ARTIST project. The realizable k-ε model predicted inlet velocity and pressure of 67 m/s and 0.36 MPa, respectively. While the standard k-ω model predicted nearly the same velocity magnitude, the pressure was slightly higher, 0.38 MPa. Outlet velocity and pressure using the realizable k-ε model predicted 225 m/s and approximately 0.099 MPa. The Phase I measured values reported for velocity and pressure at the inlet were 71 m/s and 0.36 MPa. For the outlet, the reported measured values for velocity and pressure were 242 m/s and 0.1067 MPa.

Velocity

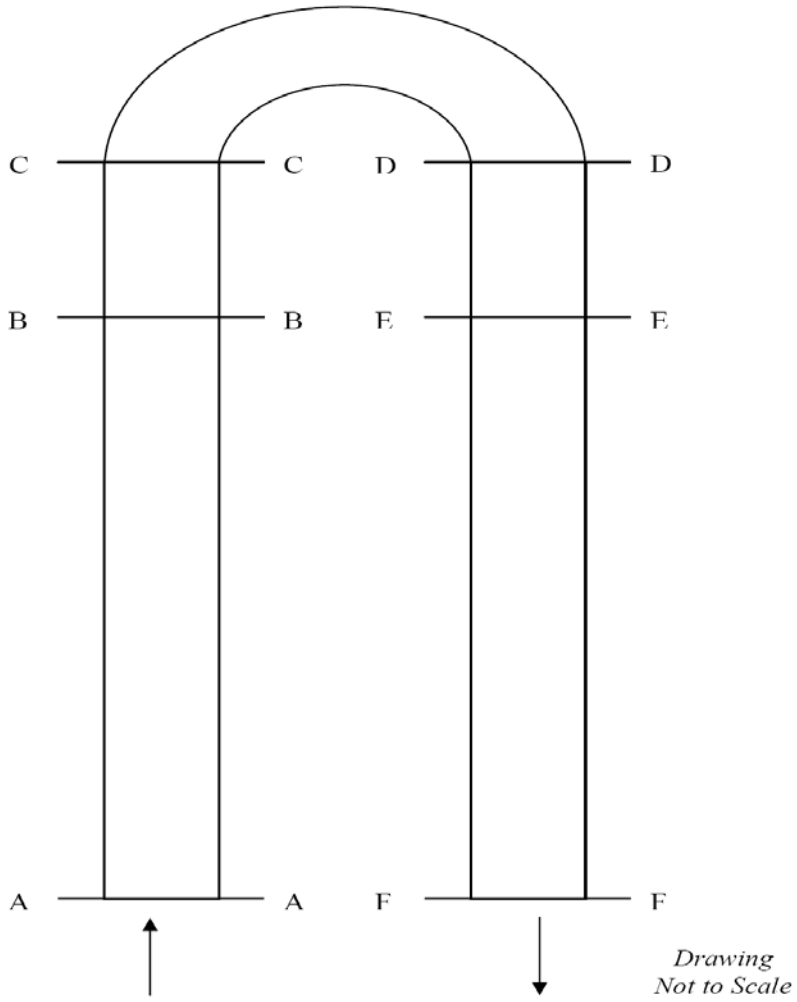
Figure 18a shows the locations of the cross sections referenced in the following discussion. Figure 18b shows for the U-shaped tube velocity contours (from rke calculation) at the indicated location. The contours for section A-A displays uniform velocity at the inlet. Section B-B located nearly 500 diameters from the inlet shows fully developed flow in the straight section. Taking the datum location at the bend entrance (9.214 m) Figure 19 shows in terms of length per pipe diameter (L/D) fully developed flow exists in the pipe inlet section for at least 450 diameters upstream of the bend. However, at the bend entrance, plane C-C, flow is somewhat accelerated near the

inner wall. Figure 20 shows the onset of this behavior occurs within approximately one diameter upstream of the bend. At the bend exit, plane D-D, the high velocity region of the primary flow move on the horizontal plane near the outer wall. At station E-E located downstream from the bend exit, centrifugal force is no longer active. This station is sixteen diameters downstream from the exit of the bend. As show in Figure 21, the bend still influences the flow. Flow acceleration is toward the outer wall. The separation zone on the convex wall at 180° (D-D) persists to E-E located sixteen diameters downstream of the bend exit. There is however a return to fully developed flow some four hundred and fifty (450 d_p) pipe diameters from the exit in the straight section, as shown in Figure 22.

Table 16. Flow Field Results

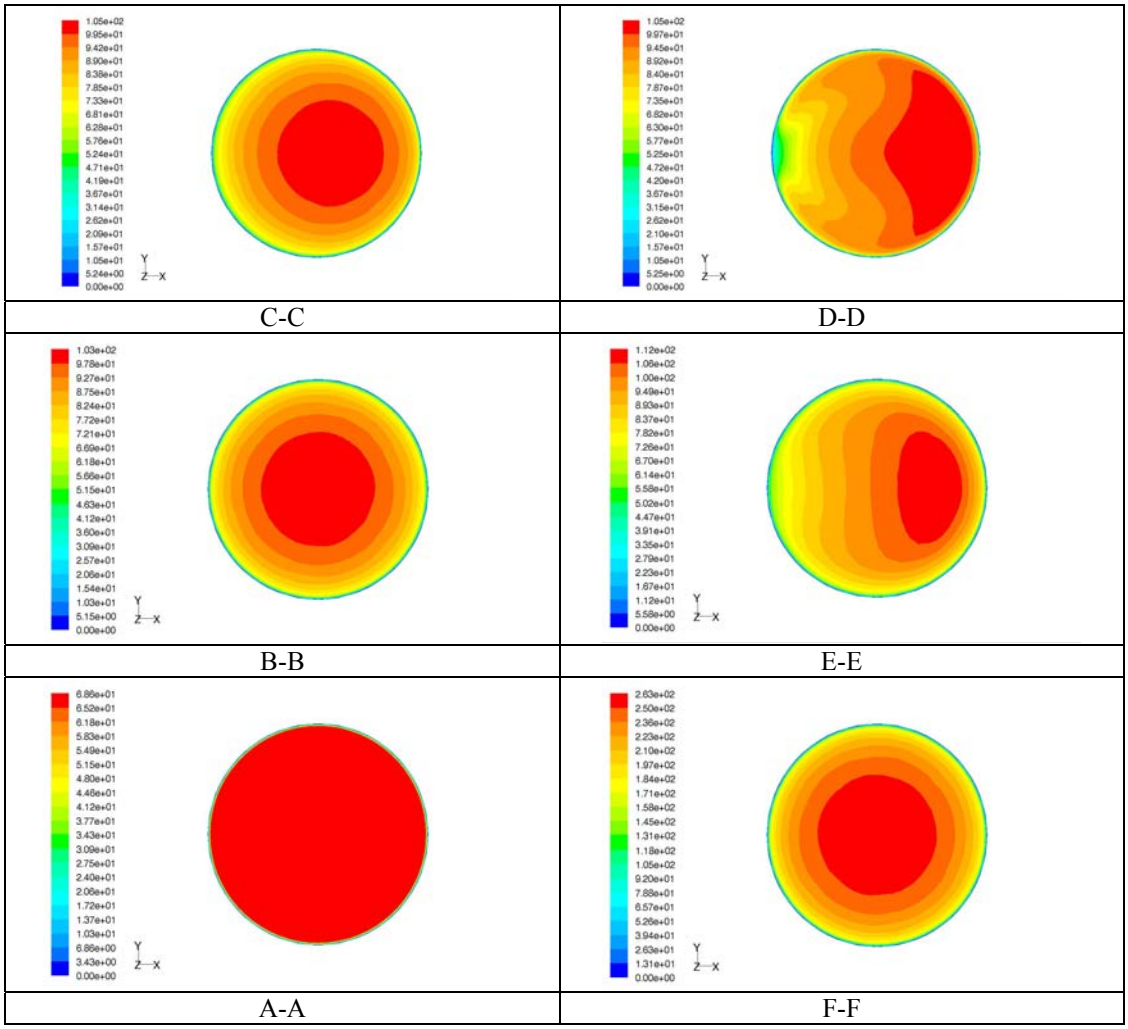
	*Velocity Magnitude (m/s)	*Pressure, Static (Pa)
System Inlet		
k-ε	67	362184
k-ω [straight section only]	67	386063
Experiment	71	360000
Test Section entrance (16 dia.)		
k-ε	89	277648
k-ω [bend only]	-	-
k-ω [straight section only]	87	295016
Experiment	-	-
u-bend entrance (R_c = 82.3 mm)		
k-ε	90	274614
k-ω [bend only]	88	291903
k-ω [straight section only]	88	292060
Experiment	-	-
u-bend exit (R_c = 82.3 mm)		
k-ε	92	270311
k-ω [bend only]	91	283507
Experiment	-	-
Test Section exit (16 dia.)		
k-ε	93	266775
k-ω	-	-
Experiment	-	-
System Outlet		
k-ε	225	99320
k-ω	-	-
Experiment	242	106700

*GUI: Report\Surface Integrals\Report Type = Area Weighted Average



(a) U-Tube: Location of Cross-Sections

Figure 18. Contours of Velocity Magnitude around the Pipe (m/s)



C-C	D-D
B-B	E-E
A-A	F-F

Note: Full-sized diagrams provided in APPENDIX B.

(b) Velocity Contours

Figure 18. Continued

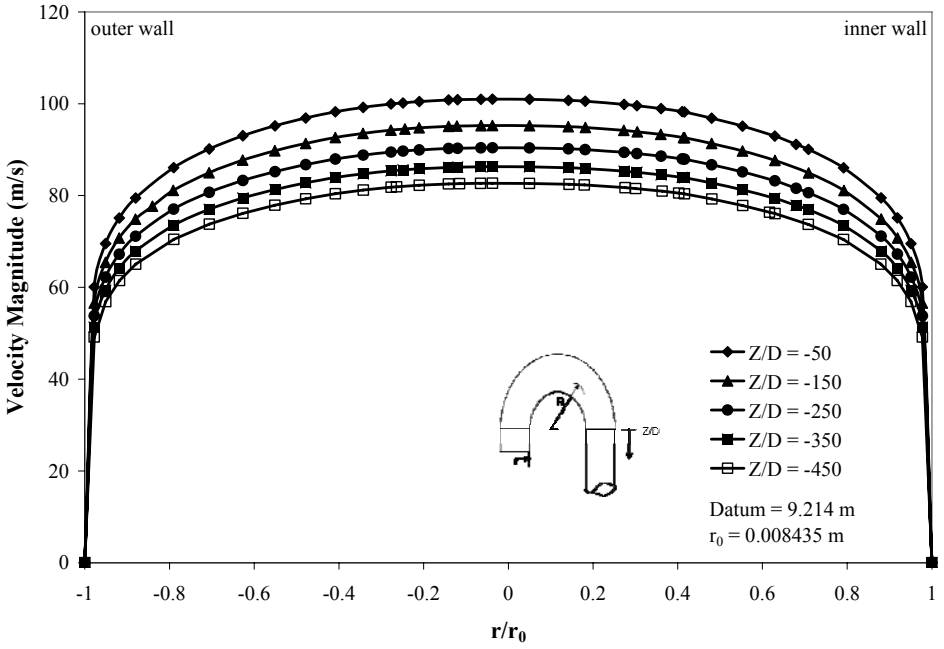


Figure 19. Velocity Magnitude, Inlet

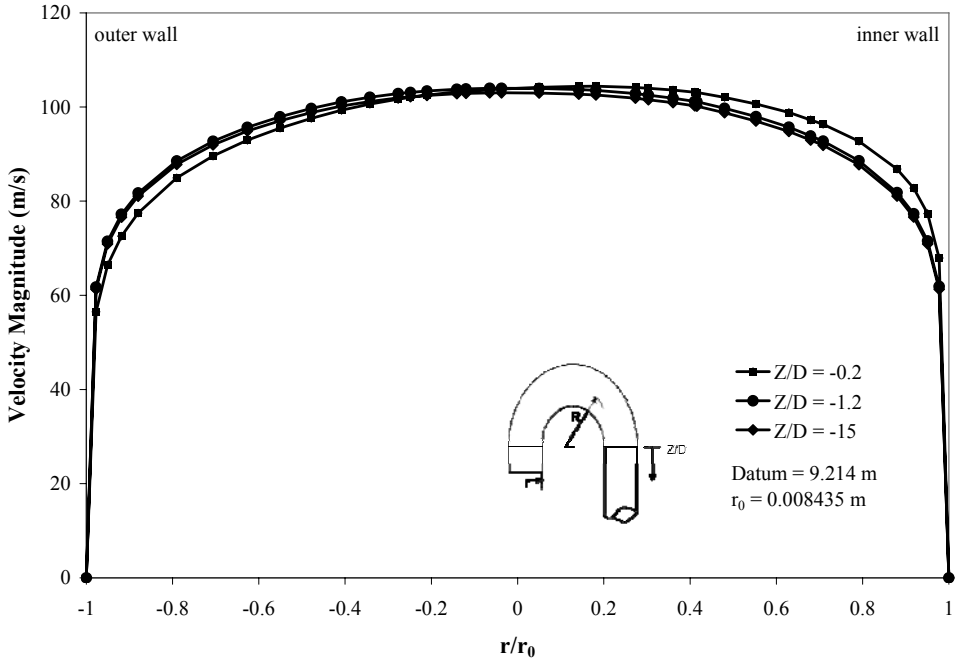


Figure 20. Velocity Magnitude, Downstream (Inlet-Side) of Bend

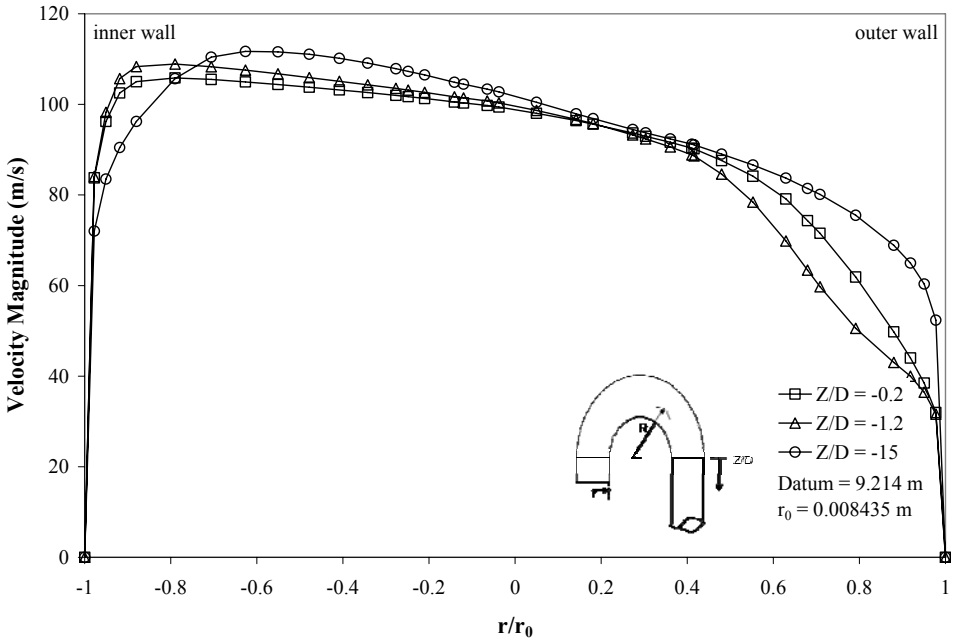


Figure 21. Velocity Magnitude, Upstream (Outlet-Side) of Bend

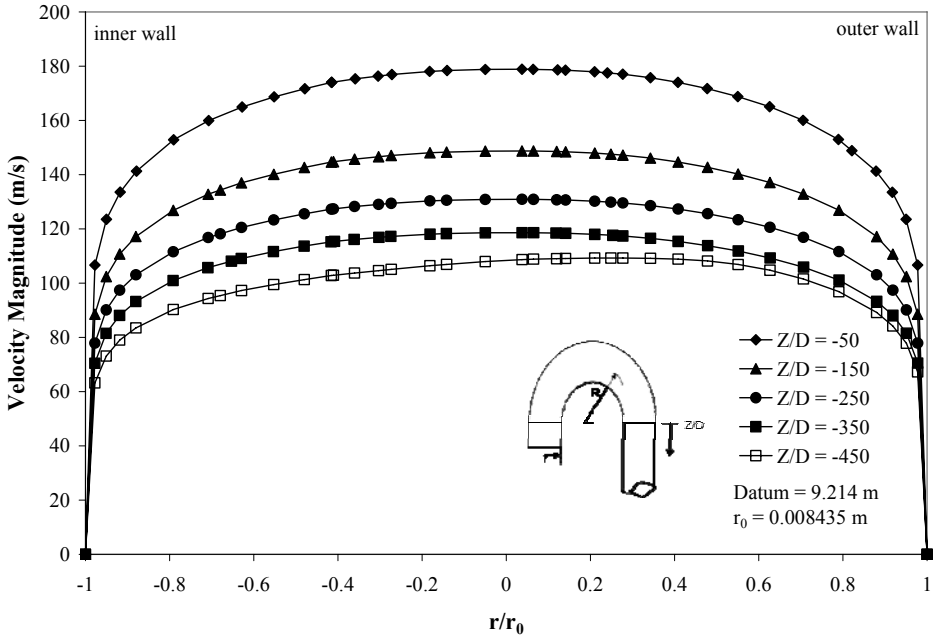


Figure 22. Velocity Magnitude, Outlet

In a curved pipe, the Dean number plays the role of the flow Reynolds number. At high Dean number ($De > 370$); centrifugal force leads to an increase in axial velocity that results in an increased amount of fluid drawn into the boundary layer near the outer bend. Secondary boundary layers adjust by thinning near the outer bend and thickening near the inner bend. There is no analytical model describing the flow field in the range encompassing large Dean number flows, but it is postulated that there is a point at which flow separation begins due to the thickening effect of the secondary boundary layer.

Taking a closer look at the bend, the 0°, 45°, 90°, 135°, and 180° planes (see Figure 23) facilitated examination of the flow field for the aforementioned behavior and general flow features. The evaluation includes both the realizable k-ε (rke) and standard k-ω (skw) results from the computational flow field analysis.

Figure 24 features velocity profiles for the entrance, mid-plane, and exit of the bend obtained from application of the turbulence models. Figures 25 through 29 displays velocity magnitude contours superimposed on velocity vectors at 45° intervals from bend inlet until the exit (i.e., 0°, 45°, 90°, 135°, and 180°). Velocity magnitude contours show flow asymmetry at entry to the bend. The realizable k-ε model shows flow slightly accelerated towards the inner wall (Figure 25 (a)). The standard k-ω model distributes the velocity in a slightly different manner due to resolution of the near wall region (Figure 25 (b)). Using directional designations (i.e., N, S, E, W) for orientation of maximum velocity, the rke model presents an eastwardly direction while a southward and slightly eastern direction is prescribed with the standard k-ω model. However, by 45° (Figure 26) both calculations agree on the maximum velocity direction towards the

concave (outer) wall. These two calculations differ in that the rke model at 45° (Figure 26 (a)) predicts on the convex wall development of the separation bubble, a region of low velocity flow near the inner wall. For both rke and skw, this separation zone on the convex wall extends from the location of onset to over a bend height downstream of the 180° location. By 90° in the skw calculation (Figure 27 (b)), a low velocity region begins to develop on the convex wall. There is continued development of this region at 135° for the skw model (Figure 28 (b)) however, the rke shows separation of the high velocity region as it wraps around the ever-intruding low velocity flow. Regardless of flow development at 180° the regions of high and low velocity are nearly the same as seen in Figure 29. Figure 29 illustrates flow separation at the exit on the convex side of the tube.

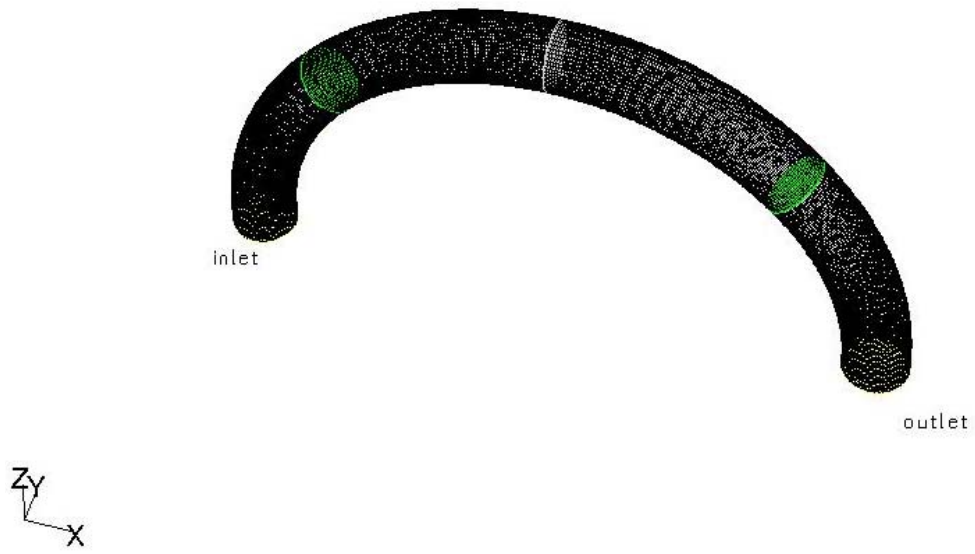


Figure 23. Plane Placement in the U-Bend

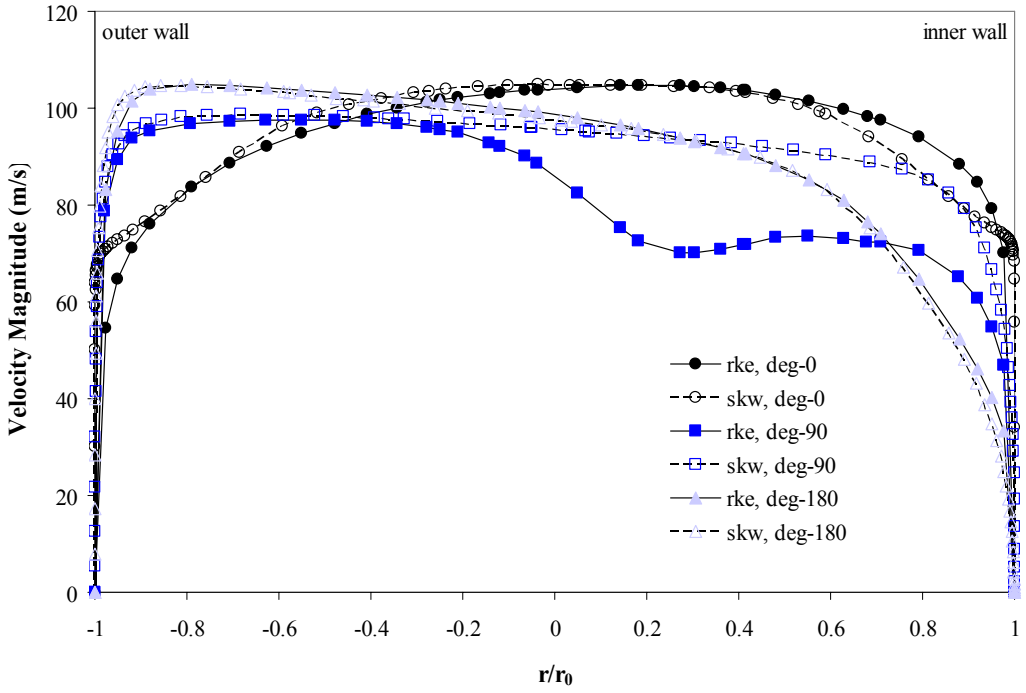
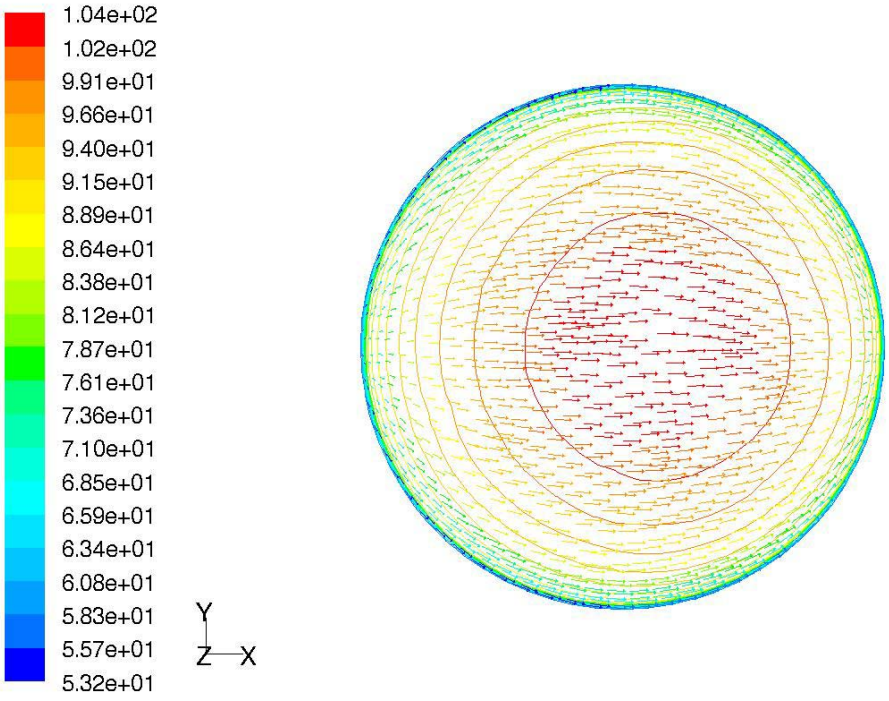
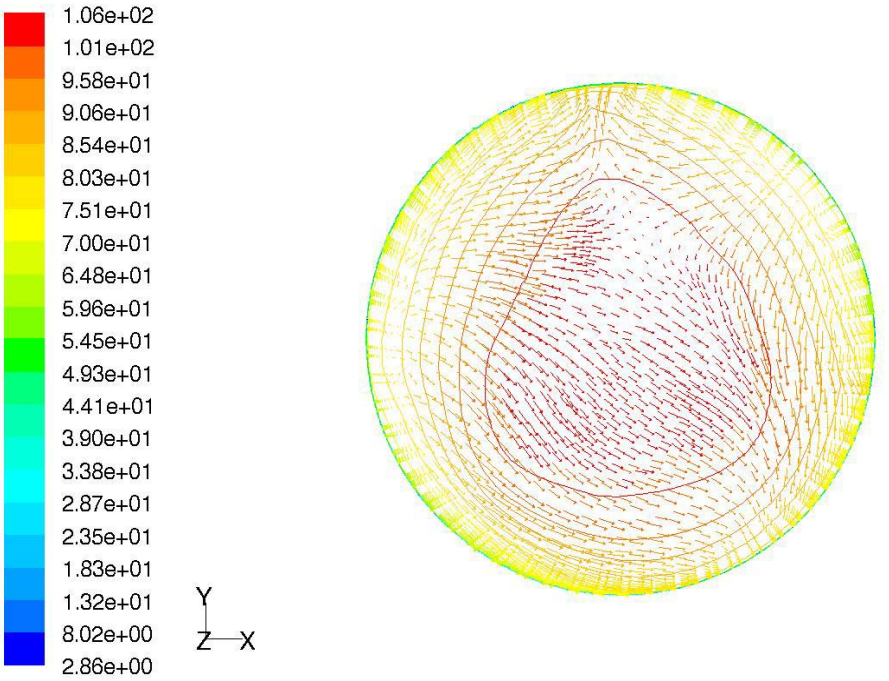


Figure 24. Velocity Magnitude, Bend

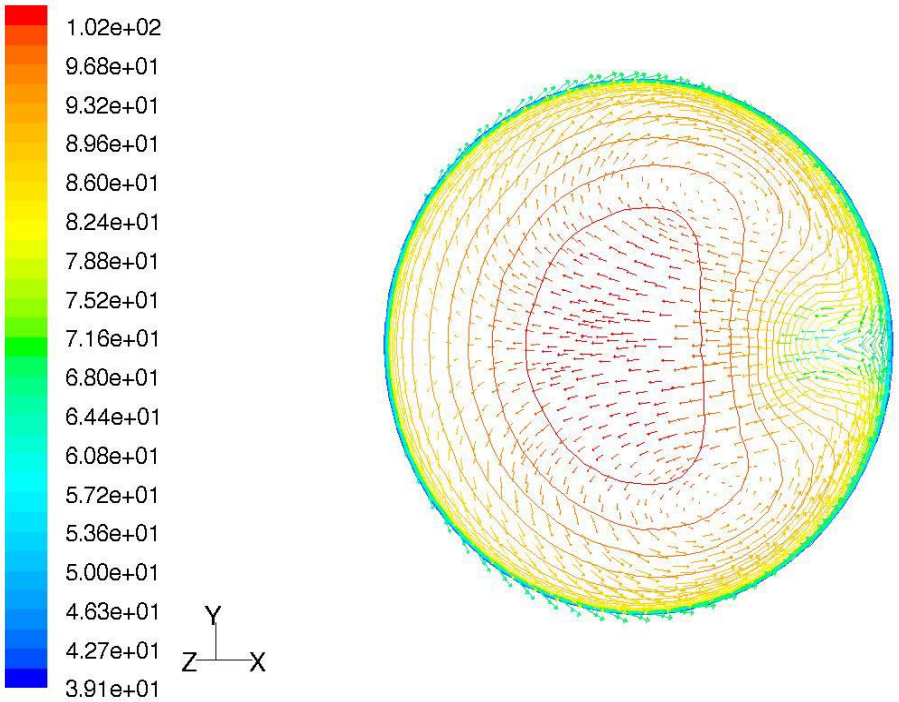


(a)

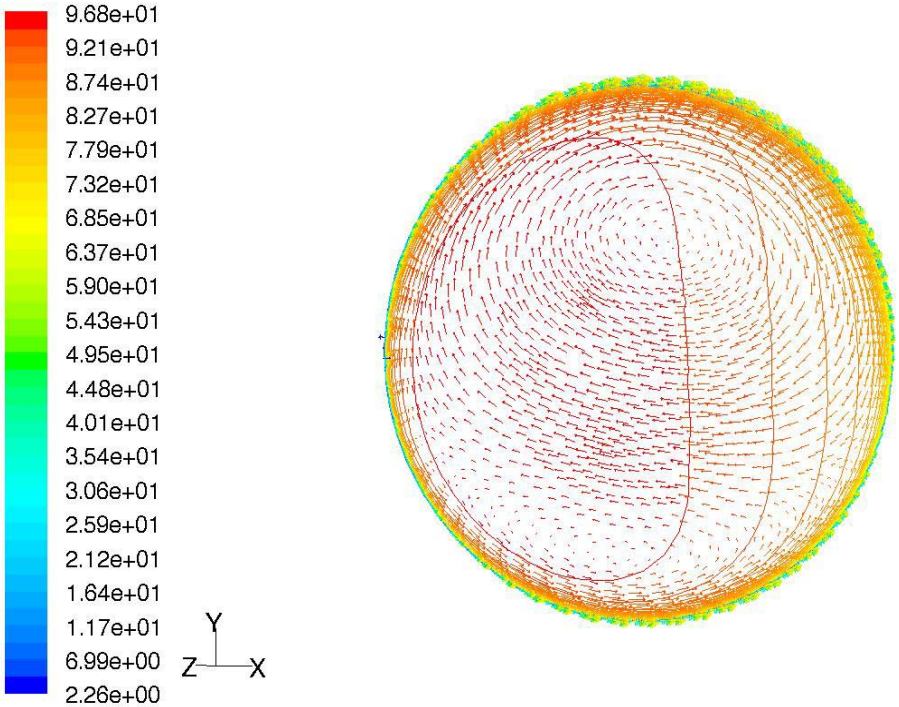


(b)

Figure 25. Velocity Contours Superimposed on Velocity Vectors at 0°

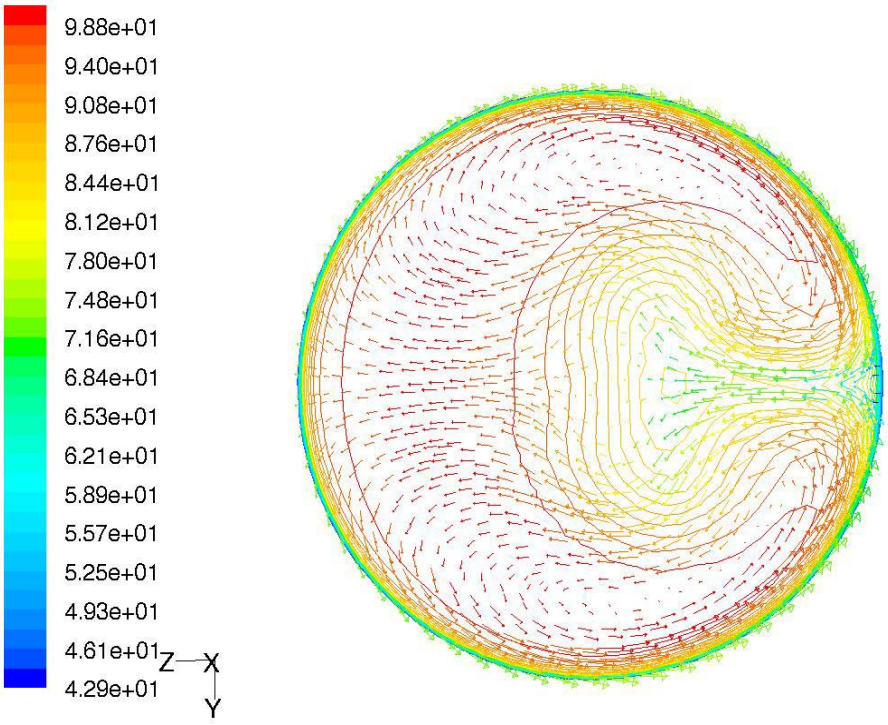


(a)

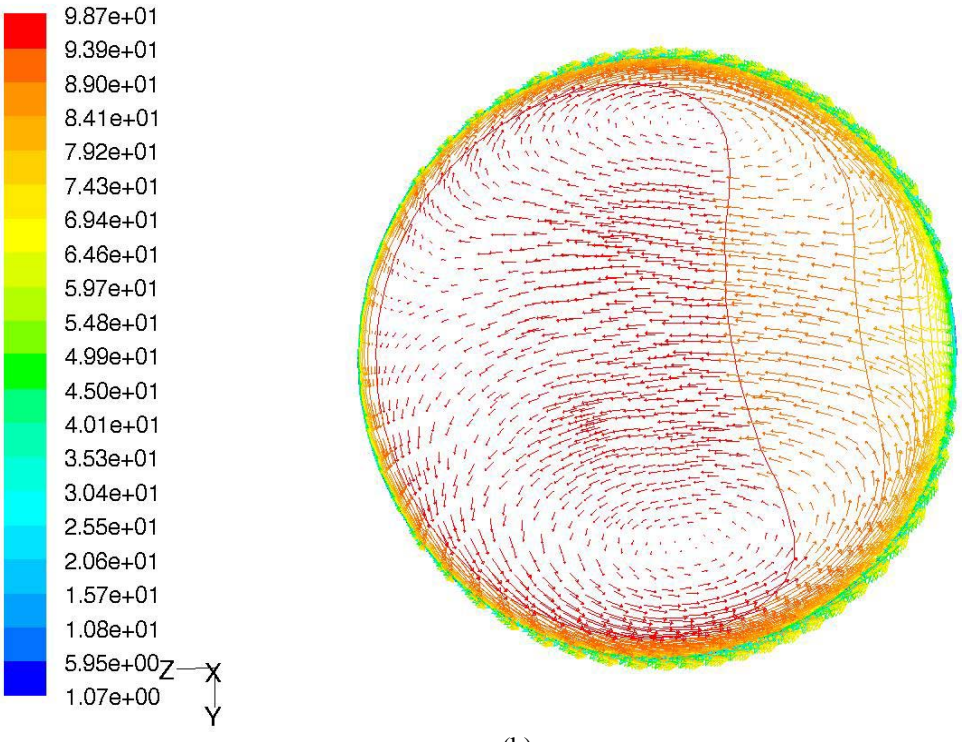


(b)

Figure 26. Velocity Contours Superimposed on Velocity Vectors at 45°

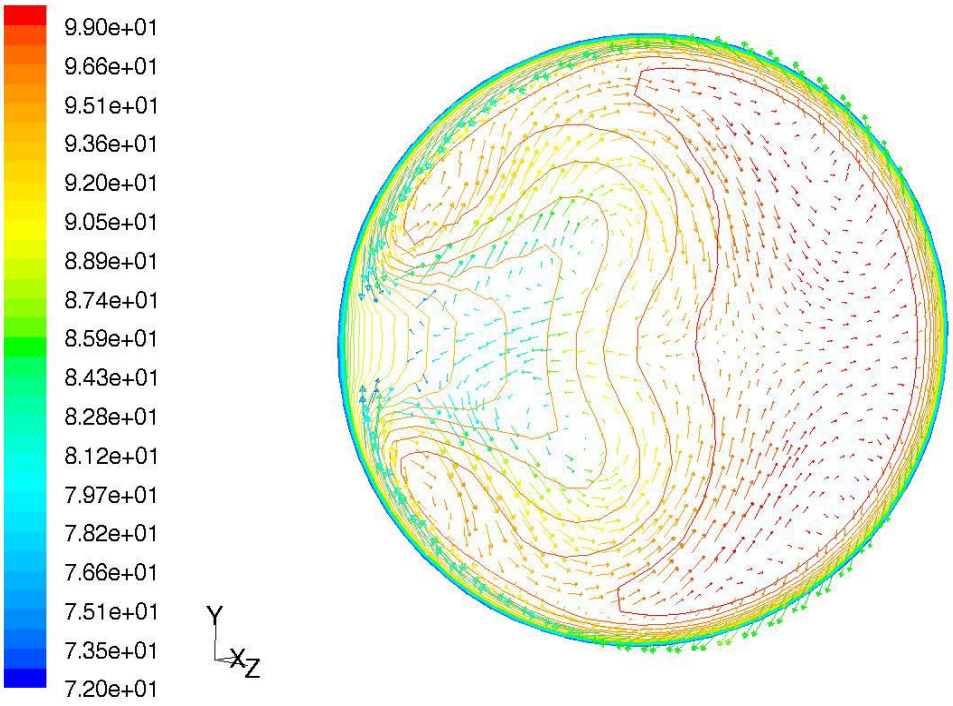


(a)

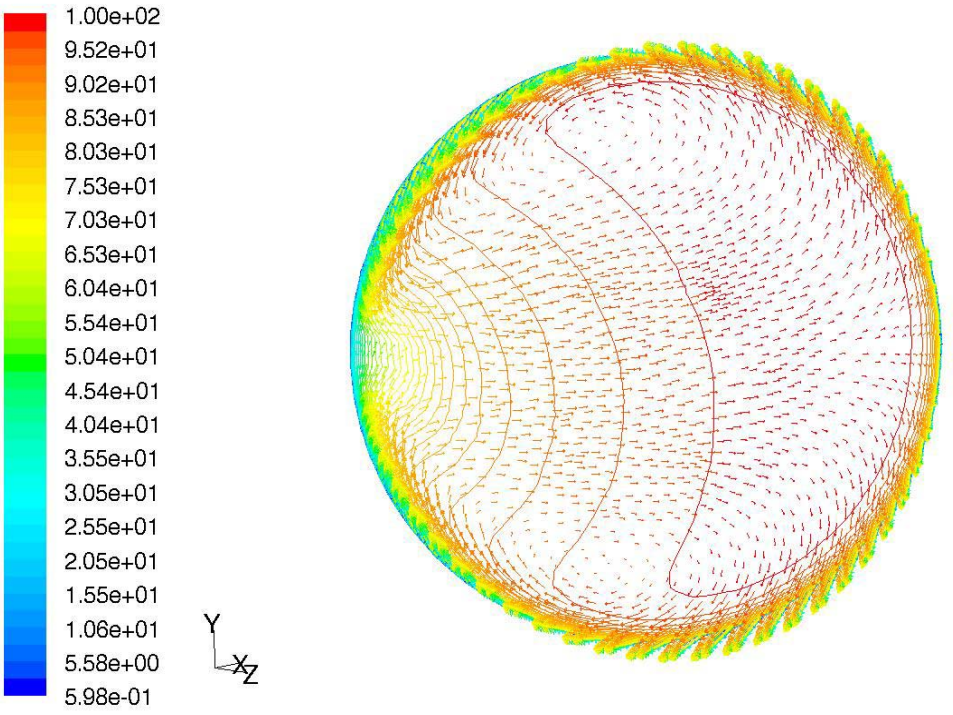


(b)

Figure 27. Velocity Contours Superimposed on Velocity Vectors at 90°

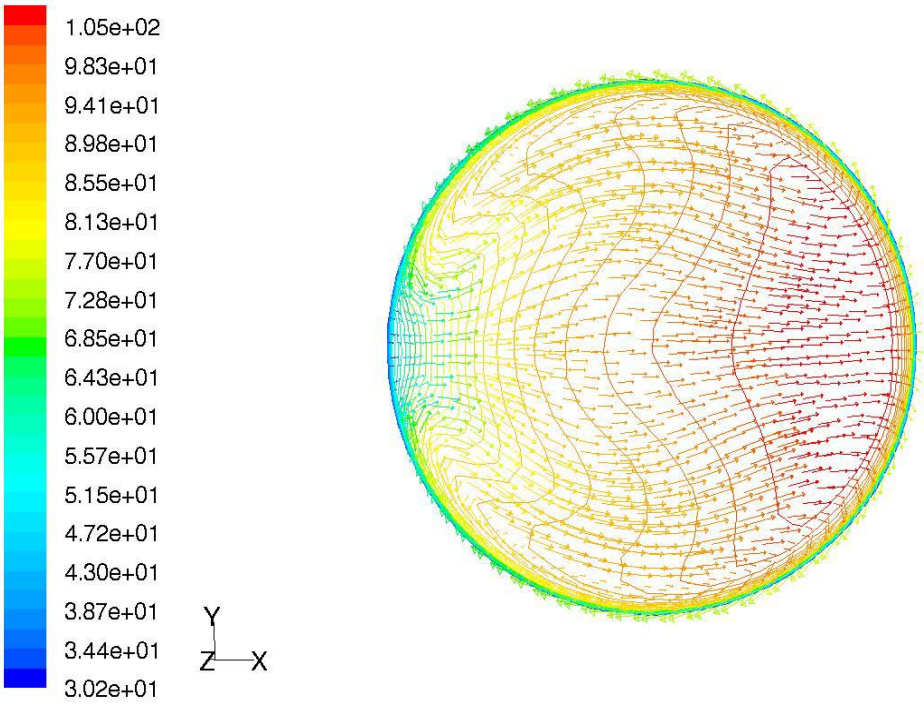


(a)

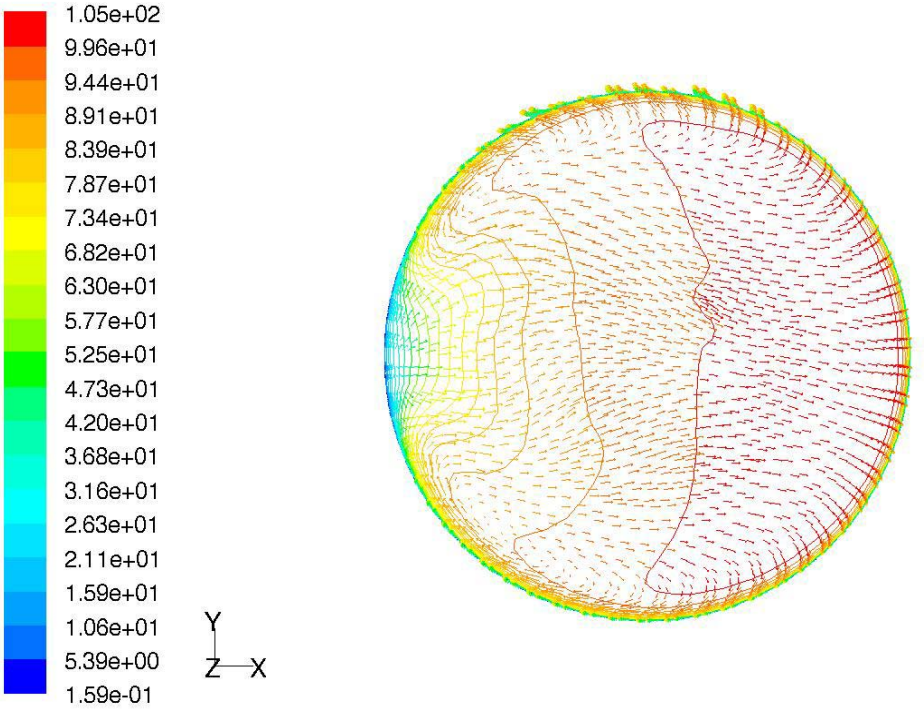


(b)

Figure 28. Velocity Contours Superimposed on Velocity Vectors at 135°



(a)



(b)

Figure 29. Velocity Contours Superimposed on Velocity Vectors at 180°

Vorticity

The descriptor for flow with vorticity is rotational; this is a key characteristic of turbulent flow. Vorticity (the curl of the velocity; $\varphi = \nabla \times \vec{V} = 2\omega$) represents the measure of the rotation of fluid elements. The magnitude of the vorticity vector is a measure of the rotation of a fluid element as it moves in the flow field. Thus, the vorticity is related to the angular velocity or spin of the fluid. A characteristic of high Reynolds-number turbulence is that the vorticity has intense, small-scale, random variations in both space and time. The spatial scale for vorticity fluctuations is the smallest in the continuum of turbulent scales, the Kolmogorov length scale.

Inlet (Figure 30), bend (Figure 31), and outlet (Figure 32) vorticity profiles reveal the influence of this parameter on wall-bounded flows. Consistently seen is that the magnitude of the vorticity in the near-wall region is significantly large, negligible in the bulk. The bend influences the near-wall vorticity upstream by increasing the fluid rotation.

In Figures 33 through 37 velocity contours overlay the in-plane vorticity vectors. As seen in Figure 33, as flow enters the bend, vorticity vectors in the plane are directed towards the inner wall of the bend. The rke result shown in Figure 33a shows as flow enters the bend vorticity vectors in the plane are directed towards the inner wall of the bend. The standard k-ω calculation on the other hand permits capture of fluid circulation and reveals wall influence on wall-ward directed flow, Figure 33b. By 45° (Figure 34), two vortices form in the asymmetric flow. The larger of the two, a clockwise rotating vortex develops in the upper portion of the pipe near the inner wall,

Figure 34a. In the lower portion, a smaller counterrotating vortex develops in close proximity to the wall. At 90° (Figure 35), the region of low velocity separates the stream of vectors still with most of the vorticity feeding the clockwise rotating vortex in the upper region, Figure 35a. The counterrotating vortex in the lower half continues to develop. As seen in the skw results (Figure 35b) in-plane rotation is practically symmetric, relatively speaking. By 135° (Figure 36), the nearly equal size vortices migrate in the direction of low velocity. At 180° (Figure 37), both vortices persist but with much less pronounced rotation as if the flow transitions from the curved geometry to the straight section.

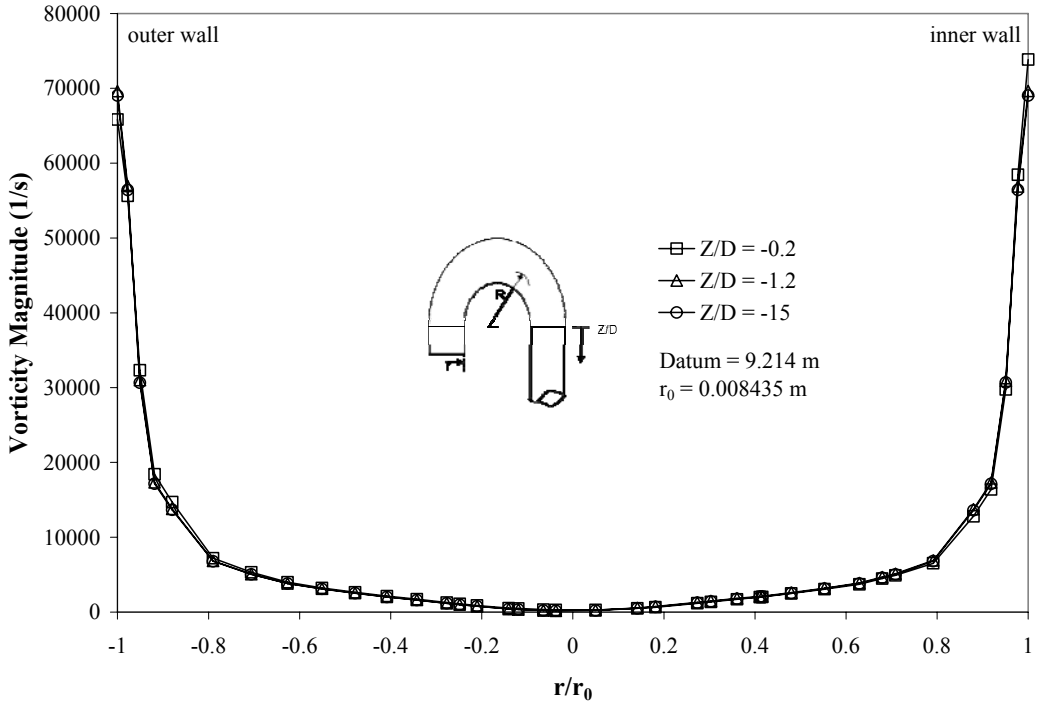


Figure 30. Vorticity Magnitude, Downstream (Inlet-Side) of Bend

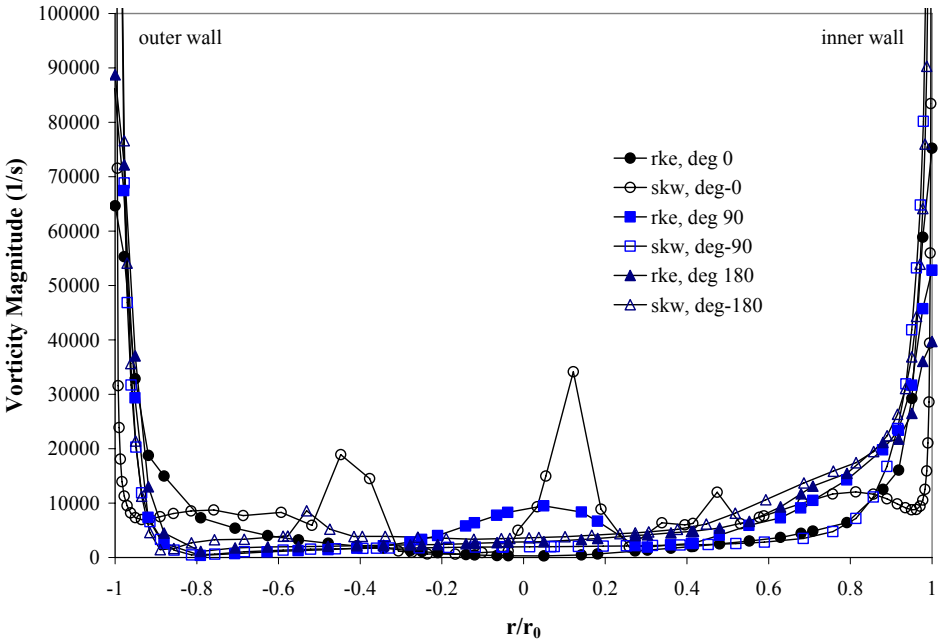


Figure 31. Vorticity Magnitude, Bend

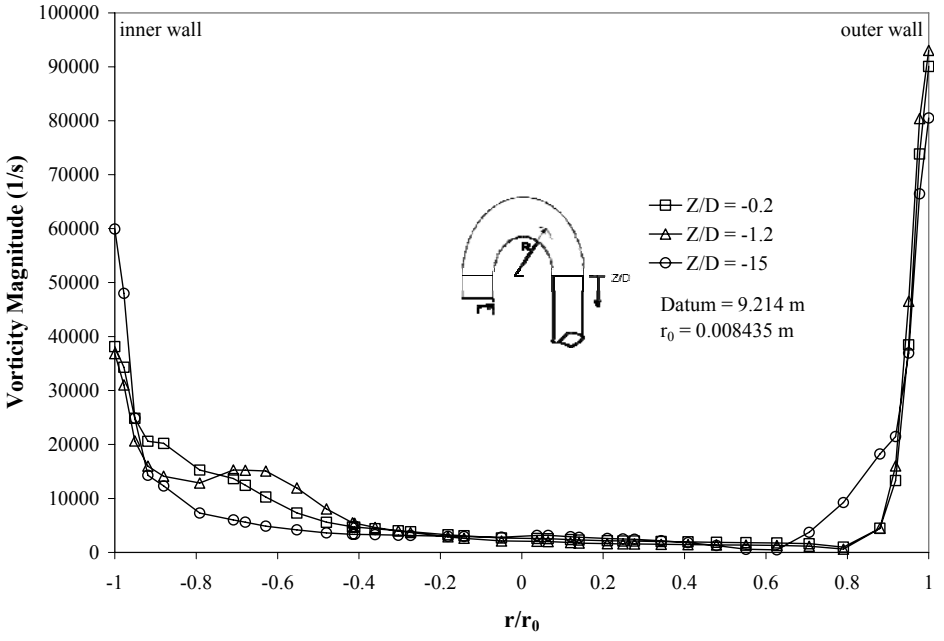
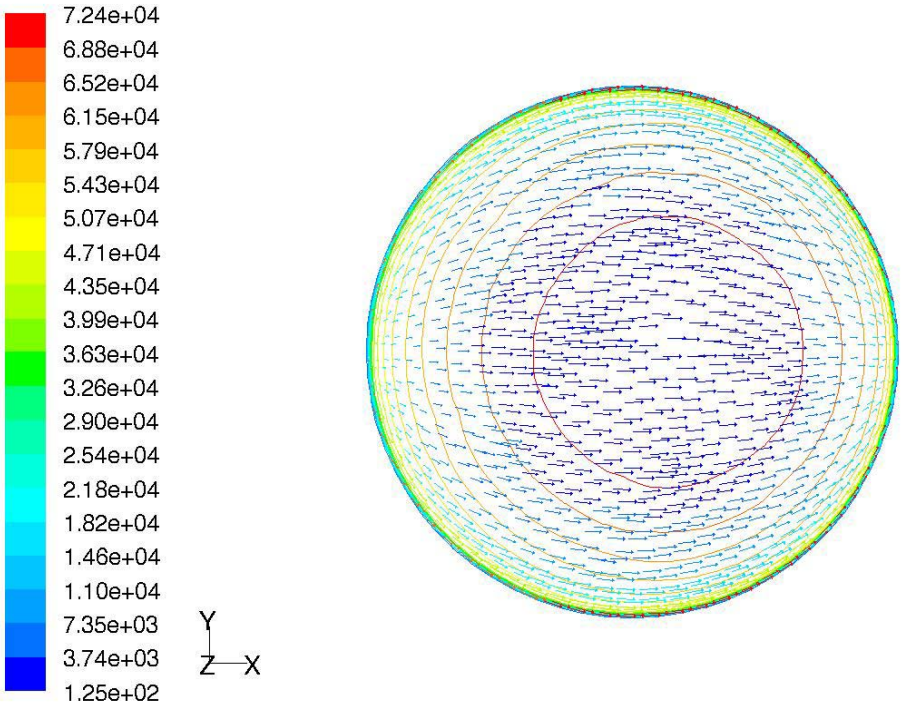
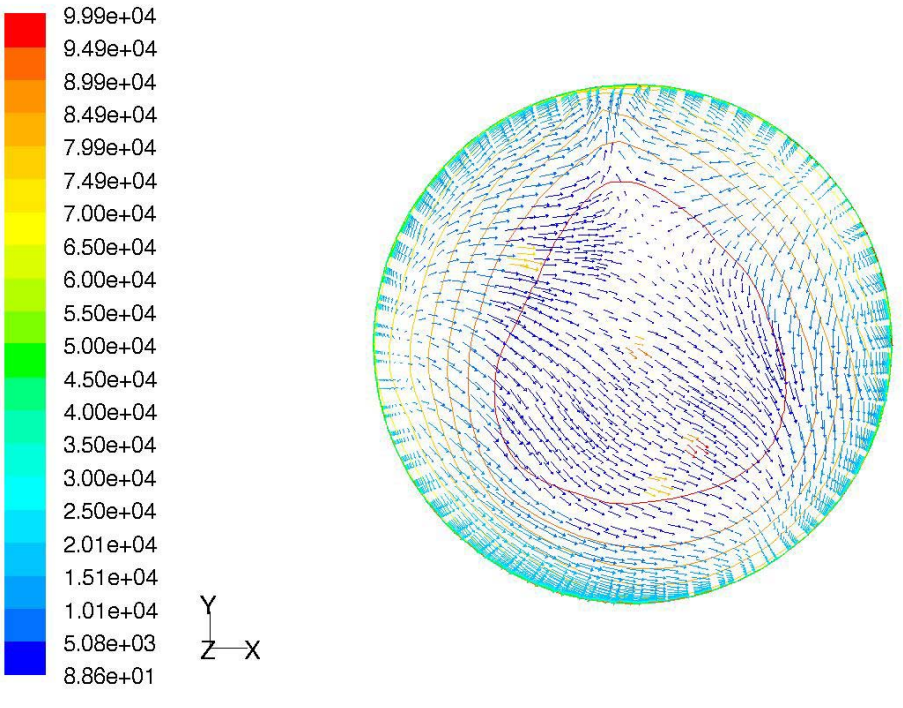


Figure 32. Vorticity Magnitude, Upstream (Outlet-Side) of Bend

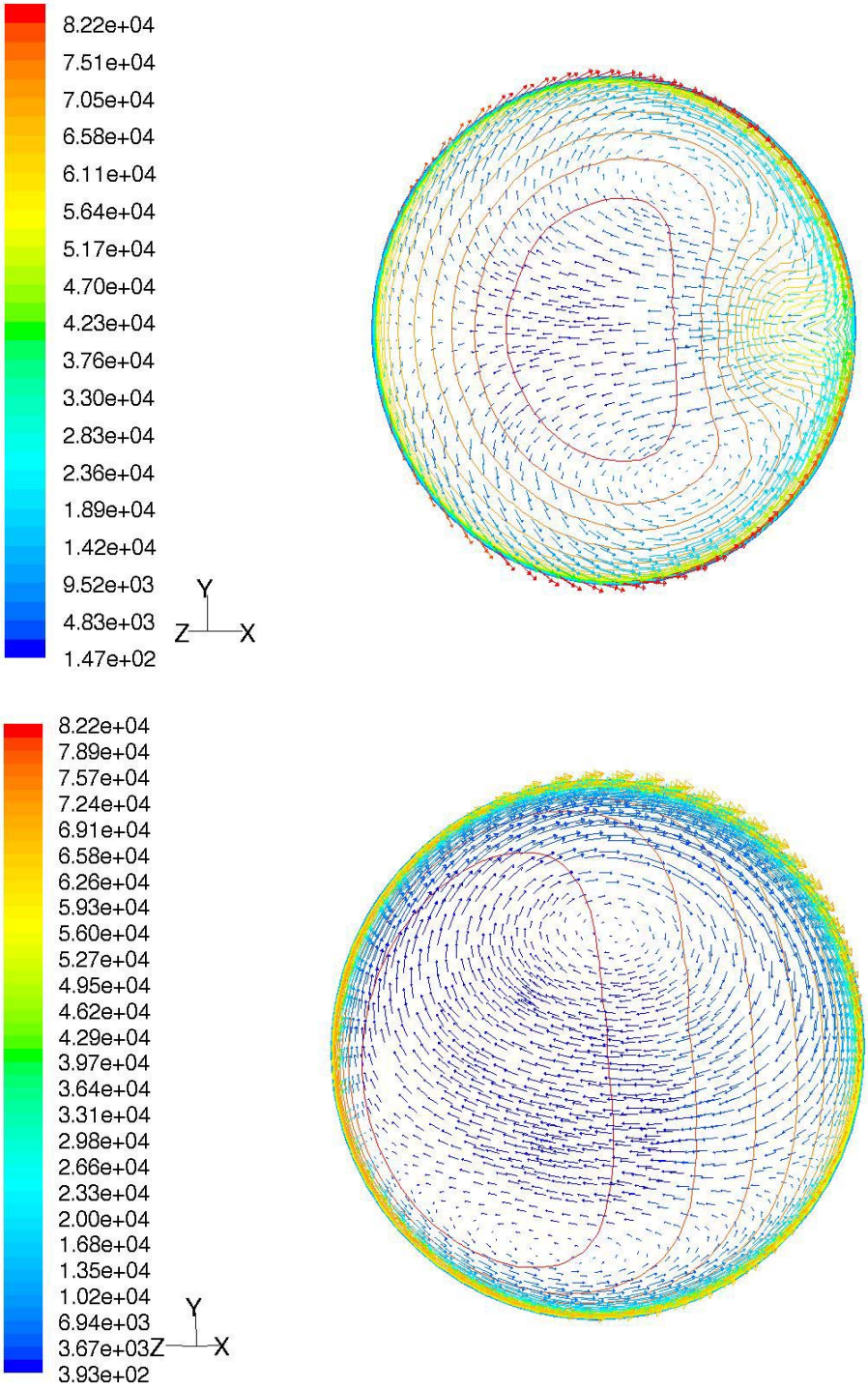


(a)



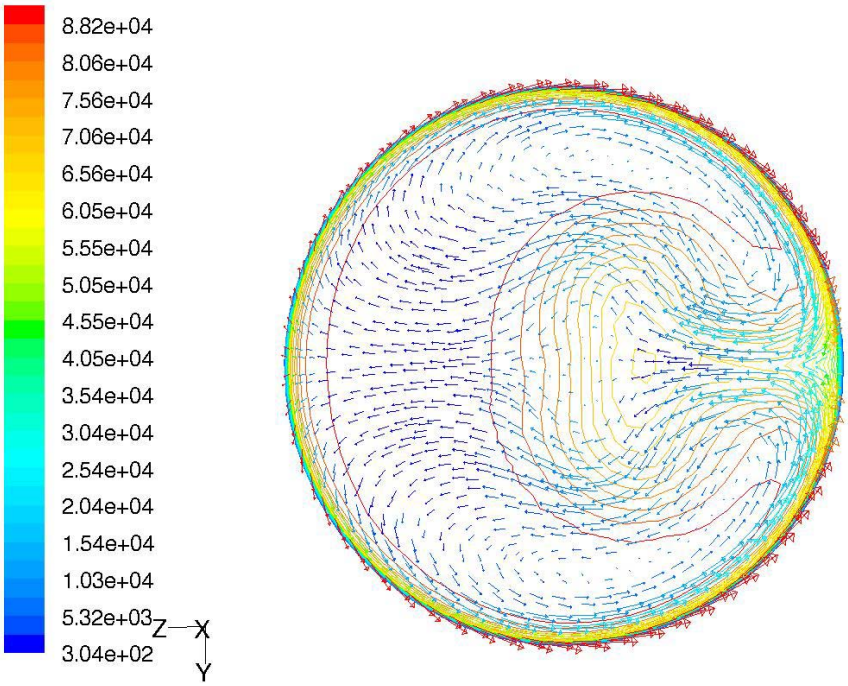
(b)

Figure 33. Velocity Contours Superimposed on Vorticity Vectors at 0°

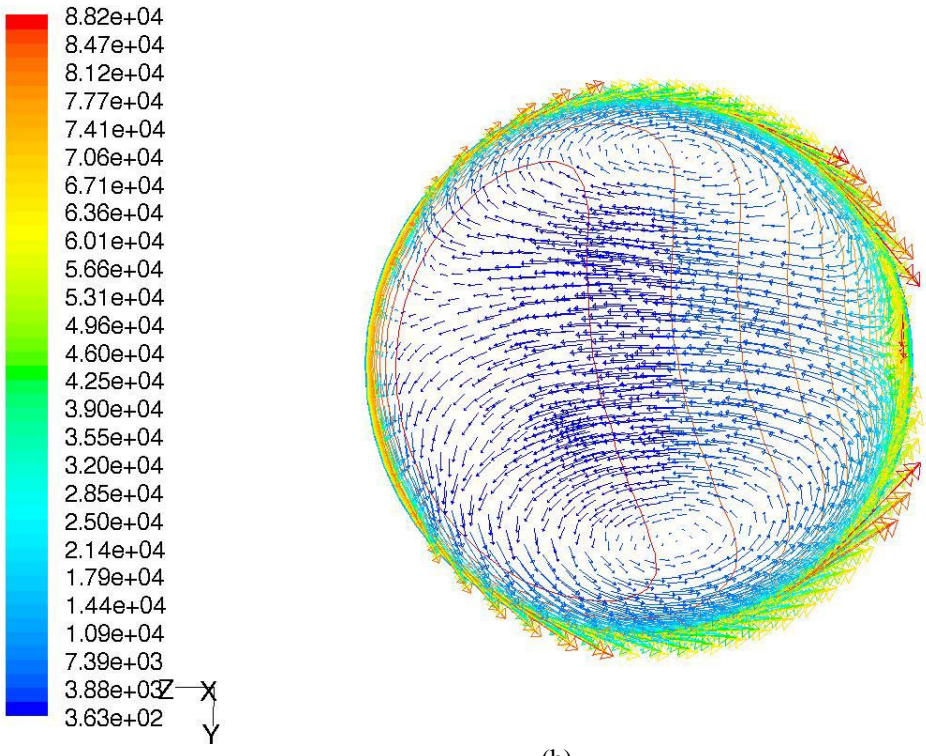


(b)

Figure 34. Velocity Contours Superimposed on Vorticity Vectors at 45°

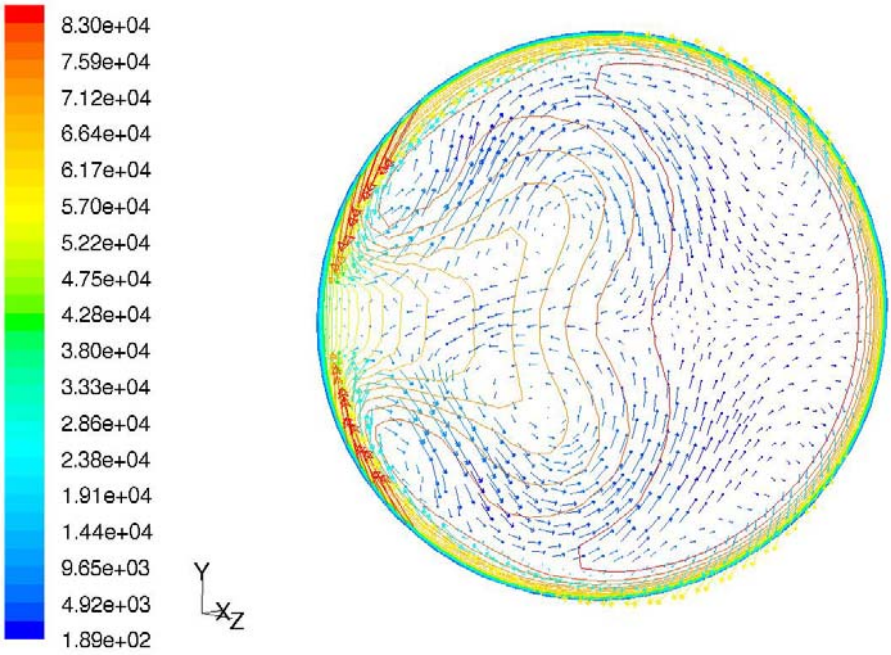


(a)

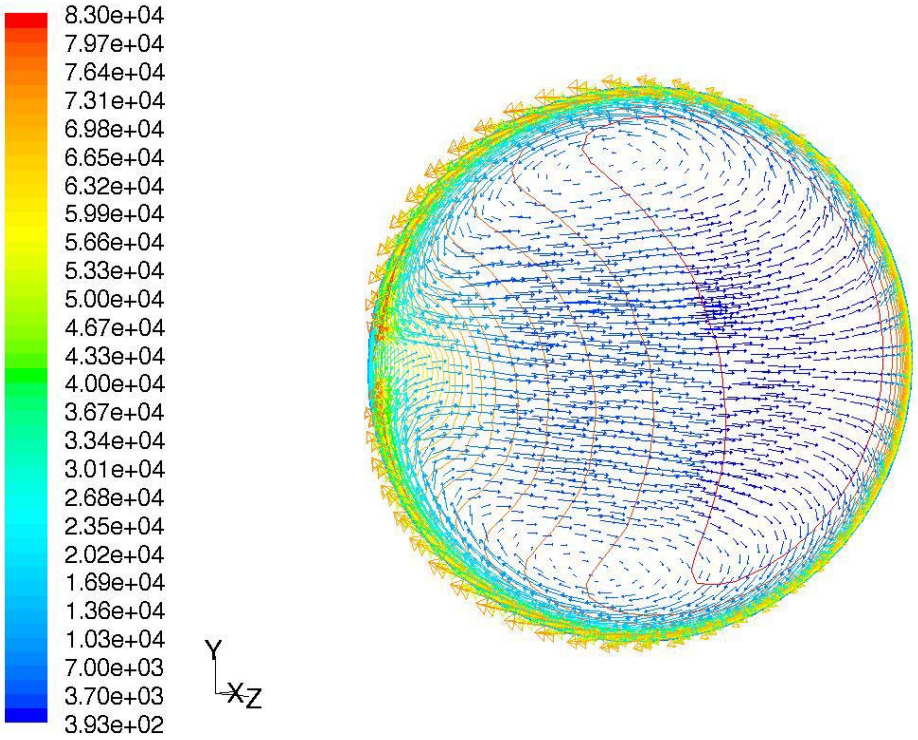


(b)

Figure 35. Velocity Contours Superimposed on Vorticity Vectors at 90°

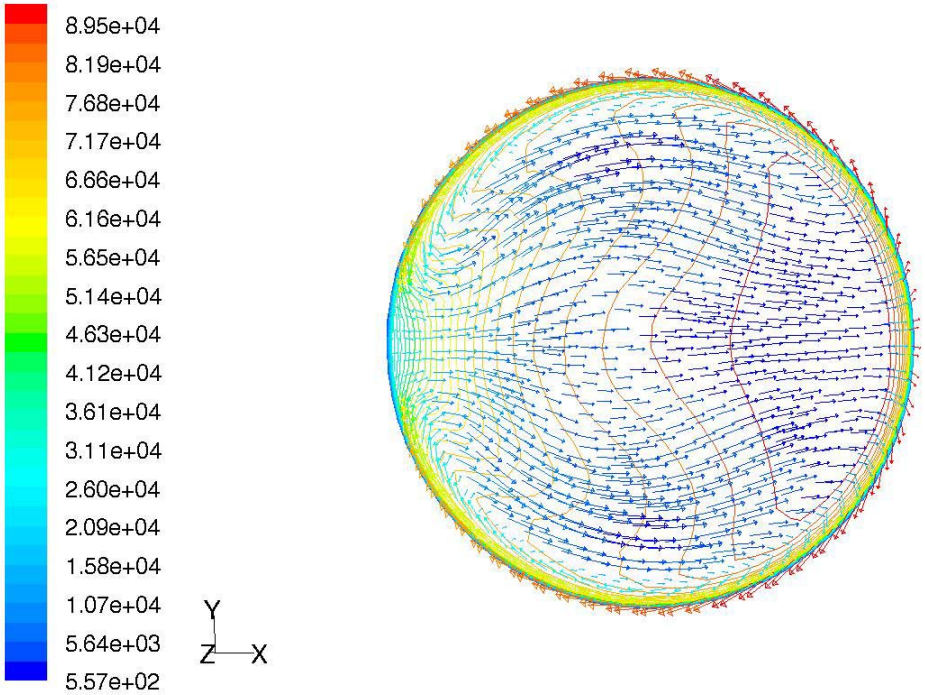


(a)

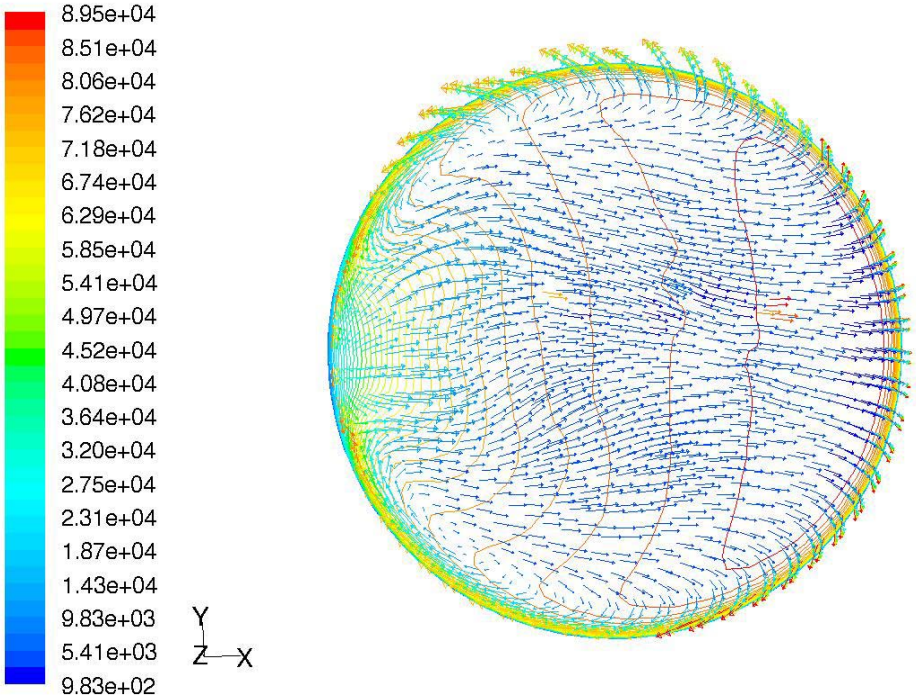


(b)

Figure 36. Velocity Contours Superimposed on Vorticity Vectors at 135°



(a)



(b)

Figure 37. Velocity Contours Superimposed on Vorticity Vectors at 180°

Helicity

Helicity[64], [65], [66], [67], [68] has been a subject of active research with regard to its possible connection with fluid turbulence. The scalar helicity ($H = \left(\nabla \times \vec{V}\right) \cdot \vec{V}$) is defined by the dot product of vorticity and the velocity vector. When the vorticity and velocity vectors are perpendicular, their scalar product is nil. Negative value ascribed to helicity means a clockwise orientation of vorticity relative to velocity and counterclockwise for positive helicity. Thus, helicity provides insight into the vorticity alignment with the fluid stream. Helicity is a useful indicator of the orientation of the velocity vector field with respect to vorticity.

Figure 38 is a plot of helicity magnitude for cross sections located at 0°, 90°, and 180°. Helicity signatures differ for the turbulence models. For the rke calculation, at the entrance to the bend the onset of flow oscillation begins left of center with counterclockwise rotation of vorticity relative to velocity and switches origination just before arriving at the center of the pipe. Counterclockwise rotation returns beyond the center and decays towards the wall. The 90° plane shows slight clockwise rotation at the inner wall. Just outside the boundary layer, counterclockwise rotation establishes itself until reaching the center of the pipe. Orientation switches just right of center over a short distance before changing back to counterclockwise rotation. This condition persists to the outer wall however; the magnitude of the helicity is slightly larger than at the inner wall. At the bend exit, similar oscillatory behavior is observed. However, the intensity is not as high as the 90° section.

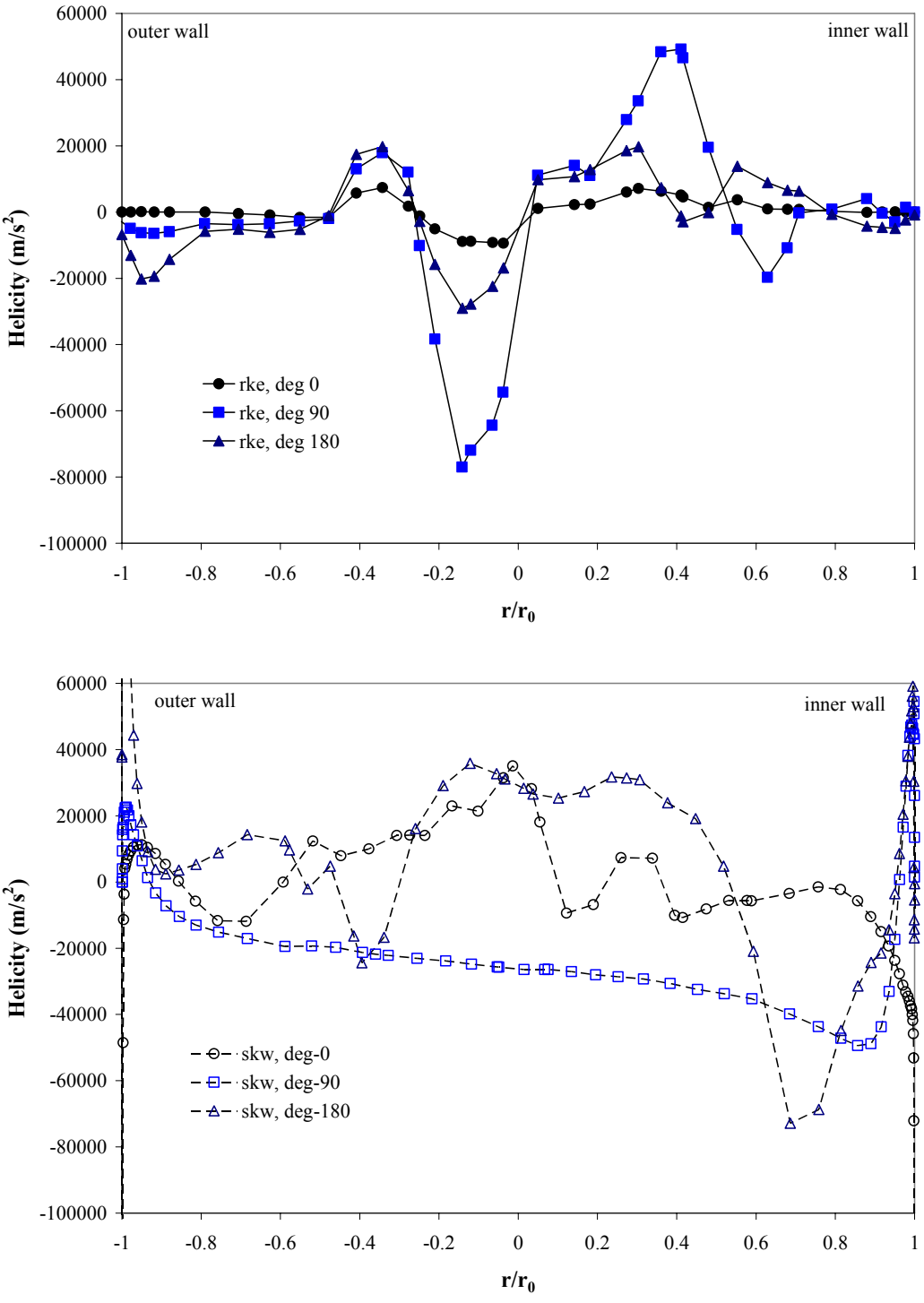


Figure 38. Helicity, Bend

Both Figures 39 and 40 are rke calculations. As seen, Figure 39 shows helicity in the bulk prior to entering the bend however, upon bend exit clockwise rotation of vorticity relative to velocity is more pronounced at the inner wall (Figure 40).

Overall, the standard k-ω calculation reveals for both bend inlet and outlet counterclockwise-oriented vorticity relative to velocity. With the exception of the near-wall region, clockwise orientation persists for the slice taken through the plant at 90°.

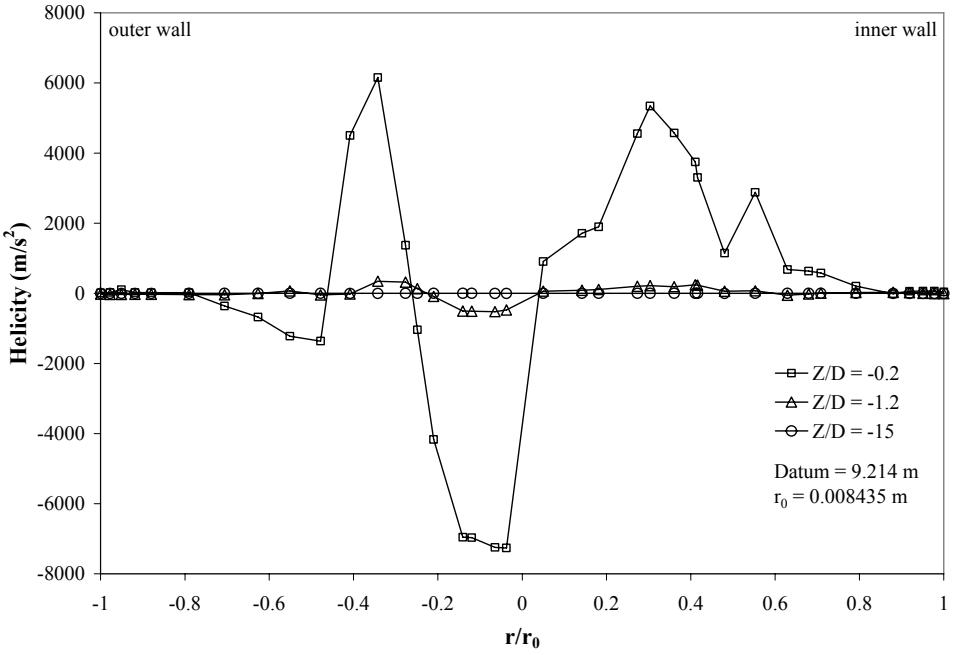


Figure 39. Helicity, Downstream (Inlet-Side) of Bend

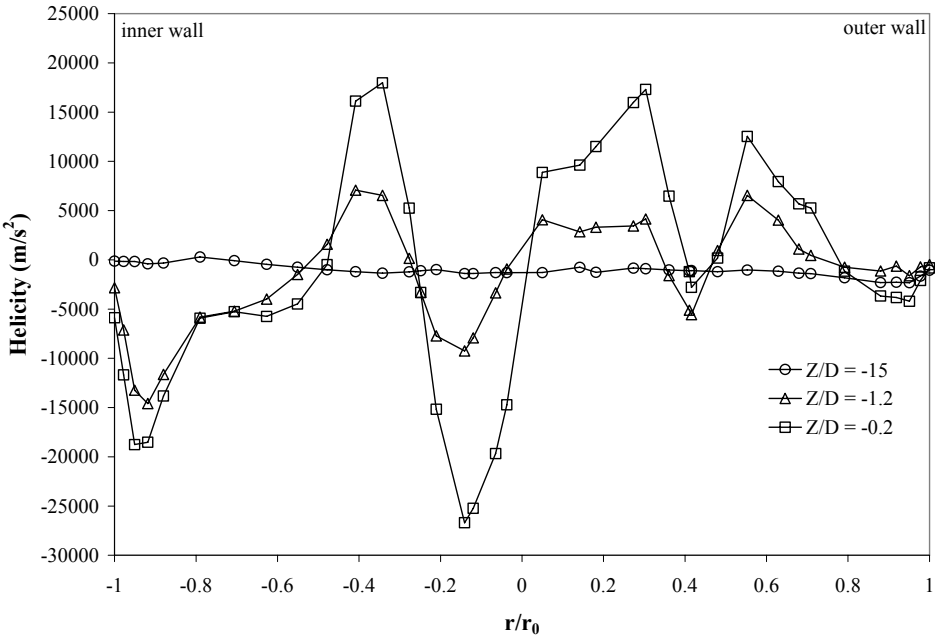


Figure 40. Helicity, Upstream (Outlet-Side) of Bend

Turbulence Intensity

Turbulence intensity ($I = \sqrt{\frac{2}{3}k} \Big/ U_{ref}$, in FLUENT) is the ratio of the magnitude of the root mean square (RMS) turbulent fluctuations to the reference velocity (U_{ref}). Two-thirds the turbulence kinetic energy (k) assumes isotropic turbulence and corresponds to averaging the three fluctuating components. The reference velocity specified in FLUENT's reference value panel was set as the mean velocity magnitude for the flow. Reference values (area, density, depth, enthalpy, length, pressure, temperature, velocity, viscosity, and ratio of specific heats) were set from conditions set on a particular boundary zone by selecting the zone from the compute from drop down list.

Figure 41 shows turbulence intensity significance in the near wall region relative to the bulk. The magnitude reflects nearly a two-fold difference between near wall region and the turbulence intensity in the bulk. Figure 42 displays the slight shift in the profile towards the inner wall over the short section just upstream of the bend entrance. Figure 43 displays the turbulence intensity distribution in the streamwise direction within the bend. Turbulence intensity at the entrance to the bend exhibit high value at the outer wall, low value at the inner wall. Even though depicted as the highest turbulence intensity at 90°, it actually occurs nearer to 45°. Figure 44, contours at the exit plane of the bend, show the two pairs of vortices in the pipe cross section. Rowe,[69] Azzola,[70] and Sudo[20] also observed such vortices. The orientation and locations differ in the two calculations perhaps due to the different near wall treatments applied in the numerical solution. Figure 45 shows that the bend influences downstream persists for some distance. At stations further downstream from the exit plane of the bend, as shown in Figure 46, turbulence intensity profiles eventually resemble the profiles upstream of the bend.

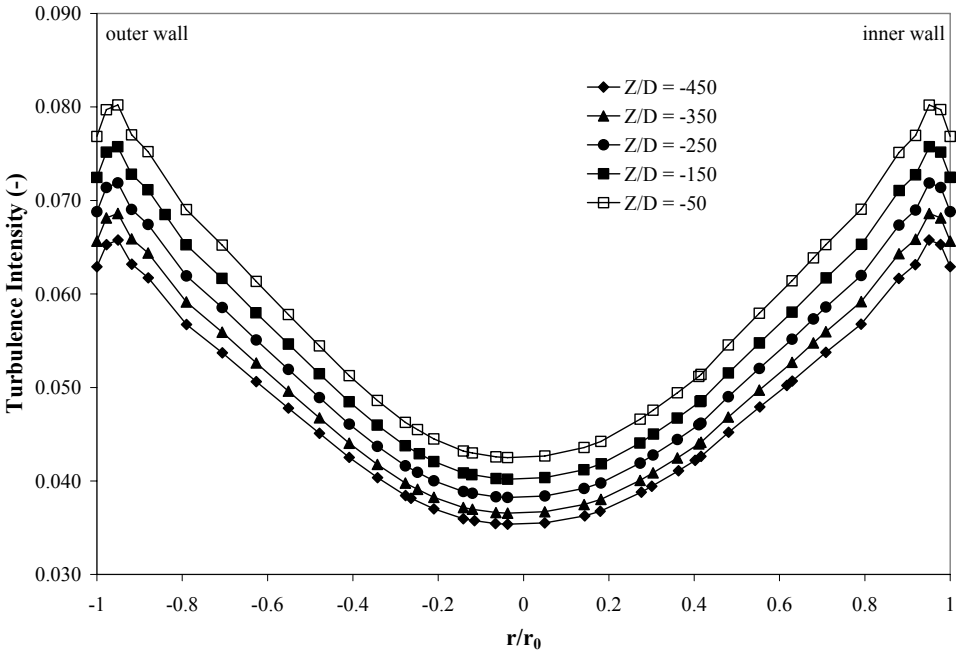


Figure 41. Turbulence Intensity, Inlet

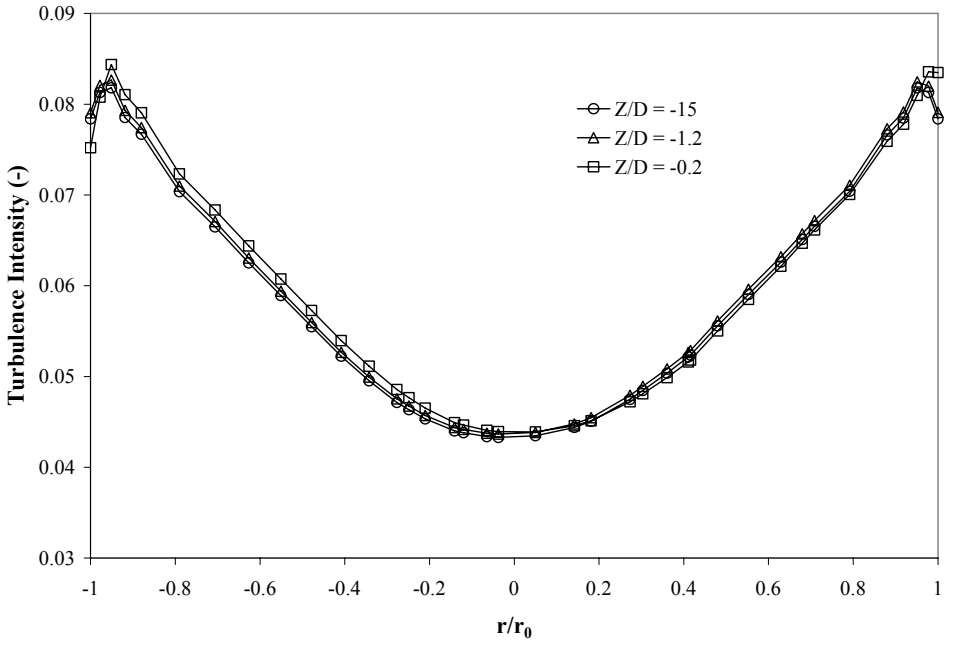


Figure 42. Turbulence Intensity, Downstream (Inlet-Side) of Bend

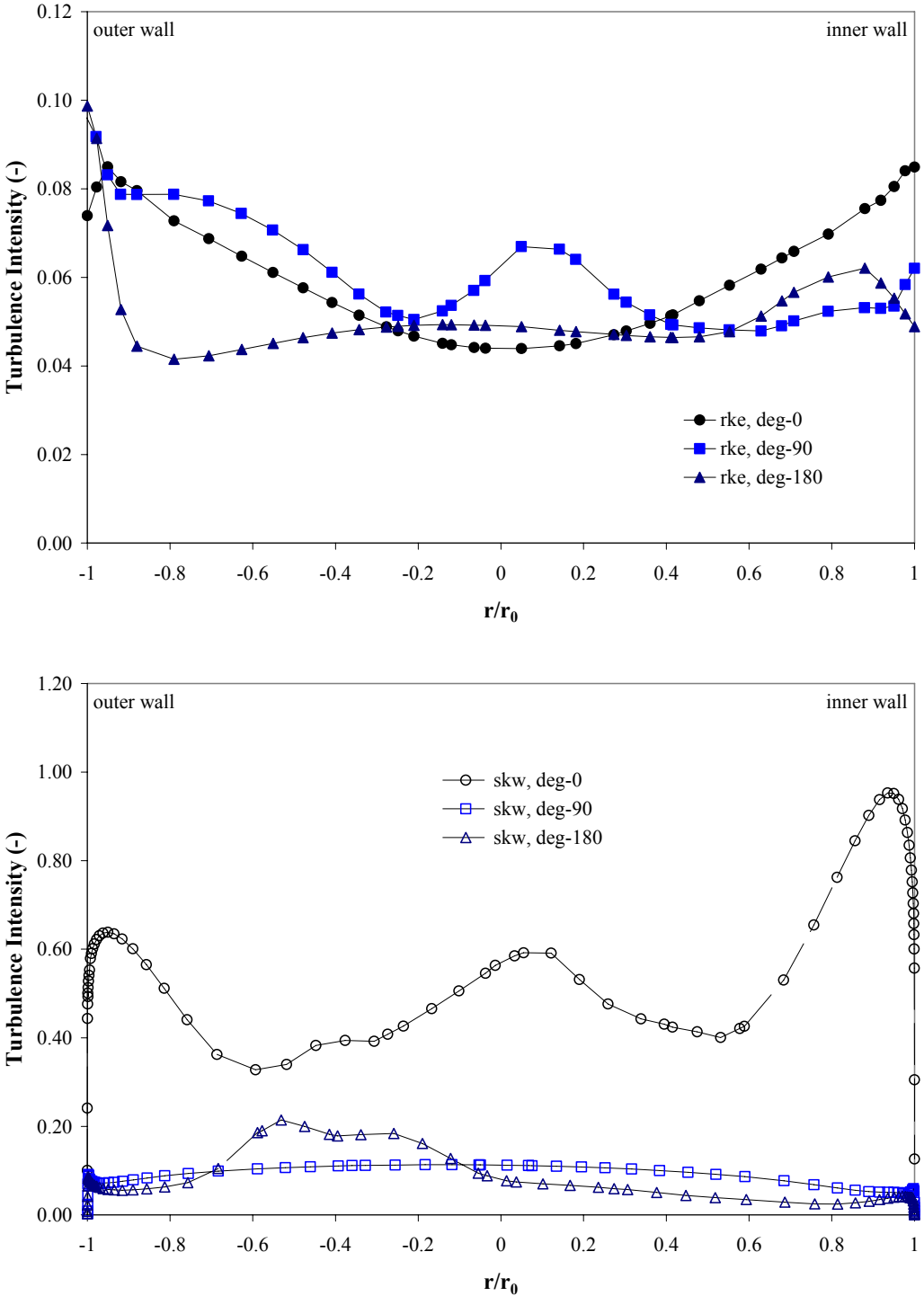


Figure 43. Turbulence Intensity, Bend

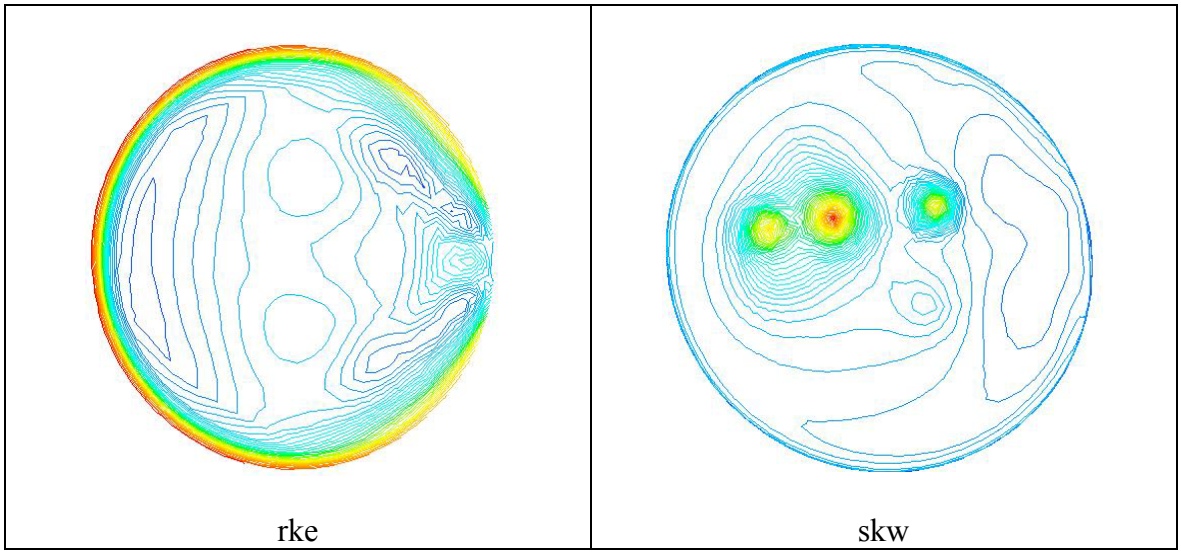


Figure 44. Turbulence Intensity at Bend Outlet (180°)

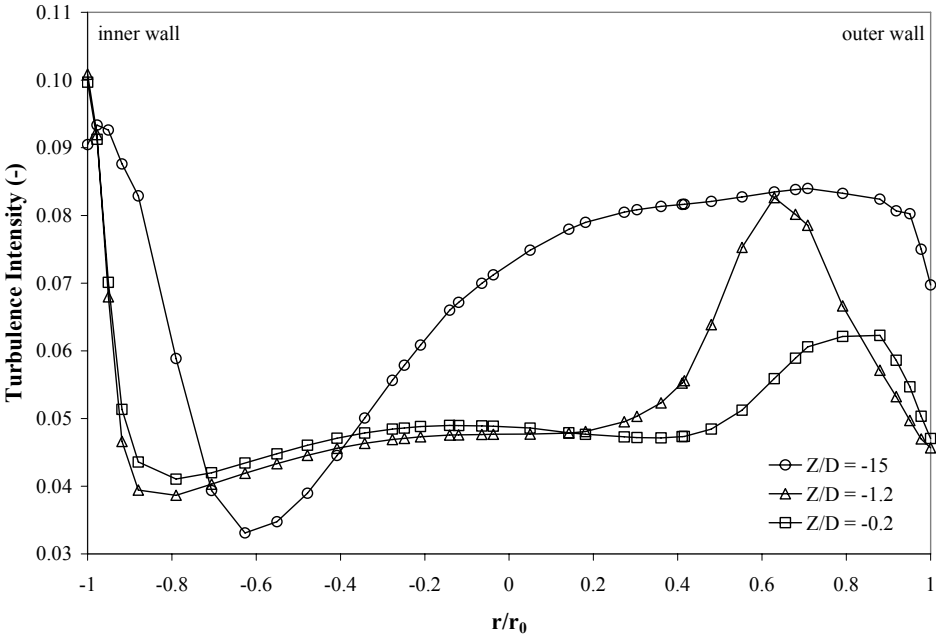


Figure 45. Turbulence Intensity, Upstream (Outlet-Side) of Bend

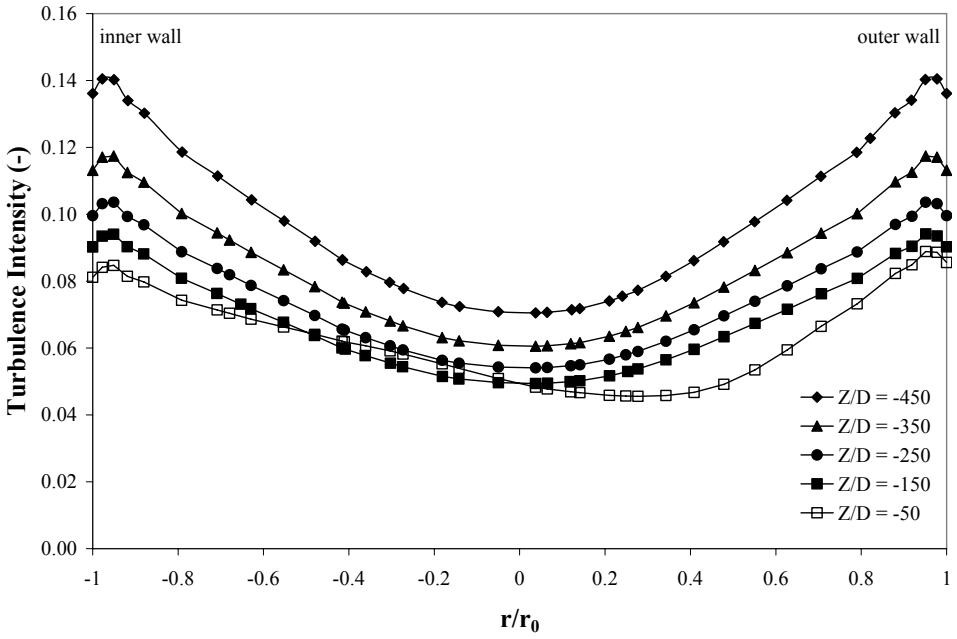


Figure 46. Turbulence Intensity, Outlet

Summary of Flow Field Results

In the preceding paragraphs, we have shown that the streamlines in the longitudinal section upstream of the bend are straight. The bend however influences the upstream flow as flow approaches the entrance to the bend. There, the flow experiences a force that causes a deviation from motion in a straight line to motion along a curved path. In physics parlance, this is centripetal force. The expected response observed, is that flow accelerates inward toward the center of curvature. The acceleration of the fluid in this region causes a weak secondary-stream flowing from the outer to inner wall over the cross section. The realizable k-ε solution illustrates this behavior more prominently.

Figure 47 shows the increased fluid static pressure toward the outer wall at the entrance to the bend. The pressure gradient near the bend inlet causes the flow to

experience a centrifugal force, the inertial force reacting against a centripetal force. The velocity vector plot at 45° downstream of the entrance to the bend shows flow acceleration near the inner wall and flow deceleration at the outer wall. This action is attributable to the increasing pressure gradient. With centrifugal force, acting on the fluid and the large pressure difference the well-known vortices form in the cross section. Owing to the centrifugal force, fluid in the central part of the bend migrate outward while the pressure difference assists fluid near the walls to migrate inward along the wall. The realizable k-ε solution shows a more dramatic transport of fluid with high velocity near the inner wall toward the outer wall, transport induced by secondary flow. This secondary flow is responsible for forcing fluid near the upper and lower walls inward along the walls. Figure 47 gives evidence of the presence of strong secondary flow in the section prior 90°. Contour lines of the longitudinal velocity in the horizontal plane are in the shape of a tongue showing the high velocity region of the primary flow resident in the central part of the cross section. Beyond 90°, the secondary flow moves closer to the convex side of the bend. Even in a weaken state, the secondary flow still assists in the development of two pairs of vortices in the exit cross section. Also observed is the significant impact of flow behavior at the bend exit on the downstream flow conditions. Flow does however recover from perturbations caused by conditions at the exit of the bend.

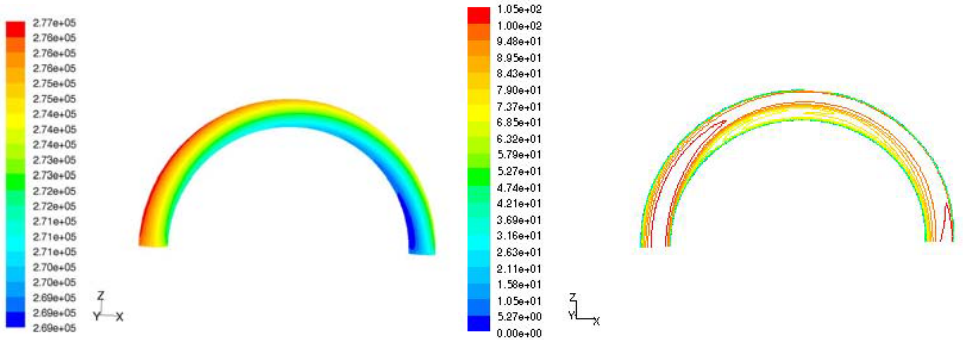


Figure 47. Contours of Static Pressure (Pa) on Bend Wall and Longitudinal Velocity (m/s) Profile

Particle Trajectory & Retention

A brief discussion of the inputs to the discrete phase model is offered prior to presentation of the numerical experiments' results. These results are compared against correlations as well as the ARTIST Phase I results. The last task performed consists of computation of the decontamination factor (DF) for the U-tube configuration for comparison against expert elicitation.

Discrete Phase Model: Settings & Options

Dispersed-phase volume fraction, and particulate loading are important for determining treatment of the dispersed phase. The dispersed-phase volume fraction (previously computed, 0.11%) is less than 10%. The particulate loading as previously computed is 1.026. Using these parameters, the average distance between the individual particles of the particulate phase, computed previously, is 7.91. Thus, for a gas-particle flow with a particulate loading of unity, interparticle space of about 8, the particle can be treated as isolated (i.e., very low particulate loading). For very low particulate loading,

the coupling between the phases is one-way (i.e., the fluid carrier influences the particles via drag and turbulence, but the particles have no influence on the fluid carrier). The discrete phase formulation used in FLUENT contains the assumption that the second phase is sufficiently dilute that particle-particle interactions and the effects of the particle volume fraction on the gas phase are negligible. Therefore having met all requirements, the discrete phase model (DPM), or "eddy lifetime" model is an appropriate choice for the problem at hand.

The Lagrangian discrete phase model in FLUENT solves the dispersed phase by tracking a large number of particles. The stochastic tracking (random walk) model includes the effect of instantaneous turbulent velocity fluctuations on the particle trajectories via stochastic methods. In this study, FLUENT's stochastic tracking (random walk) model predicted the dispersion of particles due to turbulence in the fluid phase. The discrete random walk (DRW) model with and without turbulent dispersion, and DRW modification to account for turbulent velocity fluctuations in the near-wall region represent the three variants used for computing particle trajectories. These particle trajectories facilitated the calculation of fractional deposition in the pipe configuration.

Time integration control of the particle trajectory equations in the discrete phase model required tracking parameter input (i.e., length scale/step length factor, maximum number of time steps). For advancing particles through the computational domain of 1.518×10^6 grid cells in the main flow direction required 7.59×10^6 steps (input for the maximum number of steps). The limit on the number of integration time steps

eliminates the possibility of particle capture in a recirculation region of the continuous phase flow field thus causing the code to attempt infinite tracking of the individual particle (akin to an endless loop). Additional DPM inputs include enabling physical models; spherical drag law and Saffman lift force (lift due to shear).

In the injection panel, initial conditions for surface injections required input of the point properties (velocity components, particle diameter, absolute temperature, and mass flow rate), particle type designation, material specification (inert and titanium, respectively), and diameter distribution (uniform interpolation). In addition, enabling/disabling turbulent dispersion is done in the injection properties panel. Enabling turbulent dispersion includes the effects of turbulence in the fluid phase on particle dispersion. Table 17 shows options enabled for the three variants.

Table 17. Discrete Phase Model Panel Input

	DRW, no dispersion	DRW with dispersion	Mod. stochastic model
Accuracy control	Enabled	Enabled	Disabled
Tolerance	1e-05	1e-05	-
Max. refinements	20	20	-
Automated tracking scheme selection	Enabled	Enabled	-
Tracking scheme	-	-	Trapezoidal
User defined functions (UDF)	None	None	Scalar update DPM time step
Number of scalars	-	-	11

The trap boundary condition was set for all walls initially. The trap boundary condition terminates the trajectory calculation when the particle strikes the wall. Reflect boundary condition rebounds particle off the boundary upon contact depending of the coefficient of restitution. A normal or tangential coefficient of restitution equal to unity

implies that the particle retains all of its normal or tangential momentum after the rebound (an elastic collision). The converse holds for a setting of zero for the coefficient of restitution, inelastic collision.

Comparison of Numerical Results with Correlations

We now turn our attention to the comparison of aerosol deposition obtained from numeric results using FLUENT and correlations for both the straight section and the bend. The Liu and Agarwal[71] correlation expresses turbulent inertial deposition in the straight section ($\eta = 1 - \eta_{transport} = 1 - \exp(-\pi d L V_t / Q)$) in terms of the turbulent inertial deposition velocity (V_t). Turbulent inertial deposition velocity[39] depends on the value of the dimensionless particle relaxation time ($\tau_+ = 0.0395 Stk \, \mathrm{Re}_d^{0.75}$). Equation 140 shows for the dimensionless particle relaxation time the appropriate formulation for turbulent inertial deposition velocity.

$$V_t = \begin{cases} 1.1928 \times 10^{-4} u \, \mathrm{Re}_d^{-0.125} \tau_+^2 & \tau_+ \le 12.9 \\ 1.9881 \times 10^{-2} u \, \mathrm{Re}_d^{-0.125} & \tau_+ > 12.9 \end{cases} \quad \textbf{(140)}$$

The dimensionless particle relaxation times for the problem at hand exceed 12.9 for all particle sizes considered. Using inlet velocity (71 m/s) and volumetric flow rate (2.052 x 10^5 m^3/s) for the 9.345 meters pipe (0.01687 m internal diameter), $V_t \approx 0.295$, results in a computed deposition efficiency of zero. Likewise, using velocity (242 m/s) and volumetric flow rate (7.02 x 10^5 m^3/s) for the 9.345 meters of 0.01687 m internal diameter pipe, $V_t \approx 1.004$, again deposition efficiency ≈ 0. Therefore, the Liu and Agarwal correlation predicts no deposition in the 9.345 meters of straight section for all

particle sizes considered. As mentioned previously, control of deposition on walls in general and deposition on walls of the straight section in particular largely depends on boundary condition selection.

Figure 48 shows plots of the three correlations found in the literature for expressing inertial deposition in bends. All relate deposition as a function of Stokes number. Pui and others offer a simple expression ($\eta = 1 - 10^{-0.963Stk}$) that is solely a function of Stokes number. Brockmann[30] supplied a correlation based on data describing transport efficiency of particles through a bend in turbulent flow. The Brockmann correlation includes in addition to the Stokes number dependence, the angle of the bend in radians (θ), $\eta_{transport} = \exp(-2.823Stk\theta)$. Deposition efficiency would therefore be $\eta = 1 - \eta_{transport}$. Equation 141, McFarland's correlation (based on three-dimensional flow field particle tracking) is a function of bend angle (θ, in radians), penetration (P, in percent), and coefficients (a, b, c, and d) that are a function of the curvature ratio (δ, defined as the radius of curvature of the bend (R) divided by the internal radius of the tube).

$$\ln P = \frac{4.61 + a\theta Stk}{1 + b\theta Stk + c\theta Stk^2 + d\theta^2 Stk} \qquad \textbf{(141)}$$

$$a = -0.9526 - 0.05686\delta \qquad b = \frac{-0.297 - 0.0174\delta}{1 - 0.07\delta + 0.0171\delta^2}$$

$$c = -0.306 + \frac{1.895}{\sqrt{\delta}} - \frac{2}{\delta} \qquad d = \frac{0.131 - 0.132\delta + 0.000383\delta^2}{1 - 0.129\delta + 0.0136\delta^2}$$

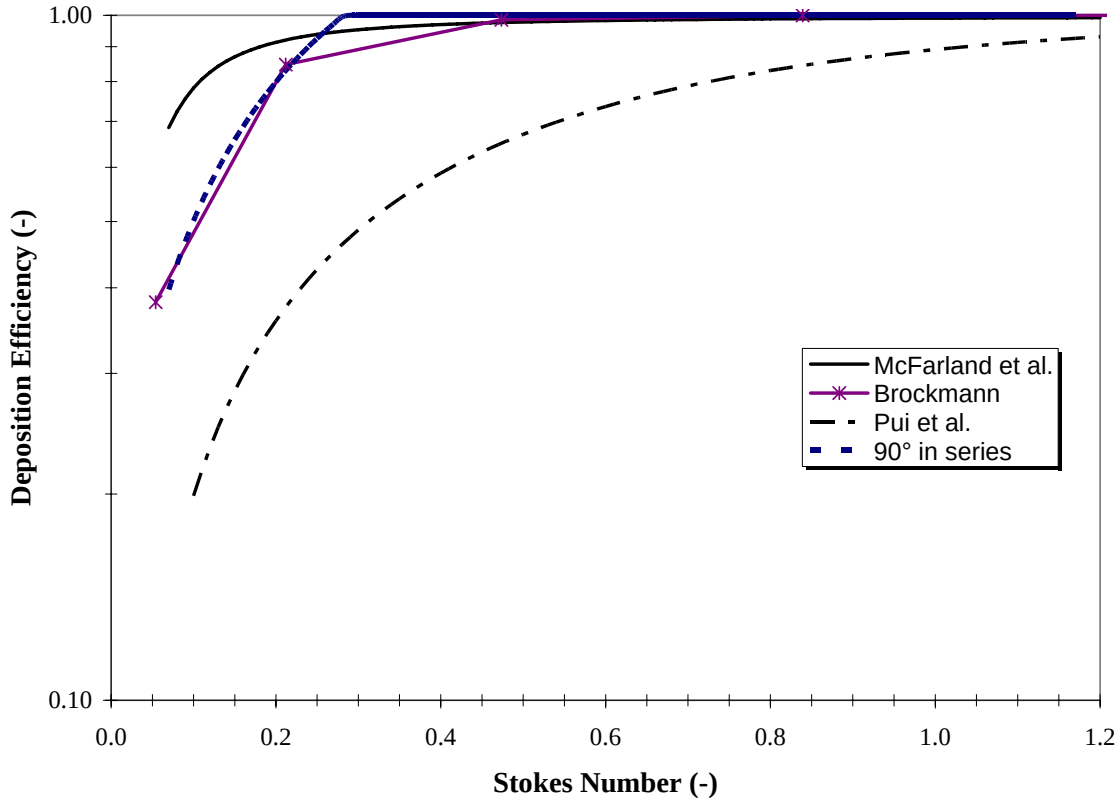


Figure 48. Deposition Efficiency Correlations

McFarland and others advanced the argument that penetration of aerosol through 180° bends should not be treated as (often is) the product of the penetration of two 90° bends in series. Figure 48 shows curves for each correlation using a log-linear scale. Pui and others' correlation was used to represent the 180° bend by computing the product of two 90° bends in series. One immediately observes that for Stokes numbers beyond 0.3, application of two 90° bends in series to represent a bend of 180° yields deposition efficiencies in excess of unity. Therefore, the plot reflects a value of unity for those deposition efficiency values. Further comparison of the curves reveals little disparity in predictions involving the Pui et al. correlation as compared to the

Brockmann correlation for Stokes numbers less than 0.2. For Stokes numbers less than 0.25 the McFarland correlation predicts deposition in excess of either correlation. However, all deposition efficiency correlations considered agree beyond a Stokes number of 0.5. Thus, the expectation is that most particles deposit on the walls of the bend if the Stokes numbers exceeds 0.5.

Table 18 lists Stokes numbers and the corresponding aerodynamic diameter specific to the current problem. It also shows that particles larger than 3-micron have an extremely low probability of penetration (Stk > 0.5). As implied above, depending on the correlation used, the estimate of deposition for very small Stokes numbers will be considerably different.

Figure 49 displays deposition efficiency curves comparing the Brockmann and McFarland correlations with results for three cases.

- Turbulent dispersion off, trap boundary conditions applied to straight section walls;
- Turbulent dispersion on, reflect boundary conditions applied to straight section walls; and
- UDF with reflect boundary conditions applied to straight section walls.

For bend walls, trap boundary conditions were applied in all cases. Figure 49 confirms that particles with Stokes numbers above 0.3 have a high probability of depositing in the bend.

Table 18. Stokes Number for ARTIST

aerodynamic particle diameter (μm)	Knudsen number (Kn) $2\lambda/d_p$	Cunningham slip correction factor (C_c) $1+Kn[\alpha + \beta \exp(-\gamma/Kn)]$	particle relaxation time $\tau = \rho_p(d_p)^2C_c/18\nu$	$Stk = \tau U/d$
1.00	0.03748932	1.04281	1.29E-05	0.1882
2.00	0.01874466	1.02141	5.04E-05	0.2120
3.00	0.01249644	1.01427	1.13E-04	0.4736
4.00	0.00937233	1.01070	1.99E-04	0.8391
5.00	0.00749786	1.00856	3.11E-04	1.31
6.00	0.00624822	1.00714	4.47E-04	1.88
7.00	0.00535562	1.00612	6.08E-04	2.56
8.00	0.00468616	1.00535	7.93E-04	3.34
9.00	0.00416548	1.00476	1.00E-03	4.22
10.00	0.00374893	1.00428	1.24E-03	5.21

Mean free path (λ) is the average distance traveled by a gas molecule between collisions with other molecules. The formula for λ was taken from Sime, 1990 [Eq. 12-11]; Atkins, 1994 [1-19]. $\lambda = kT/[(2)^{0.5}p\pi d^2]$

[72]For solid particles, $\alpha = 1.142$, $\beta = 0.558$, and $\gamma = 0.999$

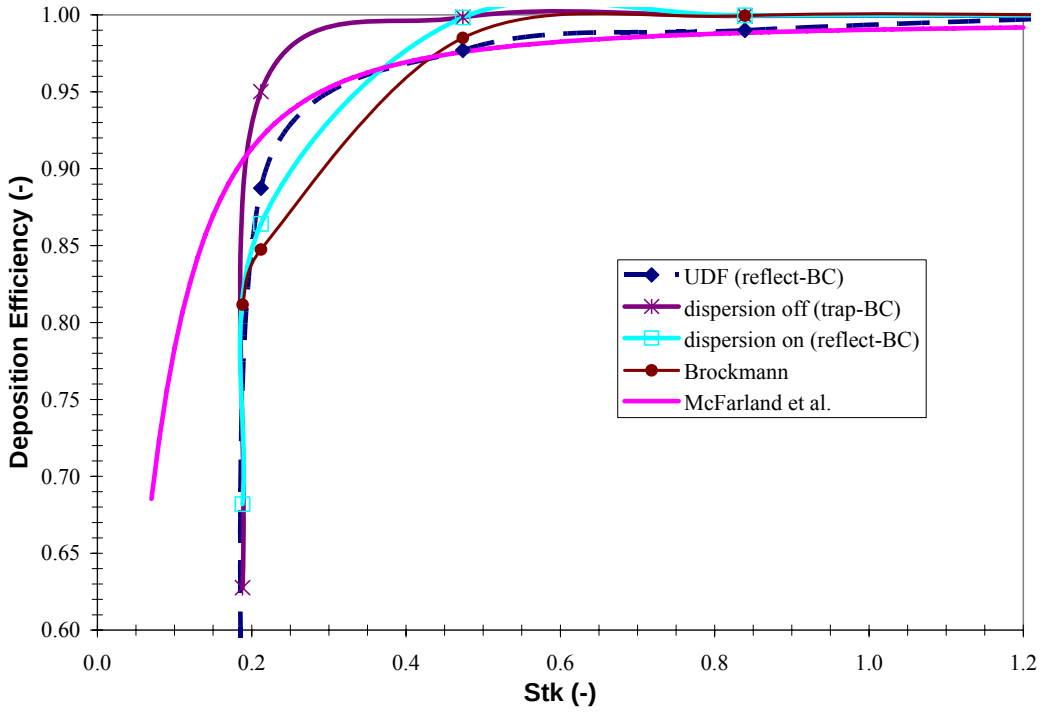


Figure 49. Deposition Efficiency: Computation versus Correlation

We also observe in Figure 49 that the Brockmann correlation gives the most conservative estimate of deposition for Stokes numbers less than about 0.425; beyond this point, the McFarland correlation predicts the most conservative estimate. Also shown is that the curves for these two correlations bound the computational results for the reflect boundary condition using the UDF. With dispersion enabled and using reflect boundary conditions this modeling approach performs relatively well over the entire range of Stokes numbers. However, for the latter condition, one must bear in mind that the boundary layer physics is incorrect. Disabling the dispersion model and permitting all particles that make contact with the wall yields the worst prediction, particularly for smaller particles. Regardless of the modeling technique or correlation employed, beyond a Stokes number of about 0.5 the result is the same, the majority of particles plate out to the walls. For this analysis, these Stokes numbers correspond to particles larger than about three micron. A concurrent observation is that, all calculations and correlations agree that smaller particles will deposit, the disagreement is on the amount of deposition that occurs (Table 19).

Table 19. Deposition Efficiency ($0.2 \leq Stk \leq 0.5$)

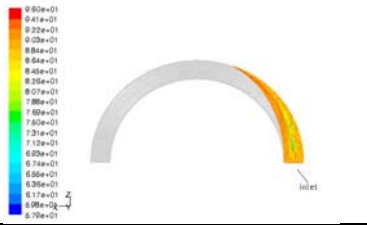
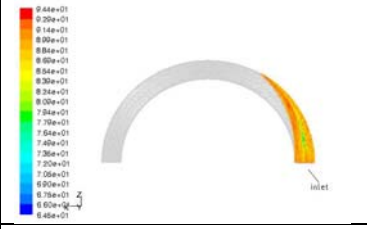
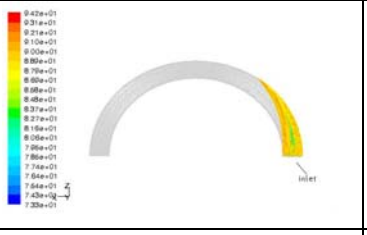
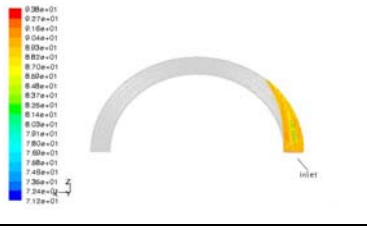
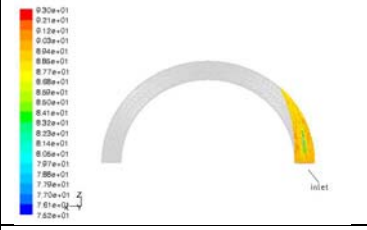
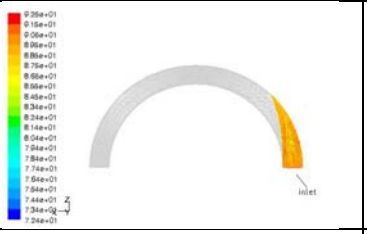
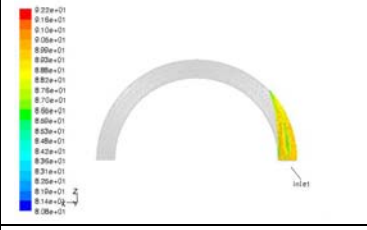
Stokes Number (Stk)	0.2	0.3	0.4	0.5
Brockmann	0.81	0.905	0.96	0.99
McFarland	0.91	0.955	0.97	0.979
UDF, reflect BC	0.835	0.95	0.96	0.98
Dispersion off, trap BC	0.63	0.99	0.997	1
Dispersion on, reflect BC	0.68	0.93	0.978	1

Comparison of Numerical Results with Experiment

Experimental data from the single U-tube test in the ARTIST project showed aerosols entered the test section having an aerodynamic mass median diameter (AMMD) of about 5.0 μm, whereas the aerosols leaving had an AMMD of 1.05 μm. The numerical simulation of this test done using FLUENT (discussed above) showed only smaller particles penetrates the bend. In fact, particles larger than three micron (3 μm) did not penetrate the bend with any regularity. In the results of the steady state solution shown in Table 20 (RHS, inlet), we see that particles larger than about six micron (6 μm) accumulate on walls of the first 45-degree section of the bend. Four-micron (4 μm) particles barely negotiate beyond the 90-degree section while five-micron (5 μm) particles lack sufficient momentum to advance to the 90-degree mark. The aforementioned correlations support observed behavior predicted for the specified conditions.

Interestingly, a conjecture arrived at from the computational results is that there is apparent particle accumulation within the entrance section of the curved pipe. Although it cannot be demonstrated with the current approach, it is believed that at some point in the experiment these particles are flushed from this location. As demonstrated in the illustration (Figure 50), this apparent flushing behavior was observed by the researchers at PSI.

Table 20. Particle Retention in the Bend

		
1 micron	2 micron	3 micron
		
4 micron	5 micron	6 micron
		
7 micron	8 micron	9 micron
		
10 micron	11 micron	12 micron

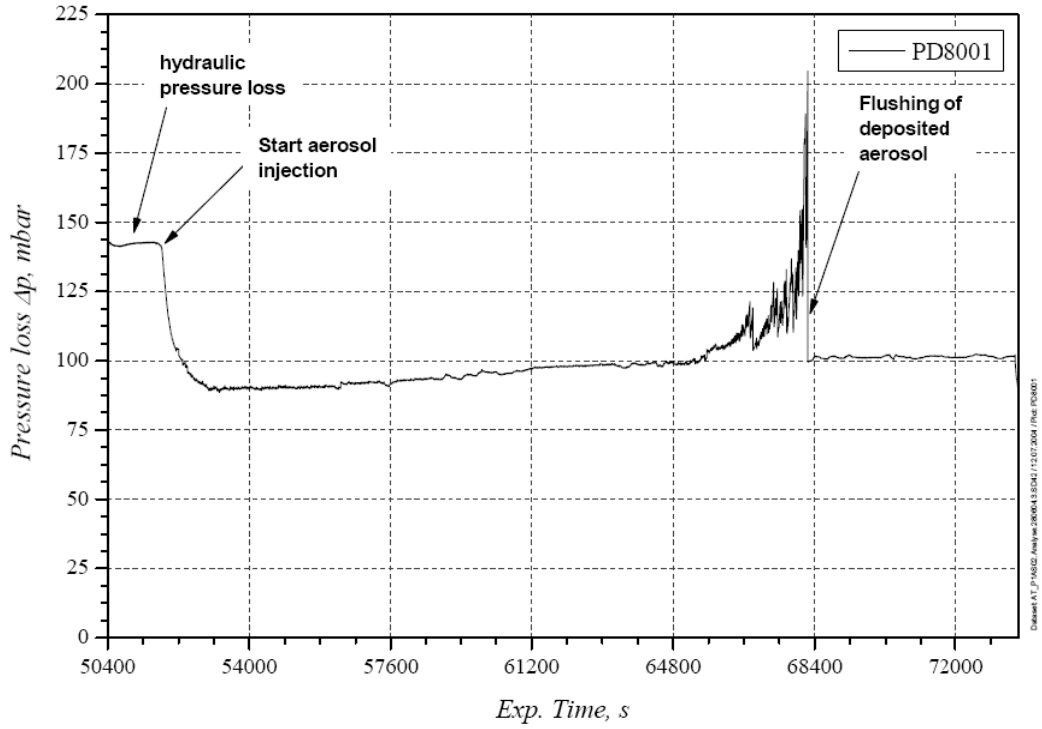


Figure 50. Pressure Drop Across 600 mm of Pipe Including the Bend

Decontamination Factor

The overall objective of this analysis was to quantify the decontamination factor (*DF*) for a single U-shaped steam generator tube. The only theoretically based work, which one may use to estimate aerosol retention in a steam generator is due to Leaver.[73] Leaver developed a simple one-dimensional model based on the correlations of Liu and Agarwal,[66] and Douglas and Ilias.[74] The model predicts high decontamination factors, but obviously needs qualification due to the inherent uncertainty of extrapolating the correlations to SGTR conditions and three-dimensional geometry.

Equation 142 represents the decontamination factor expressed in terms of the deposition (retention) efficiency (E_D, the ratio of the particle mass deposited in the tube

to the particle mass entering the tube). Listed in Table 21 are some representative values for the decontamination factor.

$$DF = \frac{\text{entrant aerosol mass}}{\text{exit aerosol mass}} = \frac{1}{1 - E_D} \tag{142}$$

Table 21. Decontamination Factors

E_D	DF
0.0	1
0.5	2
0.9	10
0.99	100
0.999	1000
1.0	Undefined (1/0)

To compute the mass dependent decontamination factor requires conversion from count statistics to particle size statistics. Therefore, division of the entire particle size range into a series of contiguous particle size intervals constitutes the initial step in the summarization of the particle size distribution. In this grouped data procedure, the number of particles (the count) is ascribed to each interval. Equation 143 gives the count mean for grouped data and for a frequency function.

$$\overline{d_p} = \frac{\sum d}{N} = \frac{\sum n_i d_i}{\sum n_i} \tag{143}$$

n_i, the number of particles in group i have a midpoint of size d_i. The midpoint size d_i can be the geometric midpoint (the square root of the product of the upper and lower limits of the interval) or the arithmetic midpoint (the mean of the limits). The following

formulation utilizes the geometric midpoint. $N = \sum n_i$ (i.e., the total number of particles) is the summation over all intervals.

Equation 144 gives the aerodynamic mass median diameter[75] (AMMD).

$$AMMD = \exp\left(\frac{\sum nd^3 \ln d}{\sum nd^3}\right) \quad \textbf{(144)}$$

To compute the geometric standard deviation (GSD), the geometric mean diameter, d_g is needed (Equation 145).

$$d_g = \exp\left(\frac{\sum n_i \ln d_i}{N}\right) \quad \textbf{(145)}$$

The geometric standard deviation σ_g or GSD is computed from Equation 146.

$$\sigma_g = \exp\left(\sqrt{\frac{\sum n_i \left(\ln d_i - \ln d_g\right)^2}{N-1}}\right) \quad \textbf{(146)}$$

The AMMD (~ 5.4 μm), GSD (range from 1.6 to 3.2) and associated decontamination factor (3.6) computed from the aforementioned procedure displayed in Table 22 shows the primary difference is in the particle size spread or GSD. Independent of the boundary condition (i.e., trap or reflect) a relatively constant value for the decontamination factor exists for the bend. With the reflect boundary condition set for the wall, at system outlet, the AMMD (1.9 μm) and GSD (1.124 μm) corresponds to a decontamination factor of 4.43. With the trap boundary condition, at the outlet the AMMD (1.9 μm) and GSD (2.76 μm) corresponds to a decontamination factor of 4.46.

Table 22. Computed AMMD, GSD, and Decontamination Factor

	N_{total}	Precision (%)	AMMD (μm)	GSD (μm)	DF
[a] Dispersion off (rke, trap)	18624	0.73	5.43	3.22	3.56
[a] Dispersion on (rke, trap)	18624	0.73	5.68	2.43	3.86
[a] Dispersion on (rke, trap)	949822	0.10	5.44	3.23	3.57
[a] Dispersion on (rke, reflect)	952928	0.10	5.43	1.61	3.56
[a] UDF (skw, trap)	41002	0.49	5.43	1.61	3.56
[a] UDF (rke, trap)	199512	0.22	5.42	2.64	3.56
[b] Dispersion on (rke, reflect)	952928	0.10	1.9	1.12	4.43
[b] Dispersion on (rke, trap)	949822	0.10	1.9	2.76	4.46
[a] Bend [b] Outlet					

Accuracy/Precision

In simulations with one-way coupling, the restriction on the number of particles is imposed primarily from a statistical viewpoint. Since each particle is considered as one realization, it is required that the number of particles *N* be large enough to provide accurate statistics.[76] Treatment of each particle as representing a different realization of the underlying Lagrangian stochastic process justifies taking ensemble averages over all the *N* particles. Appeal to the central limit theorem shows that the statistical sampling error decreases as $N^{-1/2}$. Thus, 1% precision requires 10^4 particles (Figure 51). The third column in Table 22 lists the precision corresponding to each scenario.

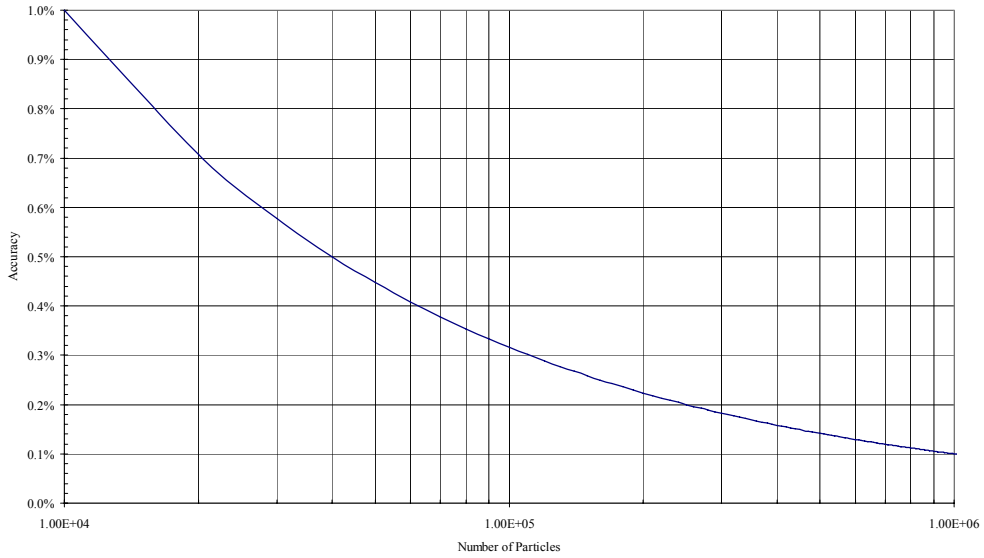


Figure 51. Accuracy as a Function of Number of Particles Sampled

CHAPTER V

CONCLUSIONS

Steam generator tubes have an important safety role in pressurized water reactor plants. They constitute one of the primary barriers between the radioactive and non-radioactive sides of the plant. For this reason, the integrity of the tubing is essential in minimizing the leakage of water between the two sides of the plant. Transients involving steam generator tube rupture (SGTR) events leading to containment bypass although infrequent are risk dominant. The high level of risk associated with this type of transient correlates with perceived consequences. In such an accident, the assumption is that a large factor of the gaseous radioactive aerosol-laden inventory vents directly to the outside environment. The current regulatory position permits no credit for retention of aerosol on the secondary side of the plant. The ambitious effort undertaken by the researcher at the Paul Scherre Institute (PSI) in Switzerland hopes to provide a quantitative measure of aerosol retention on the secondary side of a steam generator subject to a SGTR event. This study supports that effort by numerically examining and quantifying in-tube (primary side) aerosol retention.

Since this is a wall-bounded internal flow problem, only the interior of the pipe required modeling. The pre-processor GAMBIT, part of the FLUENT family of software, facilitated construction of the solid geometry and generation of the computational domain (mesh). The commercial code FLUENT 6.2 allowed evaluation of this aerosol-laden gas flow problem. The Eulerian approach using two turbulence models (rke, realizable k-ε and skw, standard k-ω) permitted analysis of the gas flow

field in the single U-shaped tube. The Lagrangian approach employing the discrete random walk model for particle dispersion permitted examination of deposition in the U-tube. For the flow conditions analyzed, the dominant removal mechanism is inertial deposition. Inertial deposition due to turbulent fluctuations is probably the least understood and the most difficult mechanism to model and predict. Further complicating the situation was the requirement to treat the gas as compressible. Additionally, the need to capture physics in the immediate vicinity of walls required application of a measure to compensate for the deficiencies in the eddy impaction model (EIM) that arise when used to predict anisotropic turbulent flows.

Introduced in this study is a user-defined function (UDF) that modifies the stochastic dispersed phase model (DPM) in the commercial computational fluid dynamics (CFD) code FLUENT. The algorithm used changes the Lagrangian time scale to a wall function. The procedure involves modification of the root-mean-square (RMS) fluctuating velocity values to account for the strong anisotropic nature of the flow in the immediate vicinity of the wall (boundary layer). Using the mean fluid velocity vector; the unit vector normal to the face of the nearest wall; and the spanwise unit vector permitted acquisition of the streamwise, normal, and spanwise unit vectors in the particle local coordinate system. The fluctuating velocity in the local coordinate system during the eddy lifetime was computed by randomizing the components ($U' = \lambda_1\sqrt{\overline{U'^2}}$, $V' = \lambda_2\sqrt{\overline{V'^2}}$, and $W' = \lambda_3\sqrt{\overline{W'^2}}$). The last step involved transforming the fluctuating velocity vector back into the computational coordinate frame (e.g., Cartesian).

Examination of the computationally derived flow field results revealed flow features consistent with those observed by previous investigators. In the bend, as observed in previous investigations:

a) the onset of a weak secondary flow accelerated flow at entrance to the bend towards the inner wall;
b) a flow separation zone develops on the convex wall that persists from the point of onset;
c) centrifugal force concentrates high velocity flow in the direction of the concave wall;
d) formation of vortices throughout the flow domain result from rotational (Dean-type) flow;
e) the weaken secondary flow assists in the formation of two twin vortices in the outflow cross section; and
f) perturbations induced by the bend influence flow recovery several pipe diameters upstream of the bend.

The Liu and Agarwal correlation indicated no deposition occurs in the straight section. Setting the boundary conditions to satisfy this criterion showed good agreement with expected behavior. With adherence to the no deposition in the straight section criterion, a consistent decontamination factor was computed (DF ≈ 3.6 and DF ≈ 4.4) for the bend and outlet, respectively. The numerical results compared favorably with the two correlations for prediction of deposition in bends from turbulent flow. Correcting for anisotropic turbulent velocity fluctuations in the near-wall region provided a slightly improved decontaminator factor prediction. The numerically derived decontamination

factors in this report are in agreement with those predicted by the NUREG-1150 expert elicitation panel.

Inclusion of the adjustment for spurious drift in the UDF should provide further improvement in the decontamination factor prediction. The modification to the algorithm discussed in the section on mathematical methods has yet to be incorporated in the FLUENT code. Therefore, it was not possible to investigate the contribution of spurious drift in this analysis.

The plan for the next series of tests in the ARTIST experiment is to use submicron-sized particles. The issue of spurious drift must be addressed when submicron-sized particles are introduced in turbulent flow situations and the proposed algorithm will be useful in the prediction of deposition. In addition, the FLUENT code will require further modification to allow implementation and thus calculation of Brownian motion under turbulent flow conditions when submicron-sized particles are used. At present, the Brownian motion option is only available for laminar flows.

REFERENCES

[1] D.A. Powers, Sandia National Laboratories (Private Communication, Nov. 2006).

[2] OECD, Nuclear Energy Agency, "Occupational Exposure and Steam Generator Replacements," ISO European Technical Center - CEPN Information Sheet No. 1, May 1994.

[3] "Steam Generator Cracking Woes Multiplying, NRC Report Says," *Nucleonics Week*, July 15, 1993, pp. 6-7.

[4] P.E. MacDonald, V.N. Shah, L.W. Ward, and P.G. Ellison, "Steam Generator Tube Failures," NUREG/CR-6365, Idaho National Engineering Laboratory, INEL-95/0383, (1996).

[5] H. Alter, *Glossary of Terms in Nuclear Science and Technology*, American Nuclear Society, Standards Committee ANS-9, La Grange, IL, (1986).

[6] Nuclear Regulatory Commission, Office of Nuclear Regulatory Research, "Severe Accident Risks: An Assessment for Five U.S. Nuclear Power Plants – Final Summary Report," NUREG-1150, (1991).

[7] Nuclear Regulatory Commission, Division of Systems Technology, "Individual Plant Examination (IPE) Program: Perspectives on Reactor Safety and Plant Performance, Part 1: Final Summary Report, Vol. 1," NUREG-1560, (1996).

[8] A. Auvinen, J.K. Jokiniemi, A. Lähde, T. Routamo, P. Lundström, H. Tuomisto, J. Dienstbier, S. Güntay, D. Suckow, A. Dehbi, M. Slootman, L. Herranz, V. Peyres, and J. Polo, "Steam Generator Tube Rupture (SGTR) Scenarios," *Nucl. Eng. Design*, **235**, 457, (2005).

[9] A.D. Gosman and E. Ioannides, "Aspects of Computer Simulation of Liquid-Fuelled Combustors," *J. Energy*, **7**, 482 (1981).

[10] V. Mehrotra, G.D. Silcox, and P.J. Smith, "Numerical Simulation of Turbulent Particle Dispersion Using a Monte Carlo Approach," *Proceedings of FEDSM' 98, 1998 ASME Fluids Engineering Division Summer Meeting*, June 21 – 25, 1998, Washington, DC (FEDSM98-5036).

[11] G. Kallio and M.W. Reeks, "A Numerical Simulation of Particle Deposition in Turbulent Boundary Layers," *Int. J. Multiphase Flow*, **15**, 433, (1989).

[12] H. Burnage and S. Moon, “Prediction of Material Particles in a Turbulent Flow,” *Compte-Rendus à l’Académie des Sci.*, **310**, Series II, 1595, (1990).

[13] L.P. Wang and D.E. Stock, “Stochastic Trajectory Models for Turbulent Diffusion: Monte Carlo Process versus Markov Chains,” *Atmos. Envir.*, **26A**, 1599, (1992).

[14] B.E. Launder, G.J. Reece, and W. Rodi, "Progress in the Development of a Reynolds-Stress Turbulence Closure," *J. Fluid Mech.*, **68**, 537 (1975).

[15] J. Kim, P. Moin, and R. Moser, "Turbulence Statistics in Fully Developed Channel Flow at Low Reynolds Number," *J. Fluid. Mech.*, **177**, 133 (1987).

[16] S.M. Chang, J.A. Humphrey, and A. Modavi, “Turbulent Flow in a Strongly Curved U-Bend and Downstream Tangent of Square Cross-Sections,” *Physico Hydrodyn.*, **4**, 243, (1983).

[17] M. Anwer, R.M.C. So, and Y.G. Lai, “Perturbation by and Recovery from Bend Curvature of a Fully Developed Turbulent Pipe Flow,” *Phys. Fluids A*, **8**, 1387, (1989).

[18] Y.G. Lai, R.M.C. So, and H.S. Zhang, “Turbulence-Driven Secondary Flows in a Curved Pipe,” *Theoret. Comput. Fluid Dynamics*, **3**, 163, (1991).

[19] G.C. Cheng and S. Farokhi, “On Turbulent Flows Dominated by Curvature Effects,” *ASME J. of Fluids Eng.*, **114**, 52, (1992).

[20] K. Sudo, M. Sumida, and H. Hibara, “Experimental Investigation on Turbulent Flow in a Circular-Sectioned 90-degree Bend,” *Experiments in Fluids*, **25**, 42, (1998).

[21] K. Sudo, M. Sumida, and H. Hibara, “Experimental Investigation on Turbulent Flow in a Circular-Sectioned 180-degree Bend,” *Experiments in Fluids*, **28**, 51, (2000).

[22] K. Sudo, M. Sumida, and H. Hibara, “Experimental Investigation on Turbulent Flow in a Square-Sectioned 90-degree Bend,” *Experiments in Fluids*, **30**, 246, (2001).

[23] H. Akilli, E.K. Levy, and B. Sahin, “Gas-Solid Flow Behavior in a Horizontal Pipe after a 90° Vertical-to-Horizontal Elbow,” *Powder Tech.*, **116**, 43, (2001).

[24] T.Y. Tu and C.A.J. Fletcher, “Numerical Computation of Turbulent Gas-Solid Flow in a 90° Bend,” *AlChE J.*, **41**, 2187, (1995).

[25] H. Arastoopour, "Numerical Simulation and Experimental Analysis of Gas/Solid flow Systems: 1999 Fluor-Daniel Plenary Lecture," *Powder Tech. J.*, **119**, 59, (2001).

[26] H. Bilirgen, E. Levy, and A. Yilmaz, "Prediction of Pneumatic Conveying Flow Phenomena Using Commercial CFD Software," *Powder Tech.*, **95,** 37, (1998).

[27] S. Benyahia, H. Arastoopour, T.M. Knowlton, and H. Massah, "Simulation of Particles and Gas Flow Behavior in the Riser Section of a Circulating Fluidized Bed Using the Kinetic Theory Approach for the Particulate Phase," *Powder Tech.*, **112**, 24, (2000).

[28] B.Y.H. Pui, and J.K. Agarwal, "Experimental Observation of Aerosol Deposition in Turbulent Flow," *Aerosol Sci. Tech.*, **5**, 145 (1974).

[29] A.R. McFarland, H. Gong, A. Muyshondt, W.B. Wente, and N.K. Anand, "Aerosol Deposition in Bends with Turbulent Flow," *Environ. Sci. Tech.*, **31**, 3371 (1997).

[30] J.E. Brockmann, Sampling and Transport of Aerosols, in *Aerosol Measurement Principles, Techniques, and Applications*, edited by K. Willeke and P.A. Baron, Van Nostrand Reinhold, New York (1993).

[31] S.B. Pope, *Turbulent Flows*, Cambridge University Press, p. 357 & 636, (2000).

[32] H. Schlichting, *Boundary-Layer Theory*, 7th ed., McGraw-Hill, New York, (1979).

[33] H. Schlichting, *Boundary-Layer Theory*, 6th ed., McGraw-Hill, New York, (1968).

[34] F.M. White, *Viscous Fluid Flow*, 3rd ed., McGraw-Hill, New York, (2006).

[35] J.O. Hinze, *Turbulence*, McGraw-Hill Publishing Co., New York, (1975).

[36] B.E. Launder and D.B. Spalding, *Lectures in Mathematical Models of Turbulence*, Academic Press, London, England, (1972).

[37] V. Yakhot and S.A. Orszag, "Renormalization Group Analysis of Turbulence: I. Basic Theory," *J. Sci Comput.*, **1**, 1 (1986).

[38] T.H. Shih, W.W. Liou, A. Shabbir, Z. Yang, and J. Zhu, "A New k-ε Eddy-Viscosity Model for High Reynolds Number Turbulent Flows - Model Development and Validation," *Comput. Fluids*, **24**, 227, (1995).

[39] D.C. Wilcox, *Turbulence Modeling for CFD*, DCW Industries, La Canada, CA, (1998).

[40] B.E. Launder and D.B. Spalding, "The Numerical Computation of Turbulent Flows," *Comp. Methods in Applied Mech. and Eng.*, **3**, 269, (1974).

[41] S.E. Kim and D. Choudhury, "A Near-Wall Treatment Using Wall Functions Sensitized to Pressure Gradient," *Separated and Complex Flows*, **217**, (1995).

[42] *Fluent 6.2, User's Guide*, Chapter 11 Modeling Turbulence, FLUENT, Inc., Lebanon, NH, (2005).

[43] M. Wolfstein, "The Velocity and Temperature Distribution of One-Dimensional Flow with Turbulence Augmentation and Pressure Gradient," *Int. J. Heat Mass Transfer*, **12**, 301, (1969).

[44] H.C. Chen and V.C. Patel, "Near-Wall Turbulence Models for Complex Flows Including Separation," *AIAA J.*, **26**, 641, (1988).

[45] S.A. Morsi and A.J. Alexander, "An Investigation of Particle Trajectories in Two-Phase Flow Systems," *J. Fluid Mech.*, **55**, 193, (1972).

[46] A. Haider and O. Levenspiel, "Drag Coefficient and Terminal Velocity of Spherical and Nonspherical Particles," *Powder Tech.*, **58**, 63, (1989).

[47] H. Ounis, G. Ahmadi, and J. B. McLaughlin, "Brownian Diffusion of Submicrometer Particles in the Viscous Sublayer," *J. Colloid and Interface Sci.*, **143**, 266, (1991).

[48] J.O. Hirschfelder, C.F. Curtiss, and R.B. Bird, *Molecular Theory of Gases and Liquids*, John Wiley & Sons, Inc., New York, (1954).

[49] R. Clift, J.R. Grace, and M.E. Weber, *Bubbles, Drops, and Particles*, Dover Publications, New York, (2005).

[50] A. Li and G. Ahmadi, "Dispersion and Deposition of Spherical Particles from Point Sources in a Turbulent Channel Flow," *Aerosol Sci. and Tech.*, **16**, 209, (1992).

[51] P.G. Saffman. "The Lift on a Small Sphere in a Slow Shear Flow," *J. Fluid Mech.*, **22**, 385, (1965).

[52] M. Shin and J.W. Lee, "Nonequilibrium Reynolds Stress for the Dispersed Phase of Solid Particles in Turbulent Flows," *Physics of Fluids*, **14**, 2898, (2002).

[53] G.A. Kallio and M.W. Reeks, "A Numerical Simulation of Particle Deposition in Turbulent Boundary Layers," *Int. J. Multiphase Flow*, **15**, 433, (1989).

[54] E.A. Matida, K. Nishino, and K. Torii, "Statistical Simulation of Particle Deposition on the Wall from Turbulent Dispersed Pipe Flow," *Int. J. Heat Fluid Flow*, **21**, 389, (2000).

[55] J.M. MacInnes and F.V. Bracco, "Stochastic Particle Dispersion Modeling and the Tracer Particle Limit," *Phy. Fluids A*, **4**, 2809, (1992).

[56] H.C. Strutt and M.F. Lightstone, "Analysis of Tracer Particle Migration in Inhomogeneous Turbulence," *Int. J. Heat and Mass Trans.*, **49**, 2557, (2006).

[57] T.L. Bocksell and E. Loth, "Random Walk Models for Particle Diffusion in Free-Shear Flows," *AIAA J.*, **29**, 1086, (2001).

[58] R.J.Garde, *Fluid Mechanics through Problems*, John Wiley & Sons, New York, (1989).

[59] R.D. Blevins, *Applied Fluid Dynamics Handbook*, Van Nostrand Reinhold Co., New York, (1984).

[60] S.A. Vasquez and V.A. Ivanov, "A Phase Coupled Method for Solving Multiphase Problems on Unstructured Meshes," in *Proceedings of ASME FEDSM'00: ASME 2000 Fluids Engineering Division Summer Meeting*, Boston, June 2000.

[61] C. Crowe, M. Sommerfeld, and Y. Tsuji, *Multiphase Flows with Droplets and Particles*, CRC Press, Baca Raton, (1998).

[62] E.W. Lemmon, A.P. Peskin, M.O. McLinden, and D.G. Friend, *Computer Code Properties of Pure Fluids – NIST Pure Fluids, Version 5.0*, NIST Standard Reference Database 12, (2000).

[63] G. Ahmadi, Turbulence (2005). http://www.clarkson.edu/fluidflow/courses/me637/T1_TURBULENCE.pdf.

[64] W. Polifke, "Aspects of Helicity in Turbulent Flows," Ph.D. dissertation, The City University of New York, (1990).

[65] H.K. Moffatt and A. Tsinober, "Helicity in Laminar and Turbulent Flow," *Ann. Rev. Fluid Mech.*, **24**, 281, (1992).

[66] Q. Chen, S. Chen, and G.L. Eyink, "The Joint Cascade of Energy and Helicity in Three-Dimensional Turbulence," *Phys. Fluids*, **15**, 361, (2003).

[67] S. Kuien, M.A. Taylor, and T. Matsumoto, "Cascade Time-Scales for Energy and Helicity in Homogeneous, Isotropic Turbulence," LA-UR 08-8194, (2004).

[68] L.T. Hoang, "Asymptotic Expansions of the Regular Solutions to the 3D Navier-Stokes Equations and Applications to the Analysis of the Helicity," Ph.D. dissertation, Texas A&M University, (2005).

[69] M. Rowe, "Measurements and Computations of Flow in Pipe Bends," *J. Fluid Mech.*, **43**, 771, (1970).

[70] J. Azzola, J.A.C. Humphrey, H. Iacovides, and B.E. Launder, "Developing Turbulent Flow in a U-bend of Circular Cross-Section: Measurement and Computation," *Trans. ASME J. Fluid Eng.*, **108**, 214, (1986).

[71] B.Y.H. Liu and J.K. Agarwal, "Experimental Observation of Aerosol Deposition in Turbulent Flow," *Aerosol Sci.*, **5**, 145, (1974).

[72] M.D. Allen and O.G. Raabe, "Slip Correction Measurements of Spherical Solid Aerosol Particles in an Improved Millikan Apparatus," *Aerosol Sci. Technol.*, **4**, 269, 1985.

[73] D. Leaver, J. Li, and R. Sher, "New Design Applications of Natural Aerosol Deposition in Nuclear Plant Accident Analysis," *Proceedings, 3rd OECD Specialist Meeting on Nuclear Aerosols in Reactor Safety*, Cologne, Germany (NEA/CSNI/R(1998)4).

[74] P. Douglas and S. Ilias, "On the Deposition of Aerosol Particles on Cylinders in Turbulent Cross Flow," *J. Aerosol Sci.*, **19**, 451, (1988).

[75] W.C. Hinds, *Aerosol Technology Properties, Behavior, and Measurement of Airborne Particles*, Second Edition, John Wiley & Sons, Inc., New York, (1999).

[76] P.K. Yeung and S.B. Pope, "An Algorithm for Tracking Fluid Particles in Numerical Simulations of Homogenous Turbulence," *J. Comp. Phys.*, **79**, 373, (1988).

APPENDIX A

stoch.c

```
/* The function do_stochastics shows an implementation of the standard stochastic
* tracking in FLUENT and an additional function for boundary layer treatment.
* Limitations:
* - Uses only spherical drag
* - tested for steady state simulations
* - does not work with accuracy control of particle tracks
* - stochastic tracking is activated for ALL defined injections
*
* The function do_wall_stoch also takes into account the effect of fluid
* boundary layer damping of turbulent fluctuations.
* Eddy lifetime is stored in scalar p->user[1]. The residence time of a particle
* in a given eddy is stored in p->user[0]
*
* Usage:
* - compile and load udfs
* - activate drag law
* - activate time step routines
* - for walls with reflection activate DPM boundary condition
* - activate 11 particle scalars and hook SCALAR_UPDATE function
* - hook DEFINE_INIT function
* - activate 11 UDM locations
* - after calc. of flow field execute the set_udm function for setting
* Y_plus values for particles in wall nearest cell (coarse grid), or
* if the sublayer is resolved choose set_udm_finegrid (may take a long time to execute)
* - finally perform particle tracking. The DPM_DRAG function evaluates the particle Y_plus
* and chooses the appropriate time scale for eddy interaction.
*
*
*
*/
#include "udf.h"
#include "dpm.h"
#include "cxsurf.h" /* for screen output with CX_Message(...) */
#include "surf.h"
#include "sg.h"
#define C_MU_FACTOR(I) (I->time_scale_constant)
#define C_MU 0.09
#define MY_WALL_SHEAR_X(f,t) F_STORAGE_R_N3V(f,t,SV_WALL_SHEAR)[0] /*
FORCES!! divide by face area */
#define MY_WALL_SHEAR_Y(f,t) F_STORAGE_R_N3V(f,t,SV_WALL_SHEAR)[1]
#define MY_WALL_SHEAR_Z(f,t) F_STORAGE_R_N3V(f,t,SV_WALL_SHEAR)[2]
#define THERMO 0
#define INTERPOLATION 1
#define DEBUGGER 0
cxboolean do_stochastics = TRUE;
cxboolean do_wall_stoch = FALSE;
cxboolean random_eddy_lifetime = FALSE;
cxboolean do_thermophoretic = TRUE;
real y_T = 11.8;
```

```
int update_interval = 100001;
int update_interval_up = 1000000;
int update_interval_old = 5000000;
static real therm_diff_factor = 100.;
static real yplus_thresh = 200.0;
static real dt_min = 1.0e-06;
static real te_min = 1.0e-05;
static int msg_count = 0;
/* global function declarations */
static real get_yplus_particle(cell_t c, Thread *t, real *pp);
static real get_tl_plus(real);
static void get_rms_vel_fluc(real,real,real*,real*,real*,real*,cphase_state_t*,cell_t c0, Thread
*t0);
static real half_zero(real,real);
enum udm_names{
Y_PLUS,
TAU_WALL,
N_VEC_X,
N_VEC_Y,
N_VEC_Z,
FACE_X,
FACE_Y,
FACE_Z,
TWALL,
QWALL,
WALL_HIT
};
enum p_names{
E_PLUS,
T_EDDY,
Y_PLUS_P,
LAMBDA1,
LAMBDA2,
LAMBDA3,
T_FORCE_X,
T_FORCE_Y,
T_FORCE_Z,
N_EDDY,
NEW_FORCE
};
/*********************/
real get_tl_plus(real yplus)
{
real TL_plus = 1.0;
#if 0
if (yplus>5.0 && yplus<yplus_thresh)
{
TL_plus = 7.122 + 0.5731*yplus - 0.00129*yplus*yplus;
}
else if (yplus <= 5.0)
{
```

```
TL_plus = 10.0;
}
#endif
TL_plus = 1./(4.529 + 0.0116*pow(yplus,1.75) + 0.768*sqrt(yplus) );
return TL_plus;
}
/*********************/
void get_rms_vel_fluc(real yplus, real vel_fric, real *Vprime_vec,
real *I, real *J, real *K,cphase_state_t* c,cell_t c0, Thread *t0)
{
real I_mag;
Vprime_vec[0] = 0.4*vel_fric*yplus/(1.+0.0239*pow(yplus,1.496));
Vprime_vec[1] = 0.0116*vel_fric*yplus*yplus/(1.+0.203*yplus+0.0014*pow(yplus,2.421));
Vprime_vec[2] = 0.19*vel_fric*yplus/(1.+0.0361*pow(yplus,1.322));
/* calc. anisoptropic local fluctuation velocities ...*/
/* and assign the local coordinate system vectors ... */
I[0] = c->V[0];
I[1] = c->V[1];
#if RP_3D
I[2] = c->V[2];
#endif
I_mag = NV_MAG(I);
NV_S(I,/=,I_mag); /* normalize */
J[0] = C_UDMI(c0,t0,N_VEC_X);
J[1] = C_UDMI(c0,t0,N_VEC_Y);
#if RP_3D
J[2] = C_UDMI(c0,t0,N_VEC_Z);
#endif
NV_CROSS(K,I,J); /* K = I x J */
#if DEBUGGER
CX_Message("I0 = %f, I1 = %f, I2 = %f\n",I[0],I[1],I[2]);
CX_Message("J0 = %f, J1 = %f, J2 = %f\n",J[0],J[1],J[2]);
CX_Message("K0 = %f, K1 = %f, K2 = %f\n",K[0],K[1],K[2]);
#endif
/* ... finally transform everything back into cartesian coordinates */
NV_S(I,*=,Vprime_vec[0]); /* x-component stored in I*/
NV_S(J,*=,Vprime_vec[1]); /* y-component stored in J*/
NV_S(K,*=,Vprime_vec[2]); /* z-component stored in K*/
/*re-assign fluctuation velocities*/
Vprime_vec[0] = I[0] + J[0] + K[0];
Vprime_vec[1] = I[1] + J[1] + K[1];
Vprime_vec[2] = I[2] + J[2] + K[2];
#if 0
Vprime_vec[0] = 0.*vel_fric;
Vprime_vec[1] = 0.*vel_fric;
Vprime_vec[2] = 0.*vel_fric;
#endif
}
/* function to determine a particle's dimensionless wall distance */
real get_yplus_particle(cell_t c,Thread *t,real *pp)
{
```

```
real Y_plus_p,Y_p;
real p_dist[ND_ND];
real A[ND_ND];
real Xf[ND_ND];
if(C_UDMI(c,t,Y_PLUS) < 0.0)
return -1.0;
A[0] = C_UDMI(c,t,N_VEC_X);
A[1] = C_UDMI(c,t,N_VEC_Y);
#if RP_3D
A[2] = C_UDMI(c,t,N_VEC_Z);
#endif
Xf[0] = C_UDMI(c,t,FACE_X);
Xf[1] = C_UDMI(c,t,FACE_Y);
#if RP_3D
Xf[2] = C_UDMI(c,t,FACE_Z);
#endif
NV_VV(p_dist,=,Xf,-,pp);
Y_p = fabs(NV_DOT(A,p_dist));
Y_plus_p = Y_p*sqrt(C_UDMI(c,t,TAU_WALL)*C_R(c,t))/C_MU_L(c,t);
return Y_plus_p;
}
/* distribute user RP variables */
DEFINE_ADJUST(get_rp_variables,domain)
{
#if !RP_NODE
therm_diff_factor = RP_Get_Real("therm_diff");
update_interval = (int)RP_Get_Real("update_thermo");
update_interval_old = (int)RP_Get_Real("update_old");
update_interval_up = (int)RP_Get_Real("update_new");
yplus_thresh = RP_Get_Real("yplus_threshold");
dt_min = RP_Get_Real("delta_t_min");
te_min = RP_Get_Real("t_eddy_min");
#endif
host_to_node_real_1(therm_diff_factor);
host_to_node_int_1(update_interval);
host_to_node_int_1(update_interval_old);
host_to_node_int_1(update_interval_up);
host_to_node_real_1(yplus_thresh);
host_to_node_real_1(dt_min);
host_to_node_real_1(te_min);
}
/* distribute user RP variables */
DEFINE_ON_DEMAND(eod_rp_variables)
{
#if !RP_NODE
therm_diff_factor = RP_Get_Real("therm_diff");
update_interval = (int)RP_Get_Real("update_thermo");
update_interval_old = (int)RP_Get_Real("update_old");
update_interval_up = (int)RP_Get_Real("update_new");
yplus_thresh = RP_Get_Real("yplus_threshold");
dt_min = RP_Get_Real("delta_t_min");
```

```
te_min = RP_Get_Real("t_eddy_min");
#endif
host_to_node_real_1(therm_diff_factor);
host_to_node_int_1(update_interval);
host_to_node_int_1(update_interval_old);
host_to_node_int_1(update_interval_up);
host_to_node_real_1(yplus_thresh);
host_to_node_real_1(dt_min);
host_to_node_real_1(te_min);
}
/* threshold for validity of equations to determine T_star*/
DEFINE_ON_DEMAND(set_y_T)
{
#if !RP_NODE
y_T = half_zero(0.001,yplus_thresh);
/* ******/
CX_Message("\ny_T = %f \n",y_T);
#endif
host_to_node_real_1(y_T);
}
/* set user defined memories with global search method (very slow) */
DEFINE_ON_DEMAND(set_udm_domain)
{
Domain *d;
face_t f,f0;
cell_t c,c0;
Thread *ft,*ft0,*ct,*ct0;
real ds;
real A_by_es ;
real es[ND_ND] ;
real dr0[ND_ND] ;
real A[ND_ND] ;
real Xf[ND_ND] ;
real Xc[ND_ND] ;
real tau_wall[ND_ND] ;
real wall_dist = 1.0e+20;
real face_dist[ND_ND];
real As;
/* ******/
d=Get_Domain(1);
thread_loop_c(ct,d)
{
begin_c_loop(c,ct)
{
wall_dist = 1.0e+20;
C_CENTROID(Xc,c,ct);
thread_loop_f(ft, d)/* loops over all face threads in a domain*/
{
if(THREAD_TYPE(ft) == THREAD_F_WALL) /* check if it is a wall */
{
begin_f_loop(f, ft) /* loops over faces in a face thread */
```

```
{
F_CENTROID(Xf,f,ft);
NV_VV(face_dist,=,Xc,-,Xf);
if (NV_MAG(face_dist)<wall_dist)
{
f0 = f;
ft0 = ft;
wall_dist = NV_MAG(face_dist);
}}
end_f_loop(f, ft)
if (msg_count%1000==0){
CX_Message("%d Cell Thread ID = %d\n",msg_count,THREAD_ID(ct));
CX_Message("wall dist. = %f\n",wall_dist);}
msg_count++;
}
}
if(NNULLP(THREAD_STORAGE(ft0,SV_WALL_SHEAR)))
{
c0 = F_C0(f0,ft0);
ct0 = THREAD_T0(ft0);
F_CENTROID(Xf,f0,ft0);
NV_VV(face_dist,=,Xc,-,Xf);
BOUNDARY_FACE_GEOMETRY(f0,ft0,A,ds,es,A_by_es,dr0);
As= NV_MAG(A);
NV_VS(tau_wall,=,F_STORAGE_R_N3V(f0,ft0,SV_WALL_SHEAR) , / , As) ;
NV_VS(A,=,A , / , As) ;
C_UDMI(c,ct,TAU_WALL) = NV_MAG(tau_wall);
wall_dist = fabs(NV_DOT(A,face_dist));
C_UDMI(c,ct,Y_PLUS) = wall_dist*sqrt(NV_MAG(tau_wall)*C_R(c,ct))/C_MU_L(c,ct);
C_UDMI(c,ct,N_VEC_X) = -A[0];
C_UDMI(c,ct,N_VEC_Y) = -A[1];
#if RP_3D
C_UDMI(c,ct,N_VEC_Z) = -A[2];
#endif
C_UDMI(c,ct,FACE_X) = Xf[0];
C_UDMI(c,ct,FACE_Y) = Xf[1];
#if RP_3D
C_UDMI(c,ct,FACE_Z) = Xf[2];
#endif
#if THERMO
if(NNULLP(THREAD_STORAGE(ft0,SV_HEAT_FLUX)))
{
C_UDMI(c,ct,TWALL) = F_T(f0,ft0);
C_UDMI(c,ct,QWALL) = WALL_HEAT_FLUX(f0,ft0)/As;
} /* equation 11.9.5 from user's guide, law-of-the-wall */
#endif
}
else
{
C_UDMI(c,ct,TAU_WALL) = -1.0;
C_UDMI(c,ct,Y_PLUS) = -1.0;
```

```
}
}
end_c_loop(c,ct)
}
}
/* set user defined memories only in near wall cell (fastest method) */
DEFINE_ON_DEMAND(set_udm_wall)
{
Domain *d;
face_t f;
cell_t c;
Thread *ft,*ct;
real ds;
real A_by_es ;
real es[ND_ND] ;
real dr0[ND_ND] ;
real A[ND_ND] ;
real Xf[ND_ND] ;
real Xc[ND_ND] ;
real tau_wall[ND_ND] ;
real wall_dist;
real As;
/* *******/
d=Get_Domain(1);
thread_loop_f(ft, d)/* loops over all face threads in a domain*/
{
begin_f_loop(f, ft) /* loops over faces in a face thread */
{
c = F_C0(f, ft);
ct = THREAD_T0(ft);
C_CENTROID(Xc,c,ct);
F_CENTROID(Xf,f,ft);
BOUNDARY_FACE_GEOMETRY(f,ft,A,ds,es,A_by_es,dr0);
As= NV_MAG(A);
if(THREAD_TYPE(ft) == THREAD_F_WALL) /* check if it is a wall */
{
if(NNULLP(THREAD_STORAGE(ft,SV_WALL_SHEAR)))
{
NV_VS(tau_wall,=,F_STORAGE_R_N3V(f,ft,SV_WALL_SHEAR) , / , As) ;
C_UDMI(c,ct,TAU_WALL) = NV_MAG(tau_wall);
NV_VS(A,=,A , / , As) ;
wall_dist = fabs(NV_DOT(A,dr0));
C_UDMI(c,ct,Y_PLUS) = wall_dist*sqrt(NV_MAG(tau_wall)*C_R(c,ct))/C_MU_L(c,ct);
}
else
{
C_UDMI(c,ct,TAU_WALL) = -1.0;
C_UDMI(c,ct,Y_PLUS) = -1.0;
}
C_UDMI(c,ct,N_VEC_X) = -A[0];
C_UDMI(c,ct,N_VEC_Y) = -A[1];
```

```
#if RP_3D
C_UDMI(c,ct,N_VEC_Z) = -A[2];
#endif
C_UDMI(c,ct,FACE_X) = Xf[0];
C_UDMI(c,ct,FACE_Y) = Xf[1];
#if RP_3D
C_UDMI(c,ct,FACE_Z) = Xf[2];
#endif
#if THERMO
if(NNULLP(THREAD_STORAGE(ft,SV_HEAT_FLUX)) && (THREAD_TYPE(ft) ==
THREAD_F_WALL) )
{
C_UDMI(c,ct,TWALL) = F_T(f,ft);
C_UDMI(c,ct,QWALL) = WALL_HEAT_FLUX(f,ft)/As;
} /* equation 11.9.5 from user's guide, law-of-the-wall */
#endif
}}
end_f_loop(f, ft)
}
}
/* set user defined memories zone by zone (recommended over global) */
DEFINE_ON_DEMAND(set_udm_fluid)
{
Domain *d;
face_t f,face;
cell_t c;
Thread *ft,*ct;
real ds;
real A_by_es ;
real es[ND_ND] ;
real dr0[ND_ND] ;
real A[ND_ND] ;
real Xf[ND_ND] ;
real Xpos[ND_ND] ;
real Xc[ND_ND] ;
real face_dist[ND_ND];
real tau_wall[ND_ND] ;
real wall_dist = 1.0e+20;
real As;
/* ******/
d=Get_Domain(1);
thread_loop_f(ft, d)/* loops over all face threads in a domain*/
{
if(THREAD_TYPE(ft) == THREAD_F_WALL) /* check if it is a wall */
{
CX_Message("Wall Thread ID = %d\n",THREAD_ID(ft));
ct = THREAD_T0(ft);
if(FLUID_THREAD_P(ct))
{
begin_c_loop(c,ct)
{
```

```
C_CENTROID(Xc,c,ct);
wall_dist = 1.0e+20;
begin_f_loop(f, ft) /* loops over faces in a face thread */
{
F_CENTROID(Xf,f,ft);
NV_VV(face_dist,=,Xc,-,Xf);
if (NV_MAG(face_dist)<wall_dist)
{
face = f;
wall_dist = NV_MAG(face_dist);
F_CENTROID(Xpos,f,ft);
}
}
end_f_loop(f, ft)
if (msg_count%1000==0){
CX_Message("%d Cell Thread ID = %d\n",msg_count,THREAD_ID(ct));
CX_Message("wall dist. = %f\n",wall_dist);}
msg_count++;
BOUNDARY_FACE_GEOMETRY(face,ft,A,ds,es,A_by_es,dr0);
As= NV_MAG(A);
if(NNULLP(THREAD_STORAGE(ft,SV_WALL_SHEAR)))
{
NV_VS(tau_wall,=,F_STORAGE_R_N3V(face,ft,SV_WALL_SHEAR) , / , As) ;
C_UDMI(c,ct,TAU_WALL) = NV_MAG(tau_wall);
NV_VS(A,=,A , / , As) ;
C_UDMI(c,ct,Y_PLUS) = wall_dist*sqrt(NV_MAG(tau_wall)*C_R(c,ct))/C_MU_L(c,ct);
C_UDMI(c,ct,N_VEC_X) = -A[0];
C_UDMI(c,ct,N_VEC_Y) = -A[1];
#if RP_3D
C_UDMI(c,ct,N_VEC_Z) = -A[2];
#endif
C_UDMI(c,ct,FACE_X) = Xpos[0];
C_UDMI(c,ct,FACE_Y) = Xpos[1];
#if RP_3D
C_UDMI(c,ct,FACE_Z) = Xpos[2];
#endif
#if THERMO
if(NNULLP(THREAD_STORAGE(ft,SV_HEAT_FLUX)))
{
C_UDMI(c,ct,TWALL) = F_T(face,ft);
C_UDMI(c,ct,QWALL) = WALL_HEAT_FLUX(face,ft)/As;
} /* equation 11.9.5 from user's guide */
#endif
}
else
{
C_UDMI(c,ct,TAU_WALL) = -1.0;
C_UDMI(c,ct,Y_PLUS) = -1.0;
}}
end_c_loop(c,ct)
}
```

```
}
}
}
/* initialize all UDMs manually */
DEFINE_ON_DEMAND(eod_udm_zero)
{
cell_t c;
Thread *ct;
Domain *d;
d = Get_Domain(1);
/* loop over all cell threads in the domain */
thread_loop_c(ct,d)
{
/* loop over all cells */
begin_c_loop_all(c,ct)
{
C_UDMI(c,ct,TAU_WALL) = -1.0;
C_UDMI(c,ct,Y_PLUS) = -1.0;
C_UDMI(c,ct,N_VEC_X) = 0.;
C_UDMI(c,ct,N_VEC_Y) = 0.;
#if RP_3D
C_UDMI(c,ct,N_VEC_Z) = 0.;
#endif
C_UDMI(c,ct,FACE_X) = 0.;
C_UDMI(c,ct,FACE_Y) = 0.;
#if RP_3D
C_UDMI(c,ct,FACE_Z) = 0.;
#endif
C_UDMI(c,ct,TWALL) = 0.;
C_UDMI(c,ct,QWALL) = 0.;
C_UDMI(c,ct,WALL_HIT) = 0.;
}
end_c_loop_all(c,ct)
}
}
/* initialize UDMs on solver init */
DEFINE_INIT(udm_init_func,d)
{
cell_t c;
Thread *ct;
if (NULLP(user_particle_vars)) Init_User_Particle_Vars();
/* loop over all cell threads in the domain */
thread_loop_c(ct,d)
{
/* loop over all cells */
begin_c_loop_all(c,ct)
{
C_UDMI(c,ct,TAU_WALL) = -1.0;
C_UDMI(c,ct,Y_PLUS) = -1.0;
C_UDMI(c,ct,N_VEC_X) = 0.;
C_UDMI(c,ct,N_VEC_Y) = 0.;
```

```
#if RP_3D
C_UDMI(c,ct,N_VEC_Z) = 0.;
#endif
C_UDMI(c,ct,FACE_X) = 0.;
C_UDMI(c,ct,FACE_Y) = 0.;
#if RP_3D
C_UDMI(c,ct,FACE_Z) = 0.;
#endif
C_UDMI(c,ct,TWALL) = 0.;
C_UDMI(c,ct,QWALL) = 0.;
C_UDMI(c,ct,WALL_HIT) = 0.;
}
end_c_loop_all(c,ct)
}
}
/* modify the DPM model to do a user defined turbulent dispersion */
DEFINE_DPM_DRAG(dpm_stochastic_drag,Re,p)
{
cphase_state_t *c = &(p->cphase); /* pointer to continuous phase properties, velocity etc.
(dpm.h) */
real Vrel;
real drag_factor;
real k,eps,eplus,TL_plus = 1.0; /* dimensionless time scale */
real Y_plus_p = 200.0; /* dimensionless particle dist. */
cell_t c0 = RP_CELL(&(p->cCell)); /* index of cell in which currently tracked particle is in */
Thread *t0 = RP_THREAD(&(p->cCell)); /* cell thread to which above cell belongs */
real coeff;
cxboolean check_time = FALSE;
int dim = ND_ND;
real pv[3] = {0.,0.,0.}; /* particle velocity vector */
real pp[3] = {0.,0.,0.}; /* particle position vector */
real I[3] = {0.,0.,0.}; /* streamwise direction vector at particle location */
real J[3] = {0.,0.,0.}; /* wall normal vector */
real K[3] = {0.,0.,0.}; /* spanwise direction vector */
int i;
real Vprime_rms;
real Vprime_vec[3]; /* fluct. vel. vector */
real vel_fric; /* u_tauwall , frictional velocity */
real y_dist;
NV_V(pv, =, p->state.V);
NV_V(pp, =, p->state.pos); /* position of particle, x,y,z-coord. */
#if !INTERPOLATION
c->V[0] = C_U(c0,t0);
c->V[1] = C_V(c0,t0);
#if RP_3D
c->V[2] = C_W(c0,t0);
#endif
#endif /* INTERPOLATION */
#if !PARALLEL
update_interval_old = (int)RP_Get_Real("update_old");
update_interval_up = (int)RP_Get_Real("update_new");
```

```
yplus_thresh = RP_Get_Real("yplus_threshold");
dt_min = RP_Get_Real("delta_t_min");
te_min = RP_Get_Real("t_eddy_min");
#endif
Y_plus_p = get_yplus_particle(c0,t0,pp);
/* CX_Message("Y_plus_p = %f, px = %f, py = %f, pz = %f\n",Y_plus_p,pp[0],pp[1],pp[2]); */
if(Y_plus_p < yplus_thresh && Y_plus_p > -0.1 )
{
do_stochastics = FALSE;
do_wall_stoch = TRUE;
}
else
{
do_stochastics = TRUE;
do_wall_stoch = FALSE;
if(p->user[1]>0.)
{
p->eddy_time=DPM_SMALL;
p->user[T_EDDY] = DPM_SMALL;
}}
y_dist = Y_plus_p/sqrt(C_UDMI(c0,t0,TAU_WALL)*C_R(c0,t0))*C_MU_L(c0,t0);
if(y_dist < 0.5*P_DIAM(p) && Y_plus_p > -0.1)
{
do_stochastics = FALSE;
do_wall_stoch = FALSE;
Trap_Particle(p);
}
#if RP_2D
if (rp_axi_swirl)
{
dim = 3;
pv[2] = p->state.V[2];
}
else
{
pv[2]=0;
pp[2]=0.;
}
#endif
/* toggle for comparison - remove later*/
/* do_stochastics = TRUE; */
/* do_wall_stoch = FALSE; */
if (do_wall_stoch)
{
if (p->eddy_time <= DPM_SMALL)
{
check_time = TRUE;
p->user[N_EDDY] ++;
p->user[E_PLUS]=0.; /* reset interaction time */
vel_fric = sqrt(C_UDMI(c0,t0,TAU_WALL)/C_R(c0,t0));
get_rms_vel_fluc(Y_plus_p,vel_fric,Vprime_vec,I,J,K,c,c0,t0);
```

```
#if DEBUGGER
CX_Message("I0 = %f, I1 = %f, I2 = %f\n",I[0],I[1],I[2]);
CX_Message("J0 = %f, J1 = %f, J2 = %f\n",J[0],J[1],J[2]);
CX_Message("K0 = %f, K1 = %f, K2 = %f\n",K[0],K[1],K[2]);
CX_Message("U' = %f, V' = %f, W' = %f\n",Vprime_vec[0],Vprime_vec[1],Vprime_vec[2]);
#endif
/*TL_plus = get_tl_plus(Y_plus_p);*/ /* calc. dimensionless time scale based on particle y_plus
*/
eplus = get_tl_plus(Y_plus_p);
eps = eplus*vel_fric*vel_fric*vel_fric*vel_fric*C_R(c0,t0)/C_MU_L(c0,t0);
k =
0.5*(Vprime_vec[0]*Vprime_vec[0]+Vprime_vec[1]*Vprime_vec[1]+Vprime_vec[2]*Vprime_
vec[2]);
/* calc. modified time scale */
p->eddy_time = TL_plus*C_MU_L(c0,t0)/C_UDMI(c0,t0,TAU_WALL); /* eq. 5 of proposal*/
p->eddy_time = 0.15*k/eps;
if (random_eddy_lifetime)
p->eddy_time *= -log(dpm_uniform_random(p)); /* Eq. 23.2-32 User's Guide, characteristic
eddy
lifetime*/
else
p->eddy_time *=2; /* Eq. 23.2-31 User's Guide, characteristic eddy lifetime */
if (p->state.time == 0.0) p->eddy_time *= dpm_uniform_random(p);
if (p->eddy_time < te_min)
p->eddy_time = te_min;
/*store eddy time*/
p->user[E_PLUS] = eplus; /* time scale constant */
p->user[T_EDDY] = p->eddy_time; /* store eddy lifetime */
p->user[Y_PLUS_P] = Y_plus_p;
p->user[NEW_FORCE] = 1.;
/* velocity fluctuations are calculated using a Gaussian deviate random number */
for (i=0; i<3; i++) p->user[3+i] = dpm_gauss_random(p); /* random fluctuations constant for
t_eddy */
/*for (i=0; i<3; i++) p->user[LAMBDA1+i] = 1.0; /* random fluctuations constant for t_eddy */
for (i=0; i<3; i++) p->V_prime[i] = p->user[LAMBDA1+i] * Vprime_vec[i];
/*******************/
if (msg_count%update_interval_up==0)
{
CX_Message("NEW: particle Y+ = %f , t_eddy = %g , eplus = %f , dt = %g\n"
,Y_plus_p,p->user[T_EDDY],p->user[E_PLUS],P_DT(p));
CX_Message("NEW: u' = %f , v' = %f , w' = %f\n"
,p->V_prime[0],p->V_prime[1],p->V_prime[2]);
msg_count++;
}
}
else
{
check_time = FALSE;
/* p->eddy_time = p->user[1] - p->user[0]; */
vel_fric = sqrt(C_UDMI(c0,t0,TAU_WALL)/C_R(c0,t0));
get_rms_vel_fluc(Y_plus_p,vel_fric,Vprime_vec,I,J,K,c,c0,t0);
```

```
for (i=0; i<3; i++) p->V_prime[i] = p->user[LAMBDA1+i] * Vprime_vec[i];
/*****************/
if (msg_count%update_interval_old==0)
{
CX_Message("OLD: particle Y+ = %f , remaining eddy-lifetime = %g ( = %3.2f \% ), dt = %g\n"
,Y_plus_p,p->eddy_time,100*p->eddy_time/p->user[T_EDDY],P_DT(p));
CX_Message("OLD: u' = %f , v' = %f , w' = %f\n"
,p->V_prime[0],p->V_prime[1],p->V_prime[2]);
msg_count++;
}
}
msg_count++;
for (i=0; i<dim; i++) c->V[i] += p->V_prime[i];/* add fluctuating vel. to mean cont. phase vel*/
#if 0
CX_Message("u = %f ; v = %f ; w = %f ; t = %f ; Y_plus_p = %f ; Y = %f\n ",
c->V[0], c->V[1],
c->V[2],p-
>state.time,Y_plus_p,Y_plus_p*C_MU_L(c0,t0)/sqrt(C_UDMI(c0,t0,TAU_WALL)*C_R(c0,t0)
));
#endif
}
/* standard stochastic tracking outside of boundary layer */
if (do_stochastics)
{
if (p->eddy_time <= DPM_SMALL)
{
check_time = TRUE;
p->user[9] ++;
p->eddy_time = C_MU_FACTOR(p->injection) * c->tke / c->ted; /* Eq. 23.2-23 User's Guide,
fluid
Lagrangian integral time */
if (random_eddy_lifetime)
p->eddy_time *= -log(dpm_uniform_random(p)); /* Eq. 23.2-32 User's Guide, characteristic
eddy
lifetime */
else
p->eddy_time *=2; /* Eq. 23.2-31 User's Guide, characteristic eddy lifetime */
if (p->state.time == 0.0) p->eddy_time *= dpm_uniform_random(p);
if (p->eddy_time < te_min)
p->eddy_time = te_min;
/* CX_Message("tke %e, ted %e, eddy_time %e\n", c->tke, c->ted, p->eddy_time); */
/* velocity fluctuations are calculated using a Gaussian deviate random number */
if (sg_rsm)
{
real random[3];
real chc11=0., chc12=0., chc13=0.;
real chc22=0., chc23=0., chc33=0.;
chc11 = sqrt(C_RUU(c0,t0));
chc12 = C_RUV(c0,t0)/chc11;
chc22 = sqrt( MAX(SMALL, C_RVV(c0,t0)-chc12*chc12) );
#if RP_2D
```

```
if (sg_swirl)
{
#endif
chc13 = C_RUW(c0,t0)/chc11;
chc23 = (C_RVW(c0,t0)-chc12*chc13)/chc22;
chc33 = sqrt( MAX(SMALL, C_RWW(c0,t0)-chc13*chc13-chc23*chc23) );
#if RP_2D
}
else
chc33 = sqrt(C_RWW(c0,t0));
#endif
for (i=0; i<3; i++) random[i] = dpm_gauss_random(p);
p->V_prime[0] = random[0] * chc11;
p->V_prime[1] = random[0] * chc12 + random[1] * chc22;
#if RP_2D
if (rp_axi)
#endif
p->V_prime[2] = random[0] * chc13 + random[1] * chc23 +
random[2] * chc33;
#if RP_2D
else
p->V_prime[2] = 0.;
if (rp_axi_swirl)
{
/* tracking done in 3-D, rotate fluctuation velocity to particle */
/* position. No need for this if isotropic fluctuations */
real R = sqrt(SQR(p->state.pos[1]) + SQR(p->state.pos[2]));
real over_R = R/SQR(MAX(R,DPM_SMALL));
real Vr=p->V_prime[1];
real Vt=p->V_prime[2];
p->V_prime[1] = (Vr*p->state.pos[1] - Vt*p->state.pos[2])*over_R;
p->V_prime[2] = (Vr*p->state.pos[2] + Vt*p->state.pos[1])*over_R;
}
#endif
}
else
{
Vprime_rms = sqrt(2./3. * c->tke); /* Eq. 23.2.27 User's Guide, RMS fluctuating components */
for (i=0; i<3; i++) p->V_prime[i] = dpm_gauss_random(p) * Vprime_rms; /* Eq. 23.2.28-30
User's
Guide, velocity fluctuations */
}
}
else
{
/* rescale the remaining portion of eddy life time and rms values */
p->eddy_time *= (p->old_ted/p->old_tke) * (c->tke/c->ted);
for (i=0; i<dim; i++) p->V_prime[i] *= sqrt(c->tke/p->old_tke);
check_time = FALSE;
}
p->old_tke = c->tke;
```

```
p->old_ted = c->ted;
for (i=0; i<dim; i++) c->V[i] += p->V_prime[i];/* add fluctuating vel. to mean cont. phase vel*/
}
Vrel = 0.0;
for (i=0; i<dim; i++)
Vrel += SQR(c->V[i]-pv[i]);
Vrel = sqrt(Vrel);
if (rp_visc)
p->Re = c->rho * P_DIAM(p) * Vrel / c->mu;
else
p->Re = 0.;
coeff = SphereDragCoeff(p->Re);
if (P_DIAM(p) != 0.0)
drag_factor = coeff * c->mu / ( P_RHO(p) * P_DIAM(p) * P_DIAM(p));
else
drag_factor = 1.;
if (check_time && (do_stochastics||do_wall_stoch) ) /* !!! */
{
real drag_f = MAX(drag_factor, ACCURACY);
/* limit the eddy_time by the eddy crossing time too */
real Le = C_MU_FACTOR(p->injection) / 1.225 * pow(c->tke,1.5)/ c->ted;
if (Le < Vrel/drag_f)
p->eddy_time = MIN(p->eddy_time,
-2*log(1. - Le/(Vrel/drag_f))/drag_f); /* Eq. 23.2.33 User's Guide, particle eddy crossing time */
/*CX_Message("std.-tracking: TL = %g \n",p->eddy_time);*/
}
return coeff;
}
/* set particle trajectory integration time step size */
DEFINE_DPM_TIMESTEP(dpm_ts_by_eddy_time,p, dt)
{
#if !PARALLEL
dt_min = RP_Get_Real("delta_t_min");
te_min = RP_Get_Real("t_eddy_min");
#endif
if (do_stochastics)
{
if (0) /* (msg_count%update_interval==0) */
{
CX_Message("dt %e, eddy_time %e\n", dt, p->eddy_time);
msg_count++;
}
if (dt>p->eddy_time)
return p->eddy_time;
}
if (do_wall_stoch)
{
if (0) /* (msg_count%update_interval==0) */
{
CX_Message("dt %e, eddy_time %e\n", dt, p->eddy_time);
msg_count++;
```

```
}
/*if (dt>p->eddy_time) return p->eddy_time;*/
dt = p->user[T_EDDY]/10.;
if (dt < dt_min) dt = dt_min;
}
return dt;
}
/* name particle scalars for post-processing */
DEFINE_DPM_SCALAR_UPDATE(dpm_sc_interact,cell,thread,initialize,p)
{
cell_t c = RP_CELL(&p->cCell); /* Get Cell and Thread from */
Thread *t = RP_THREAD(&p->cCell); /* Particle Structure using new macros*/
real pp[3] = {0.,0.,0.}; /* particle position vector */
NV_V(pp, =, p->state.pos); /* position of particle, x,y,z-coord. */
if (initialize)
{
/* this is the initialization call */
p->user[E_PLUS] = 0.; /* e_plus */
p->user[T_EDDY] = 0.; /* eddy lifetime */
p->user[Y_PLUS_P] = get_yplus_particle(c,t,pp); /* y_plus value at particle location */
p->user[LAMBDA1] = dpm_gauss_random(p);
p->user[LAMBDA2] = dpm_gauss_random(p);
p->user[LAMBDA3] = dpm_gauss_random(p);
p->user[T_FORCE_X] = 0.; /* thermophoretic body force x */
p->user[T_FORCE_Y] = 0.; /* thermophoretic body force y */
p->user[T_FORCE_Z] = 0.; /* thermophoretic body force z */
p->user[N_EDDY] = 0.; /* number of eddies */
p->user[NEW_FORCE] = 0.; /* flag for thermophoretic model */
strcpy(user_particle_vars[E_PLUS].name,"e_plus");
strcpy(user_particle_vars[E_PLUS].label,"e_Plus");
strcpy(user_particle_vars[T_EDDY].name,"t_eddy");
strcpy(user_particle_vars[T_EDDY].label,"t_Eddy");
strcpy(user_particle_vars[Y_PLUS_P].name,"y_plus");
strcpy(user_particle_vars[Y_PLUS_P].label,"Y_Plus");
strcpy(user_particle_vars[LAMBDA1].name,"lambda_1");
strcpy(user_particle_vars[LAMBDA1].label,"Lambda_1");
strcpy(user_particle_vars[LAMBDA2].name,"lambda_2");
strcpy(user_particle_vars[LAMBDA2].label,"Lambda_2");
strcpy(user_particle_vars[LAMBDA3].name,"lambda_3");
strcpy(user_particle_vars[LAMBDA3].label,"Lambda_3");
strcpy(user_particle_vars[T_FORCE_X].name,"t_force_x");
strcpy(user_particle_vars[T_FORCE_X].label,"t_Force_x");
strcpy(user_particle_vars[T_FORCE_Y].name,"t_force_y");
strcpy(user_particle_vars[T_FORCE_Y].label,"t_Force_y");
strcpy(user_particle_vars[T_FORCE_Z].name,"t_force_z");
strcpy(user_particle_vars[T_FORCE_Z].label,"t_Force_z");
strcpy(user_particle_vars[N_EDDY].name,"n_eddy");
strcpy(user_particle_vars[N_EDDY].label,"N_Eddy");
strcpy(user_particle_vars[NEW_FORCE].name,"new_force");
strcpy(user_particle_vars[NEW_FORCE].label,"New_Force");
}
```

```
else
{
/* not needed */
}
}
/* UD boundary condition for reflection */
DEFINE_DPM_BC(dpm_bc_fluct,p,t,f,f_normal,dim)
{
Reflect_Particle(p,f_normal,dim,f,t);
if (do_stochastics || do_wall_stoch)
{
/* Reflect turbulent fluctuations also */
/* No effect of normal wall velocity onto turbulent fluctuations */
/* Compute normal velocity. */
real vn = 0;
real normal[3] = {0., 0., 0.};
int i, idim = dim;
/* estimate the normal */
#if RP_2D
if (rp_axi_swirl)
{
real R = sqrt(p->state.pos[1]*p->state.pos[1] +
p->state.pos[2]*p->state.pos[2]);
if (R > 1.e-20)
{
idim = 3;
normal[0] = f_normal[0];
normal[1] = (f_normal[1]*p->state.pos[1])/R;
normal[2] = (f_normal[1]*p->state.pos[2])/R;
}
else
{
for (i=0; i<idim; i++)
normal[i] = f_normal[i];
}
}
else
#endif
for (i=0; i<idim; i++)
normal[i] = f_normal[i];
/* reflect turbulent fluctuation */
for(i=0; i<idim; i++)
vn += p->V_prime[i]*normal[i];
/* Subtract off normal velocity. */
for(i=0; i<idim; i++)
p->V_prime[i] -= 2.*vn*normal[i];
}
return PATH_ACTIVE;
}
/* UD boundary condition for trapping particles on a wall and mark position of impact */
DEFINE_DPM_BC(bc_trap,p,t,f,f_normal,dim)
```

```
{
cell_t c0;
Thread *ct0; /*adjacent cell thread */
c0 = F_C0(f,t);
ct0 = THREAD_T0(t);
C_UDMI(c0,ct0,WALL_HIT)++; /* mark cell as particle-containing*/
F_UDMI(f,t,WALL_HIT)++; /* mark cell as particle-containing*/
/* complete deposition, particles being removed from domain*/
if(1)
{
p->stream_index = -1;
p->gvtp.n_trapped++;
return PATH_END;
}
return PATH_ABORT;
}
/* UD boundary condition for particles leaving domain and mark position of impact */
DEFINE_DPM_BC(bc_escape,p,t,f,f_normal,dim)
{
cell_t c0;
Thread *ct0; /*adjacent cell thread */
c0 = F_C0(f,t);
ct0 = THREAD_T0(t);
C_UDMI(c0,ct0,WALL_HIT)++; /* mark cell as particle-containing*/
F_UDMI(f,t,WALL_HIT)++; /* mark cell as particle-containing*/
/* complete deposition, particles being removed from domain*/
if(1)
{
p->stream_index = -1;
p->gvtp.n_escaped++;
return PATH_END;
}
return PATH_ABORT;
}
/* function to add fluctuations of thermophoretic force */
DEFINE_DPM_BODY_FORCE(particle_body_force,p,i)
{
cphase_state_t *c = &(p->cphase); /* pointer to continuous phase properties, velocity etc.
(dpm.h) */
real bforce[3] = {0.,0.,0.};
real Y_plus_p = 200.0; /* dimensionless particle dist. */
cell_t c0 = RP_CELL(&(p->cCell)); /* index of cell in which currently tracked particle is in */
Thread *t0 = RP_THREAD(&(p->cCell)); /* cell thread to which above cell belongs */
int dim = ND_ND;
int j;
real I[3] = {0.,0.,0.}; /* streamwise direction vector at particle location */
real J[3] = {0.,0.,0.}; /* wall normal vector */
real K[3] = {0.,0.,0.}; /* spanwise direction vector */
real x[ND_ND];
real p_dist[ND_ND];
real Vprime_vec[3]; /* fluct. vel. vector */
```

```
real vel_fric; /* u_tauwall , frictional velocity */
/* variables for thermophoretic force */
real T_p;
real T_plus;
real Tstar;
real T_frac;
real F_prime;
real E = 9.793;
real k;
real ka = 0.4187;
real P,Pr,Pr_t;
real alpha;
real T1=0.;
real T2=0.;
real pp[3] = {0.,0.,0.}; /* particle position vector */
#if !PARALLEL
therm_diff_factor = RP_Get_Real("therm_diff");
update_interval = (int)RP_Get_Real("update_thermo");
#endif
NV_V(pp, =, p->state.pos); /* position of particle, x,y,z-coord. */
Y_plus_p = get_yplus_particle(c0,t0,pp);
vel_fric = sqrt(C_UDMI(c0,t0,TAU_WALL)/C_R(c0,t0));
get_rms_vel_fluc(Y_plus_p,vel_fric,Vprime_vec,I,J,K,c,c0,t0);
/* CX_Message("Y_plus_p = %f, px = %f, py = %f, pz = %f\n",Y_plus_p,pp[0],pp[1],pp[2]); */
if(Y_plus_p < yplus_thresh && Y_plus_p > -0.1 )
{
do_thermophoretic = TRUE;
if (THERMO && p->user[NEW_FORCE] >= DPM_SMALL)
{
p->user[NEW_FORCE] = 0.;
/*********************************************************/
T_plus = 0.0116*Y_plus_p*Y_plus_p/(1. + 0.203*Y_plus_p+0.0014*pow(Y_plus_p,2.421));
T_frac = (0.0232*Y_plus_p + 0.002355*Y_plus_p*Y_plus_p - 6.844e-6*pow(Y_plus_p,3.421))/
(1.+0.203*Y_plus_p+0.0014*pow(Y_plus_p,2.421))/(1.+0.203*Y_plus_p+0.0014*pow(Y_plus_
p,2.421));
C_CENTROID(x,c0,t0);
NV_VV(p_dist,=,pp,-,x);
T_p = C_T(c0,t0);
/* temperature interpolation - 1st order Taylor series */
for(j=0;j<dim;j++) T_p += C_T_RG(c0,t0)[j]*p_dist[j];
k =
0.5*(Vprime_vec[0]*Vprime_vec[0]+Vprime_vec[1]*Vprime_vec[1]+Vprime_vec[2]*Vprime_
vec[2]);
T1 = (C_UDMI(c0,t0,TWALL) - T_p)*c-
>rho*C_CP(c0,t0)*pow(C_MU,0.25)*sqrt(k)/C_UDMI(c0,t0,QWALL);
#if 1
Pr_t = 0.85;
Pr = C_MU_L(c0,t0)/C_K_L(c0,t0)*C_CP(c0,t0);
P = 9.24*(pow(Pr/Pr_t,0.75)-1.)*(1.+0.28*exp(-0.007*Pr/Pr_t));
if(Y_plus_p <= y_T)
T2 = Pr*Y_plus_p;
```

```
else
T2 = Pr_t*(1/ka*log(E*Y_plus_p) + P);
#endif
/*Tstar = MAX(T1,T2);*/
Tstar = T2;
alpha = 1. - T_plus*Tstar/T_p;
F_prime = therm_diff_factor*Thermophoretic_Diff(p,c0,t0);
F_prime *= Tstar/T_p*vel_fric*c->rho/c->mu*T_frac;
F_prime *= alpha*dpm_gauss_random(p);
/*****************************************************************************/
J[0] = C_UDMI(c0,t0,N_VEC_X);
J[1] = C_UDMI(c0,t0,N_VEC_Y);
#if RP_3D
J[2] = C_UDMI(c0,t0,N_VEC_Z);
#endif
NV_S(J,*=,F_prime); /* wall-normal-component of force stored in J ...*/
for(j=0;j<dim;j++) p->user[T_FORCE_X+j] = J[j];
/* ...and then permamently stored in scalars for use in DPM_BODY_FORCE */
if (msg_count%update_interval==0)
{
CX_Message("THERMO: T1 = %f, T2 = %f, Tstar = %f, y_plus = %f , Fx' = %lg , Fy' = %lg ,
Fz' =
%lg\n"
,T1,T2,Tstar,Y_plus_p,J[0]/P_MASS(p),J[1]/P_MASS(p),J[2]/P_MASS(p));
msg_count++;
}
}
}
else
{
do_thermophoretic = FALSE;
}
if (do_thermophoretic)
{
for(j=0;j<dim;j++) bforce[j]=-p->user[T_FORCE_X+j];
}
else
{
/* do nothing */
}
/* an acceleration should be returned */
msg_count++;
return (bforce[i]/P_MASS(p));
}
/* Talbot thermophoretic diffusion coefficient*/
DEFINE_DPM_PROPERTY(beta_diff,c,t,p)
{
real coeff=0.;
real Rgas, Lambda, Knudsen, Kfactor;
Material *m = p->injection->material;
cphase_state_t *gas = &(p->cphase);
```

```
real op_press = RP_Get_Real("operating-pressure");
op_press += C_P(c,t);
Rgas = op_press / gas->rho / gas->temp;
Lambda = 1.253314137*gas->mu/op_press*sqrt(Rgas * gas->temp);
Knudsen = Lambda/(.5 * p->state.diam);
Kfactor = (3.75 * gas->mu * Rgas)/MATERIAL_PROP(m,PROP_ktc);
coeff = 6. * M_PI * gas->mu * gas->mu * p->state.diam * 1.17 *
(Kfactor + 2.18 * Knudsen) / MAX(1.e-29, gas->rho *
(1.+ 3.42 * Knudsen) * (1. + 2.* Kfactor + 4.36 * Knudsen));
return coeff;
}
/* for right-hand-side evaluation of eq. 11.9.5, law-of-the-wall */
real half_zero(real x_l ,real x_r)
{
real x_m,y_l,y_m,y_r;
real tolerance = 1.0e-4;
real P,Pr,Pr_t,E=9.793,k=0.4187;
int i = 0;
Pr_t = 0.85;
Pr = 0.7;
P = 9.24*(pow((Pr/Pr_t),.75)-1.)*(1.+0.28*exp(-0.007*Pr/Pr_t));
do
{
x_m = x_l + 0.5*(x_r-x_l); /*split previous interval in two halves*/
/*evaluate target function at new interval borders...*/
y_m = x_m - Pr_t/Pr*(1/k*log(E*x_m)+P);
y_l = x_l - Pr_t/Pr*(1/k*log(E*x_l)+P);
y_r = x_r - Pr_t/Pr*(1/k*log(E*x_r)+P);
/*don't forget to adjust proper function call*/
/* CX_Message("P = %f ; x_m = %f ; y_m = %f ; y_l = %f ; y_r = %f\n",P,x_m,y_m,y_l,y_r); */
/*...and set new interval borders*/
if(y_l*y_m < 0.0)
{
y_l = x_l;
y_r = x_m;
y_m = y_l + 0.5*(y_r-y_l); /*split new interval in two halves*/
}
else if(y_m*y_r < 0.0)
{
y_l = x_m;
y_r = x_r;
y_m = y_l + 0.5*(y_r-y_l); /*split new interval in two halves*/
}
else
{
i++;
if(i > 3)
CX_Message("\nwarning: no valid interval\n");
}
if(y_l < 0.)
y_l = 0.;
```

```
x_l = y_l;
x_r = y_r;
}
while(fabs(y_m-x_m) > tolerance);
return y_m;
}
```

Instructions for executing Stochastic UDF

File\Read\Case & Data
Define\User-Defined/Memory\Number of User-Defined Memory Locations = 11
Define\User-Defined\Functions\Compiled\Build
   under Source Files select Add & select file to work with *.c
   change Library Name or keep default libudf (changed to pl)
Define\Models\Discrete Phase\Tracking\Drag Parameters\Drag Law = udf
dpm_stochastic_drag::pl
   set Tracking Parameters (i.e., Max. Number of Steps, Specify Length Scale or Step Length Factor)
Define\Models\Discrete Phase\UDF
   under User-Defined Functions select
      Scalar Update = dpm_sc_interact::pl
      SPM Time Step = dpm_ts_by_eddy_time::pl
   under User Variables set Number of Scalars = 11
Define\Models\Discrete Phase/Numerics
   under Options: disable Accuracy Control
   under Tracking Scheme Selection: Tracking Scheme = trapezoidal
Solve\Iterate (at least one iteration)
Define\Boundary Conditions\Set = trap
   under Zone select wall (Type = wall)
Define\User Defined\Execute On Demand\set_udm_domain::pl <Execute>
Define\Injections
File\Read\Scheme
Define\User-Defined\DPM Tracking
   set yplus_threshold; delta_t_min, t_eddy_min
File\Write\Case & Data <choose a different name>
Display\Particle Track\Draw Grid etc. <takes long time>
Report\Discrete Phase\Sample
   select Boundaries, Planes, & Release From Injections <Compute>

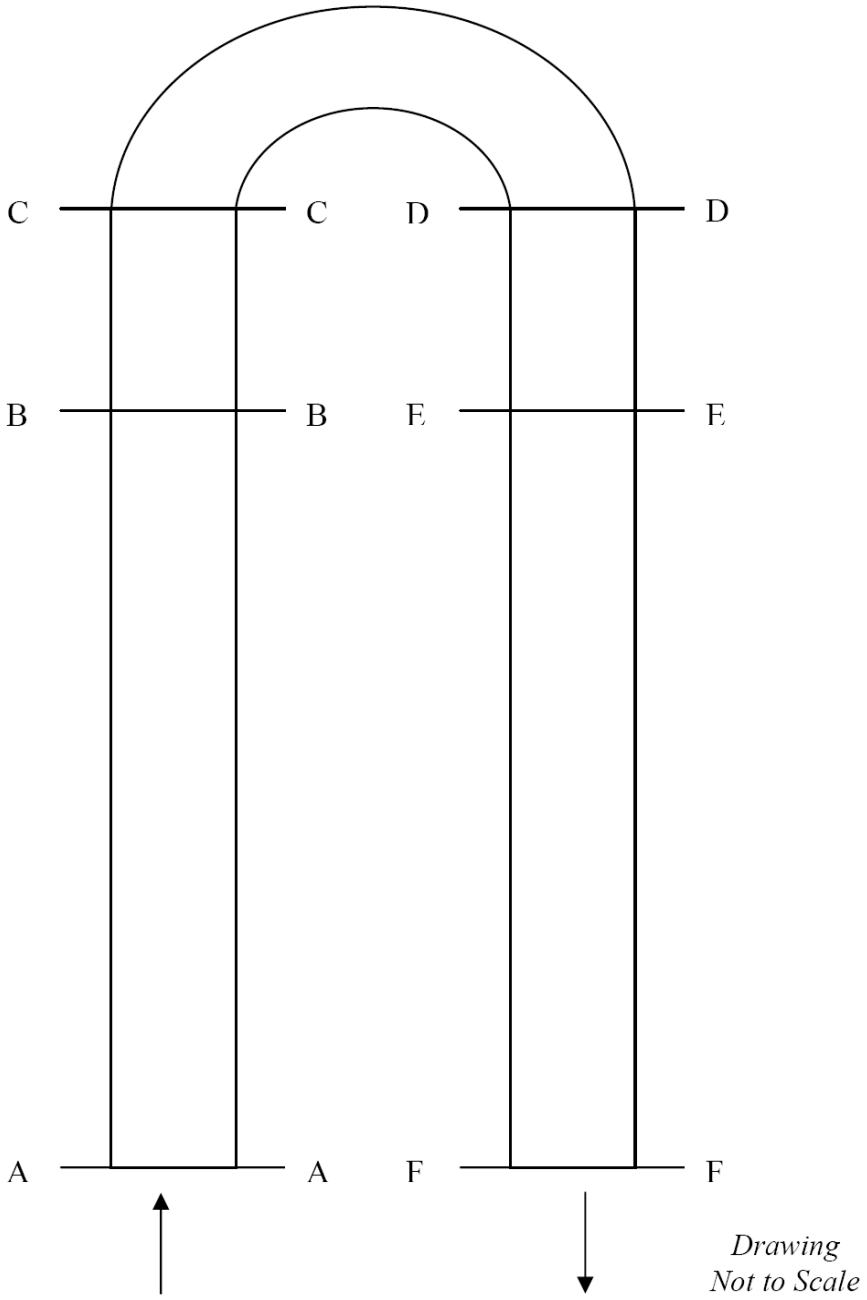
stoch.scm

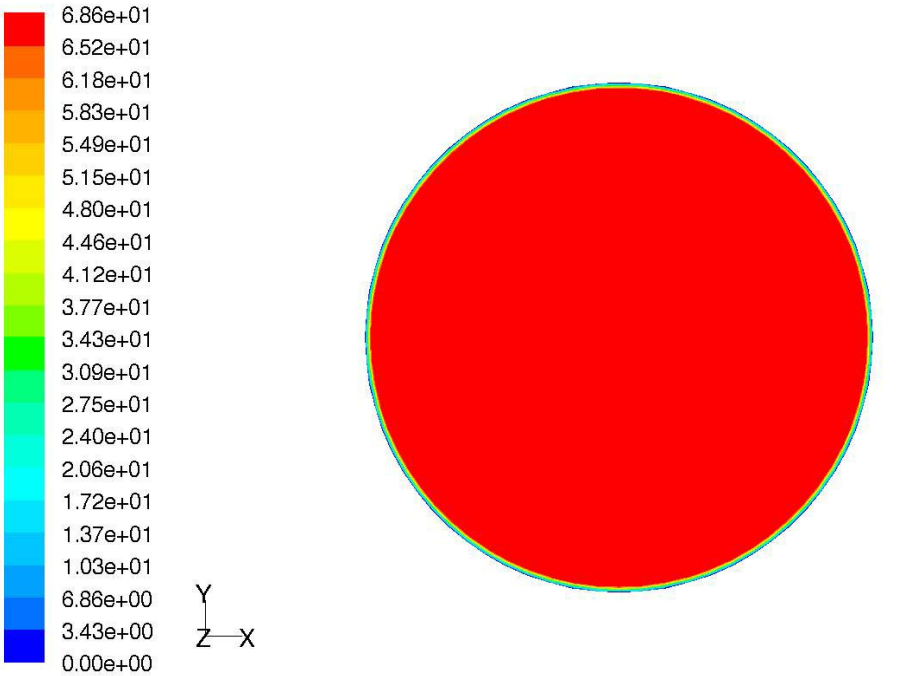
```
( if (not (rp-var-object 'therm_diff))(rp-var-define 'therm_diff 1.0 'real #f))
( if (not (rp-var-object 'update_thermo))(rp-var-define 'update_thermo 100001 'real #f))
( if (not (rp-var-object 'update_old))(rp-var-define 'update_old 1000000 'real #f))
( if (not (rp-var-object 'update_new))(rp-var-define 'update_new 5000000 'real #f))
( if (not (rp-var-object 'yplus_threshold))(rp-var-define 'yplus_threshold 200.0 'real #f))
( if (not (rp-var-object 'delta_t_min))(rp-var-define 'delta_t_min 1.0e-06 'real #f))
( if (not (rp-var-object 't_eddy_min))(rp-var-define 't_eddy_min 1.0e-05 'real #f))

(define gui-stochtrack        ;create panel
  (let ((panel
#f)(THERM_DIFF)(UPDATE_THERMO)(UPDATE_OLD)(UPDATE_NEW)(YPLUS_
THRESHOLD)(DELTA_T_MIN)(T_EDDY_MIN))
    (define (update-cb . args)     ;update panel fields
      (cx-set-real-entry THERM_DIFF (rpgetvar 'therm_diff))
         (cx-set-real-entry UPDATE_THERMO (rpgetvar 'update_thermo))
         (cx-set-real-entry UPDATE_OLD (rpgetvar 'update_old))
         (cx-set-real-entry UPDATE_NEW (rpgetvar 'update_new))
         (cx-set-real-entry YPLUS_THRESHOLD (rpgetvar 'yplus_threshold))
         (cx-set-real-entry DELTA_T_MIN (rpgetvar 'delta_t_min))
         (cx-set-real-entry T_EDDY_MIN (rpgetvar 't_eddy_min))   )
    (define (apply-cb . args)
      (rpsetvar 'therm_diff (cx-show-real-entry THERM_DIFF))
         (rpsetvar 'update_thermo (cx-show-real-entry UPDATE_THERMO))
         (rpsetvar 'update_old (cx-show-real-entry UPDATE_OLD))
         (rpsetvar 'update_new (cx-show-real-entry UPDATE_NEW))
         (rpsetvar 'yplus_threshold (cx-show-real-entry YPLUS_THRESHOLD))
         (rpsetvar 'delta_t_min (cx-show-real-entry DELTA_T_MIN))
         (rpsetvar 't_eddy_min (cx-show-real-entry T_EDDY_MIN))  )
    (lambda args
      (if (not panel)
          (let ((table) (form))
        (set! panel (cx-create-panel "DPM Tracking" apply-cb update-cb))
           (set! table (cx-create-table panel "" 'border #f 'below 0 'right-of 0))
           (set! form  (cx-create-frame table "" 'border #f ))
           (set! THERM_DIFF  (cx-create-real-entry table "therm_diff" 'width 14 'row 1
'col 0 ))
           (set! UPDATE_THERMO  (cx-create-real-entry table "update_thermo" 'width
14 'row 2 'col 0 ))
           (set! UPDATE_OLD  (cx-create-real-entry table "update_old" 'width 14 'row 3
'col 0 ))
           (set! UPDATE_NEW  (cx-create-real-entry table "update_new" 'width 14 'row
4 'col 0 ))
           (set! YPLUS_THRESHOLD  (cx-create-real-entry table "yplus_threshold"
'width 14 'row 5 'col 0 ))
```

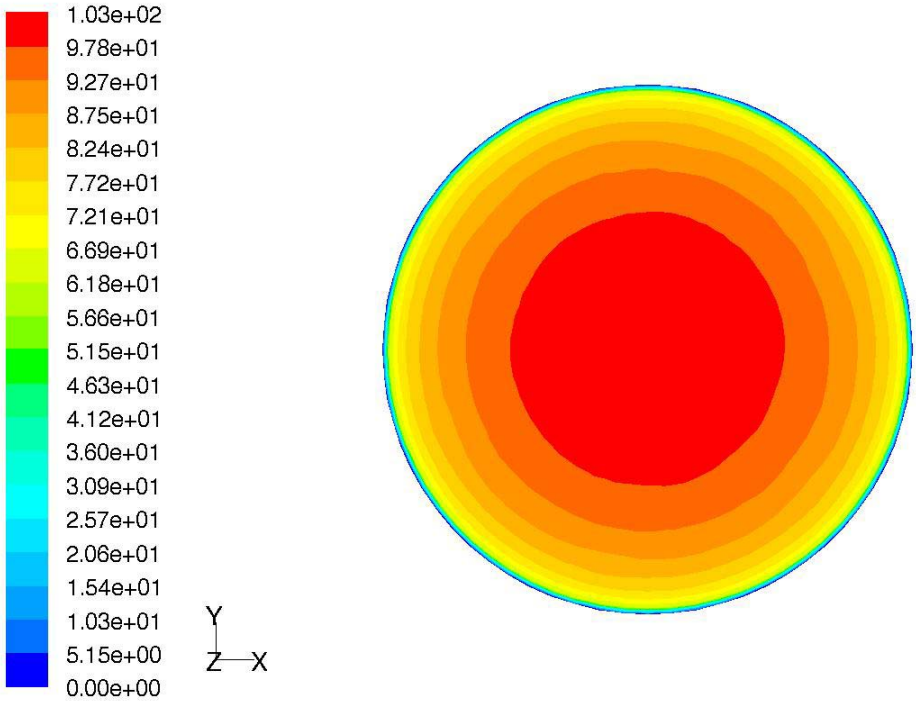
```
            (set! DELTA_T_MIN  (cx-create-real-entry table "delta_t_min" 'width 14 'row
6 'col 0 ))
            (set! T_EDDY_MIN  (cx-create-real-entry table "t_eddy_min" 'width 14 'row 7
'col 0 ))
           )
          ) (cx-show-panel panel) ) ) )
(cx-add-item "User-Defined" "DPM Tracking..." #\U #f cx-client? gui-stochtrack)
```

APPENDIX B

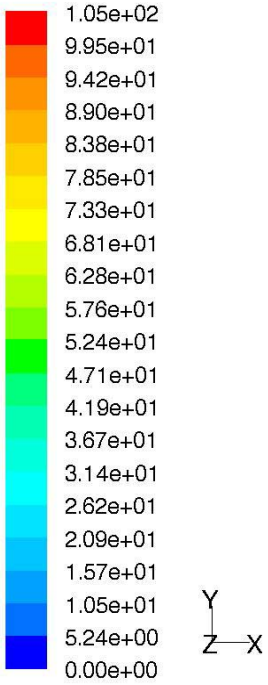

C
C
D
D
B
B
E
E
A
A
F
F
Drawing
Not to Scale

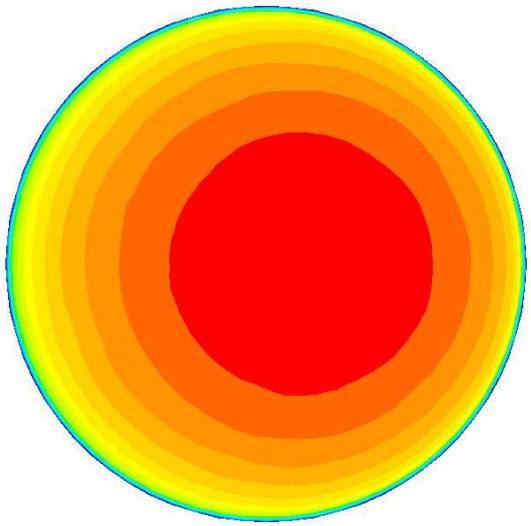


Section A-A

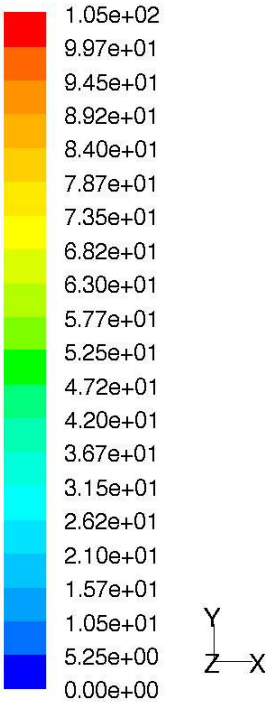


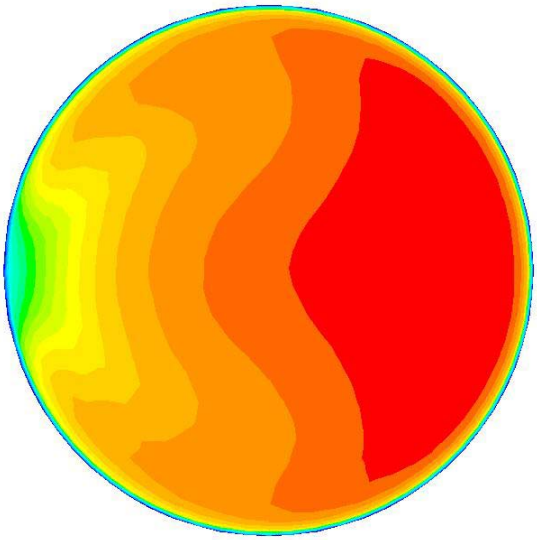
Section B-B



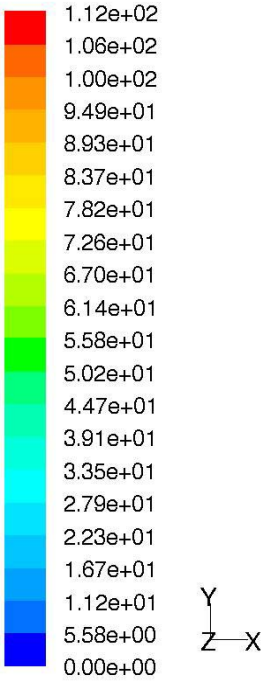


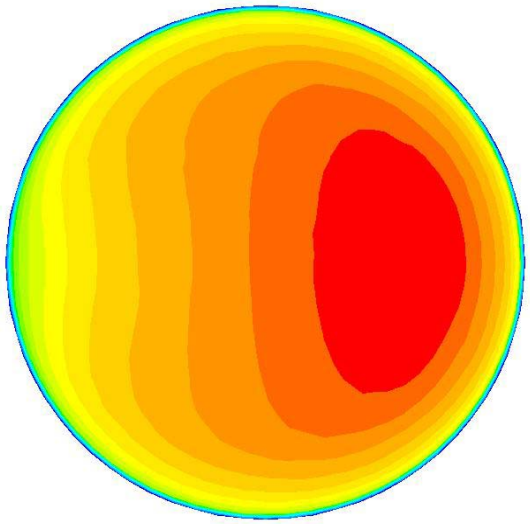
Section C-C



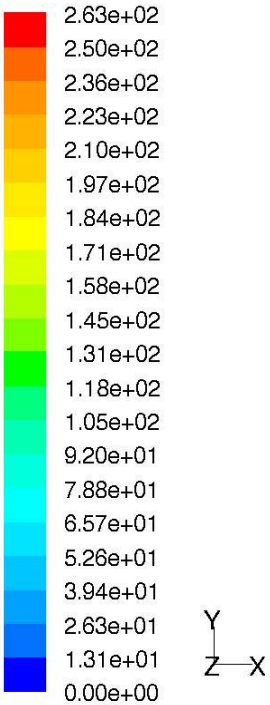


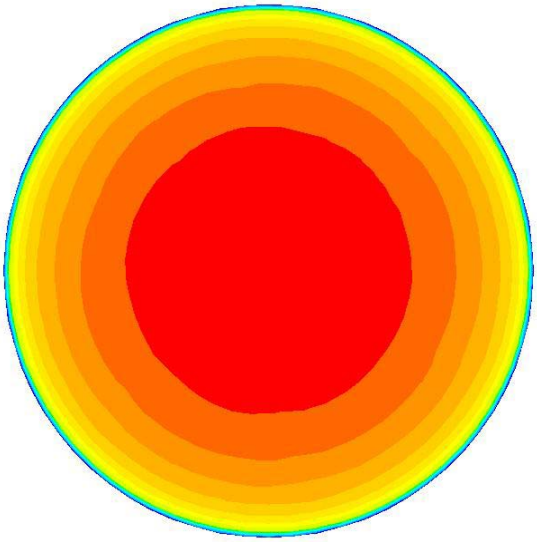
Section D-D



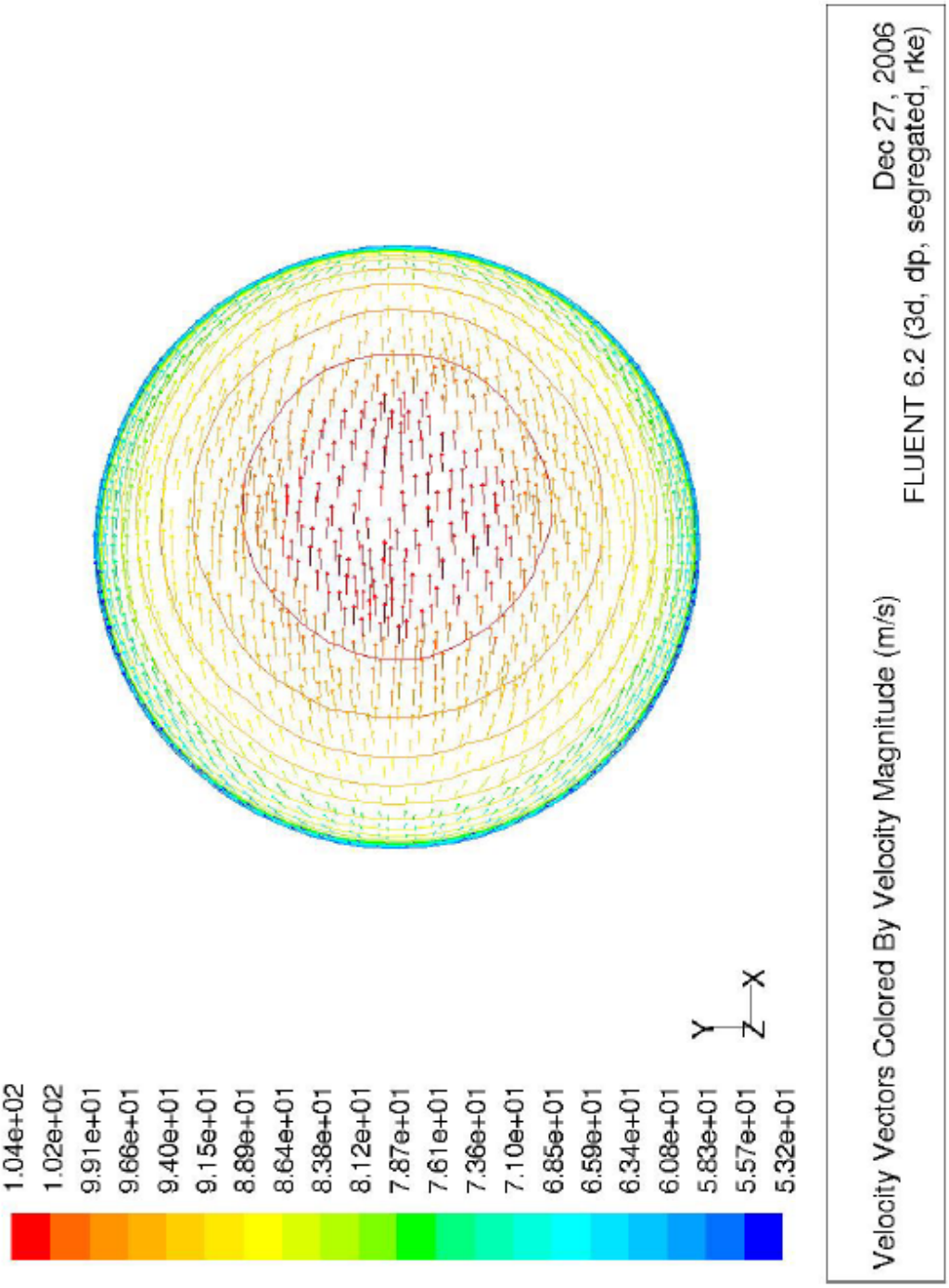


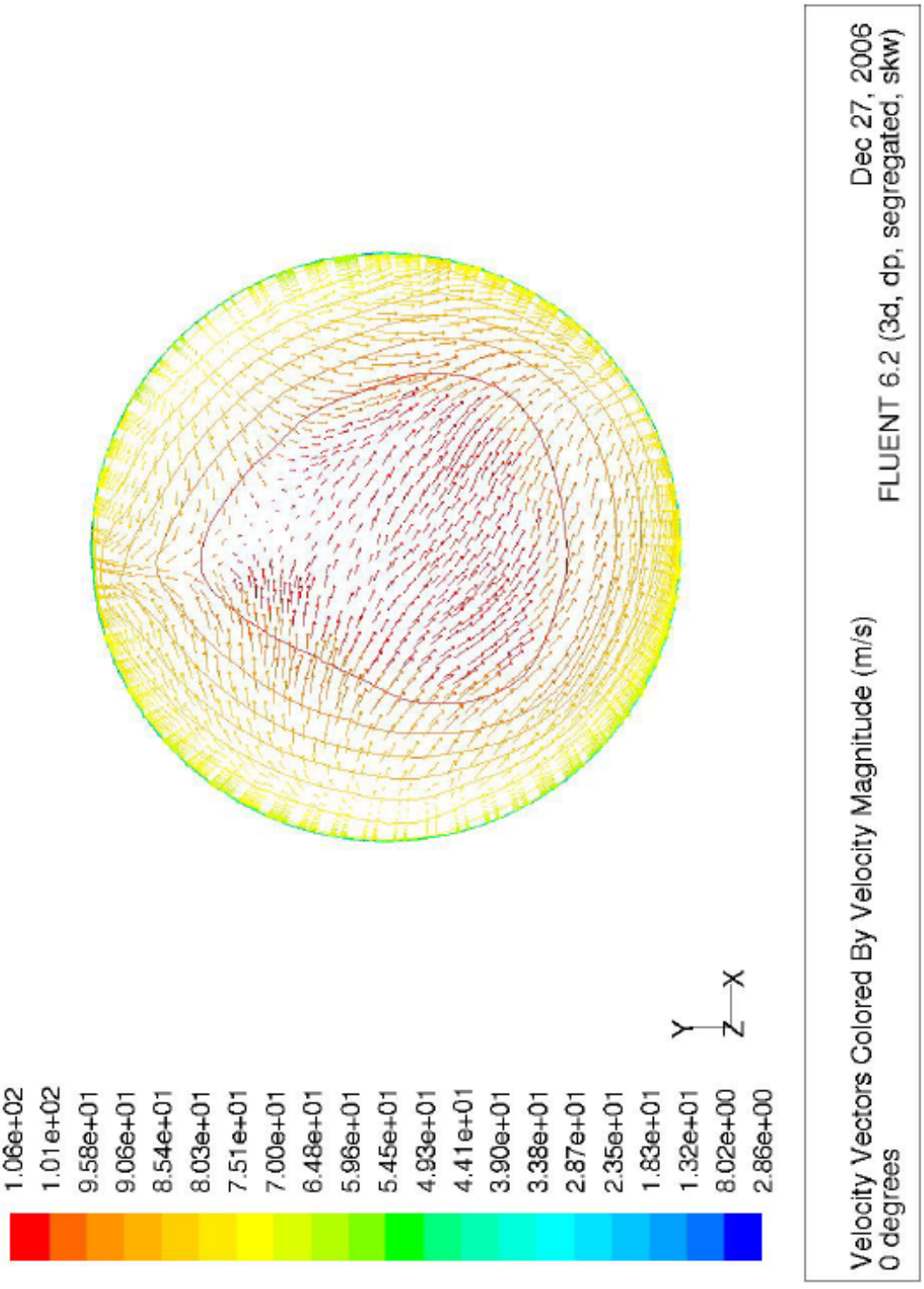
Section E-E

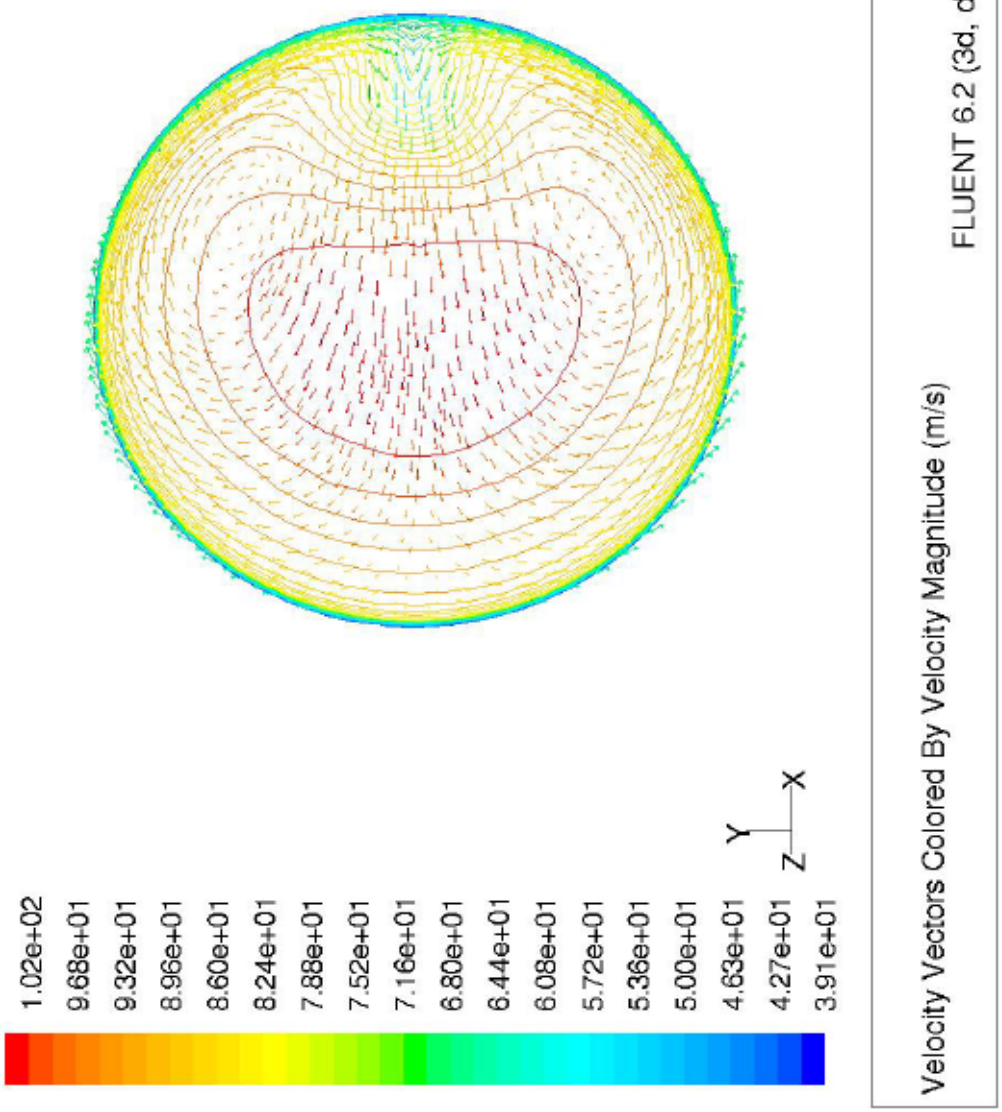


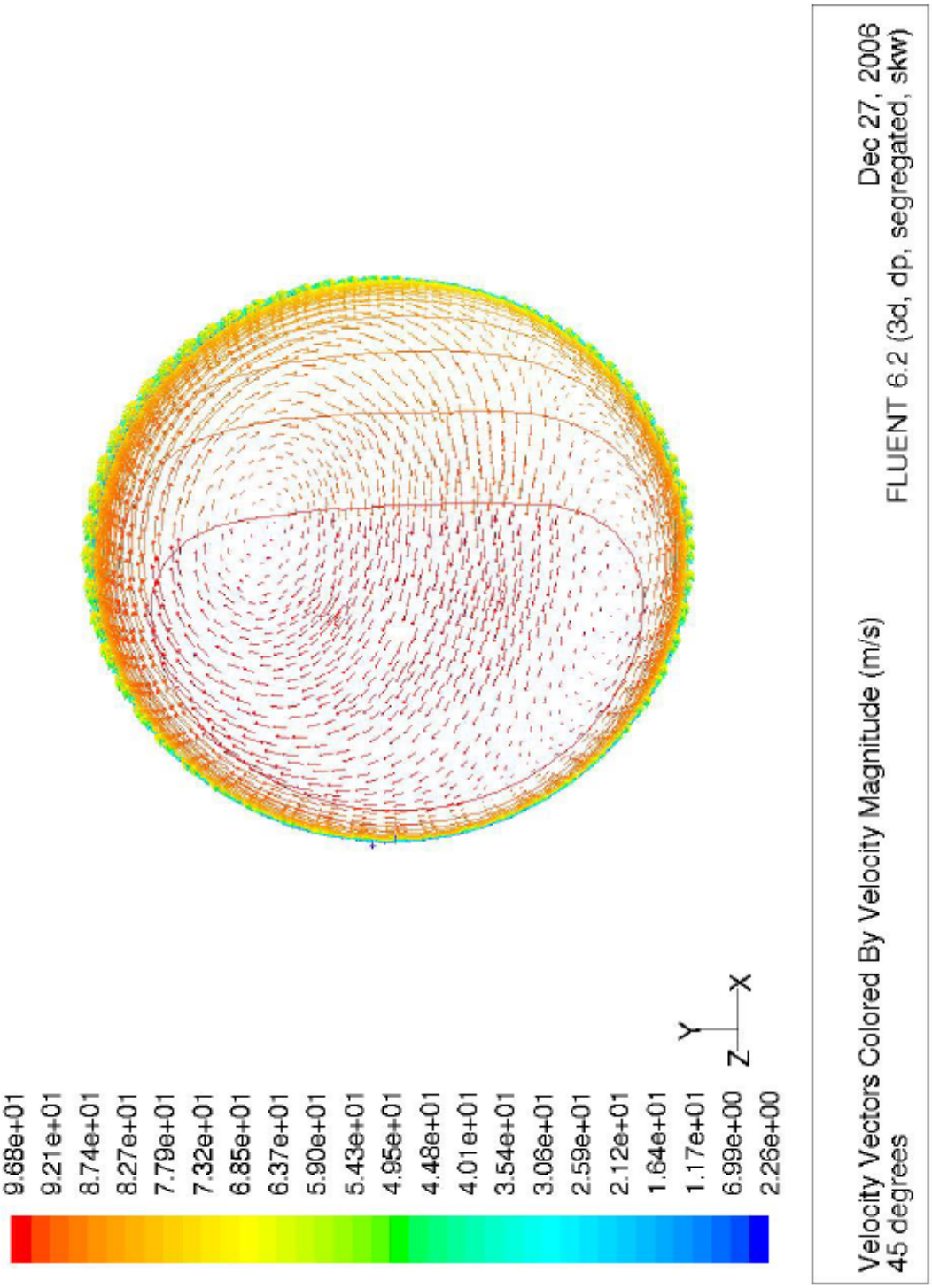


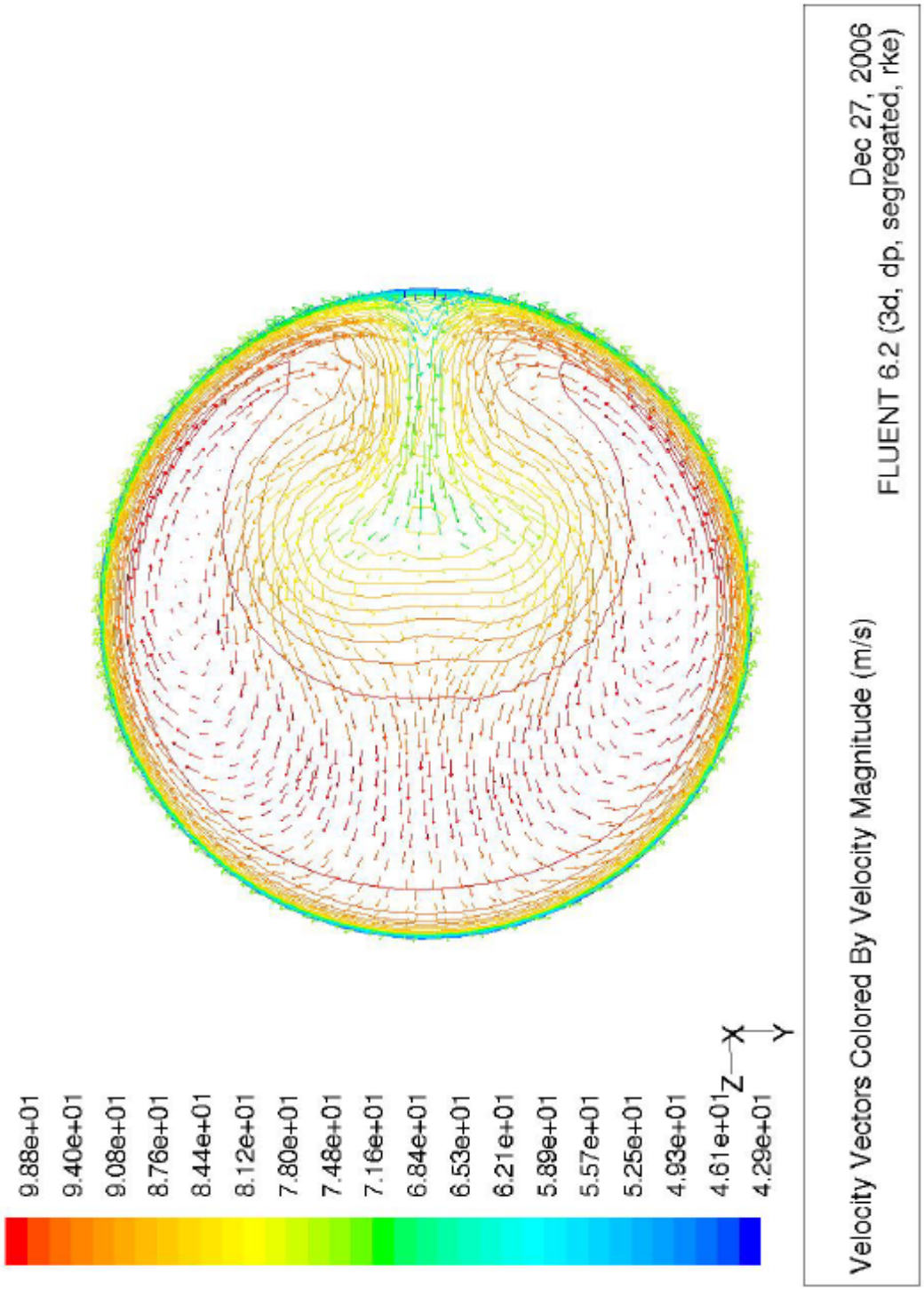
Section F-F

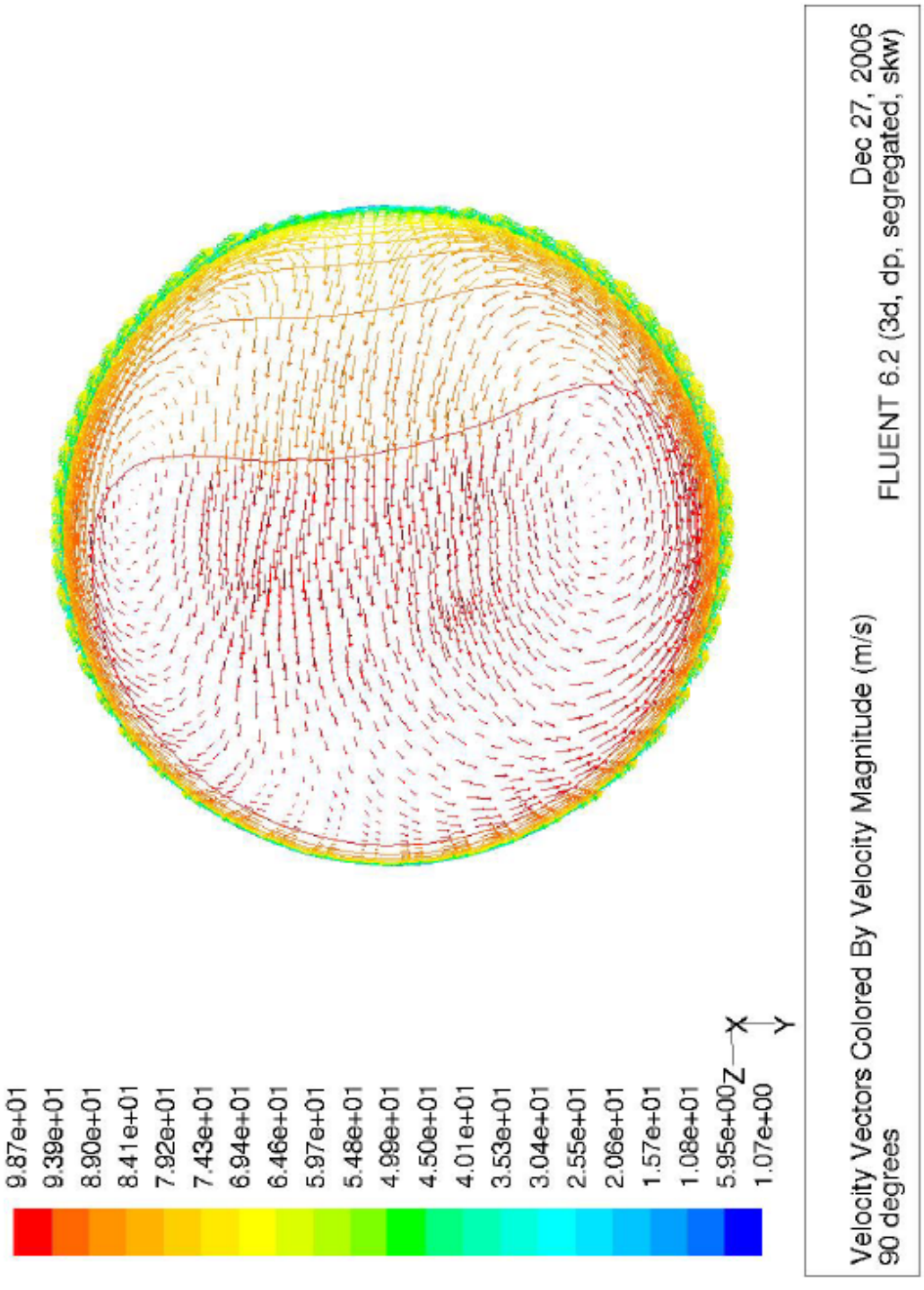

Velocity Vectors Colored By Velocity Magnitude (m/s)
Dec 27, 2006
FLUENT 6.2 (3d, dp, segregated, rke)
1.04e+02
1.02e+02
9.91e+01
9.66e+01
9.40e+01
9.15e+01
8.89e+01
8.64e+01
8.38e+01
8.12e+01
7.87e+01
7.61e+01
7.36e+01
7.10e+01
6.85e+01
6.59e+01
6.34e+01
6.08e+01
5.83e+01
5.57e+01
5.32e+01
Y
Z
X

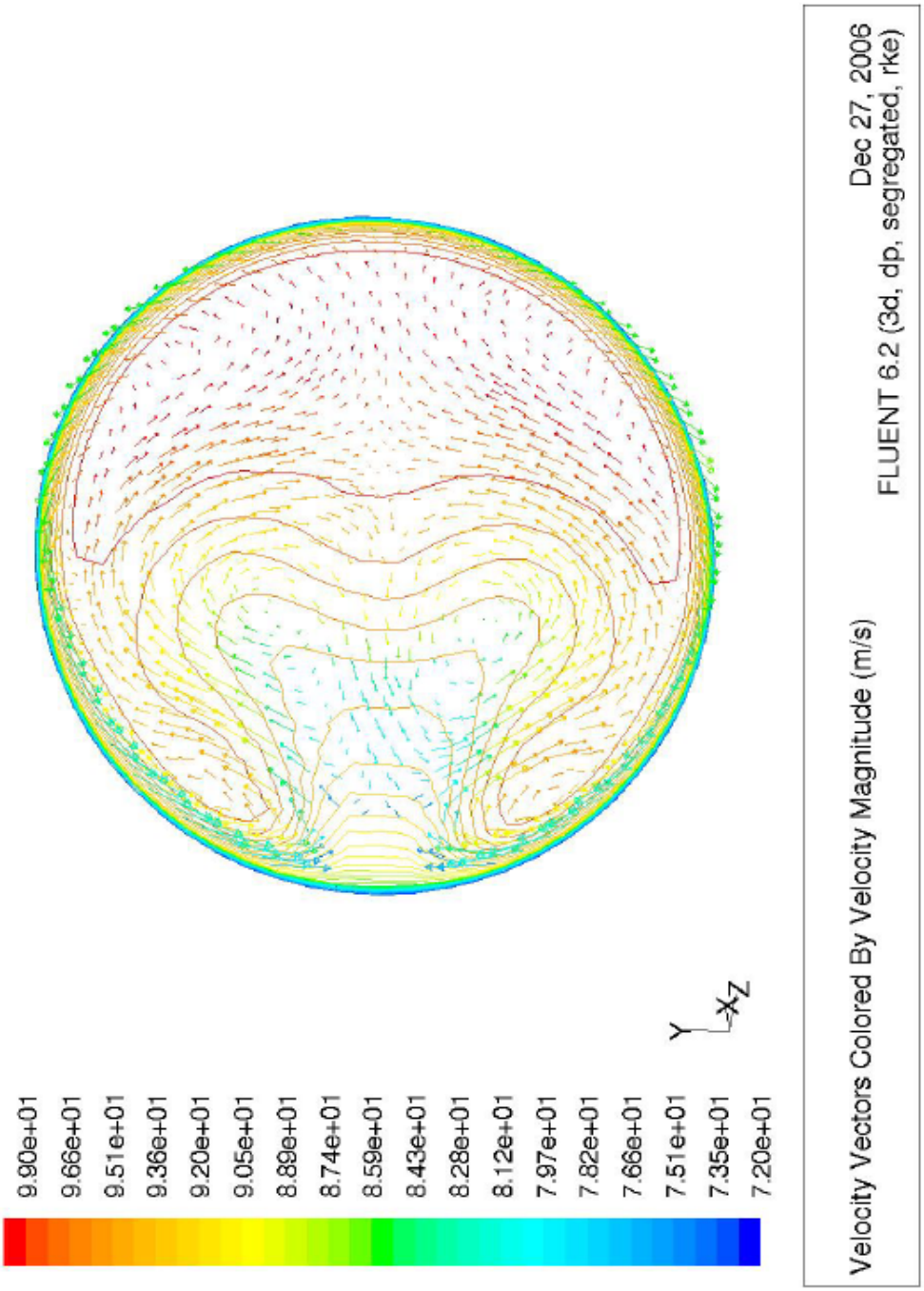

1.06e+02
1.01e+02
9.58e+01
9.06e+01
8.54e+01
8.03e+01
7.51e+01
7.00e+01
6.48e+01
5.96e+01
5.45e+01
4.93e+01
4.41e+01
3.90e+01
3.38e+01
2.87e+01
2.35e+01
1.83e+01
1.32e+01
8.02e+00
2.86e+00
Y
Z
X
Velocity Vectors Colored By Velocity Magnitude (m/s)
0 degrees
Dec 27, 2006
FLUENT 6.2 (3d, dp, segregated, skw)

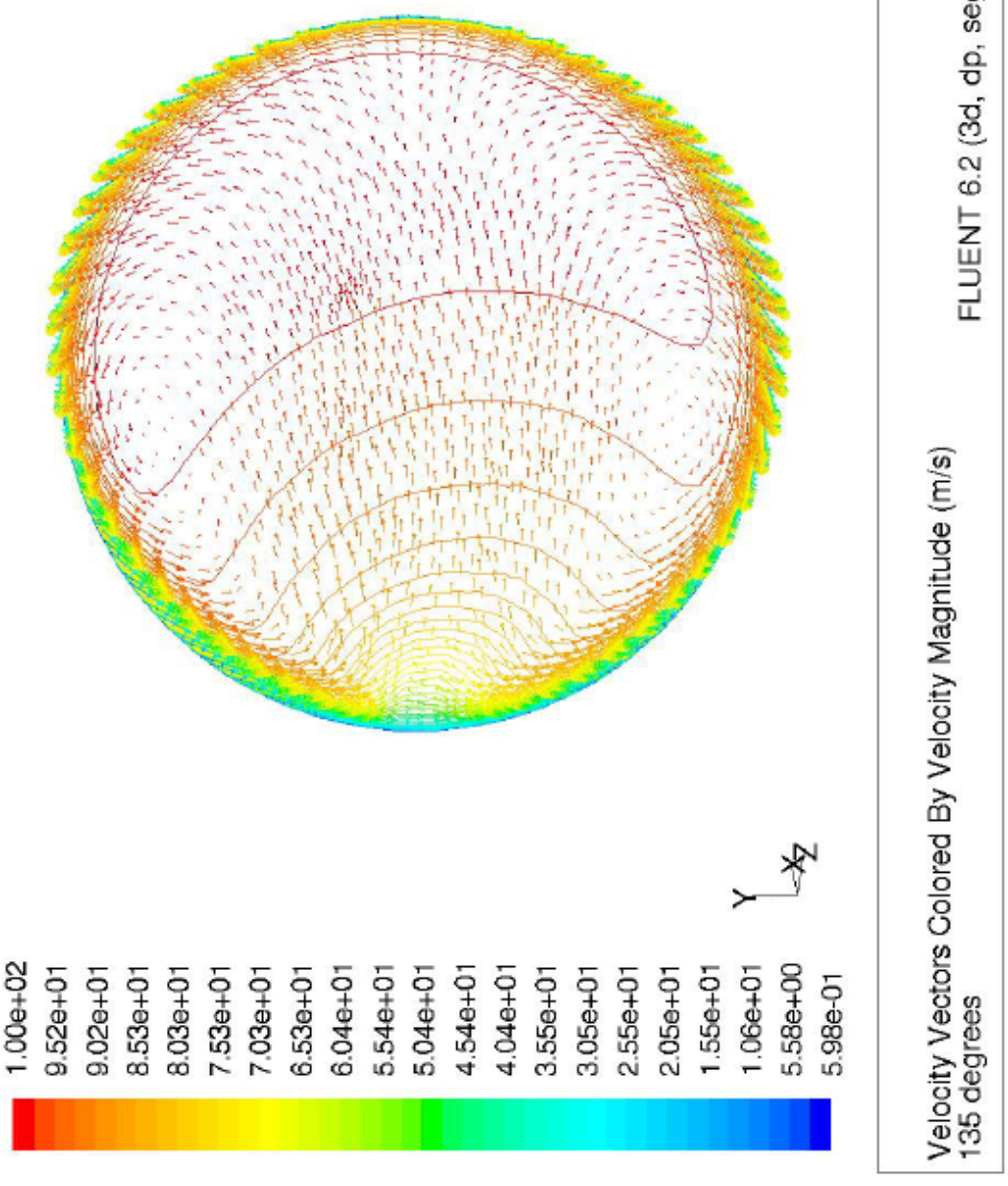

1.02e+02
9.68e+01
9.32e+01
8.96e+01
8.60e+01
8.24e+01
7.88e+01
7.52e+01
7.16e+01
6.80e+01
6.44e+01
6.08e+01
5.72e+01
5.36e+01
5.00e+01
4.63e+01
4.27e+01
3.91e+01
Y
X
Z
Velocity Vectors Colored By Velocity Magnitude (m/s)
Dec 27, 2006
FLUENT 6.2 (3d, dp, segregated, rke)

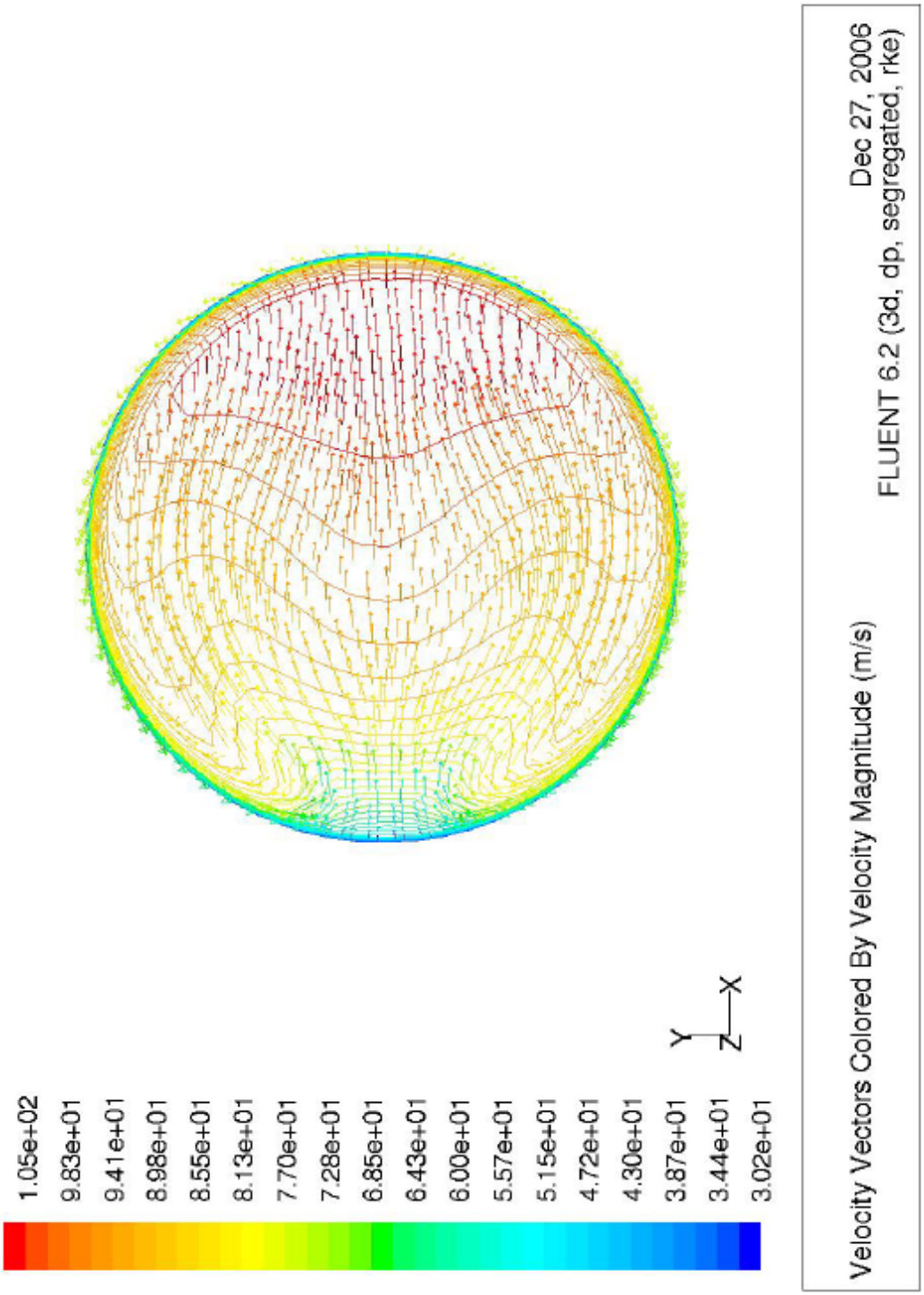

9.68e+01
9.21e+01
8.74e+01
8.27e+01
7.79e+01
7.32e+01
6.85e+01
6.37e+01
5.90e+01
5.43e+01
4.95e+01
4.48e+01
4.01e+01
3.54e+01
3.06e+01
2.59e+01
2.12e+01
1.64e+01
1.17e+01
6.99e+00
2.26e+00
Y
Z
X
Velocity Vectors Colored By Velocity Magnitude (m/s)
45 degrees
Dec 27, 2006
FLUENT 6.2 (3d, dp, segregated, skw)

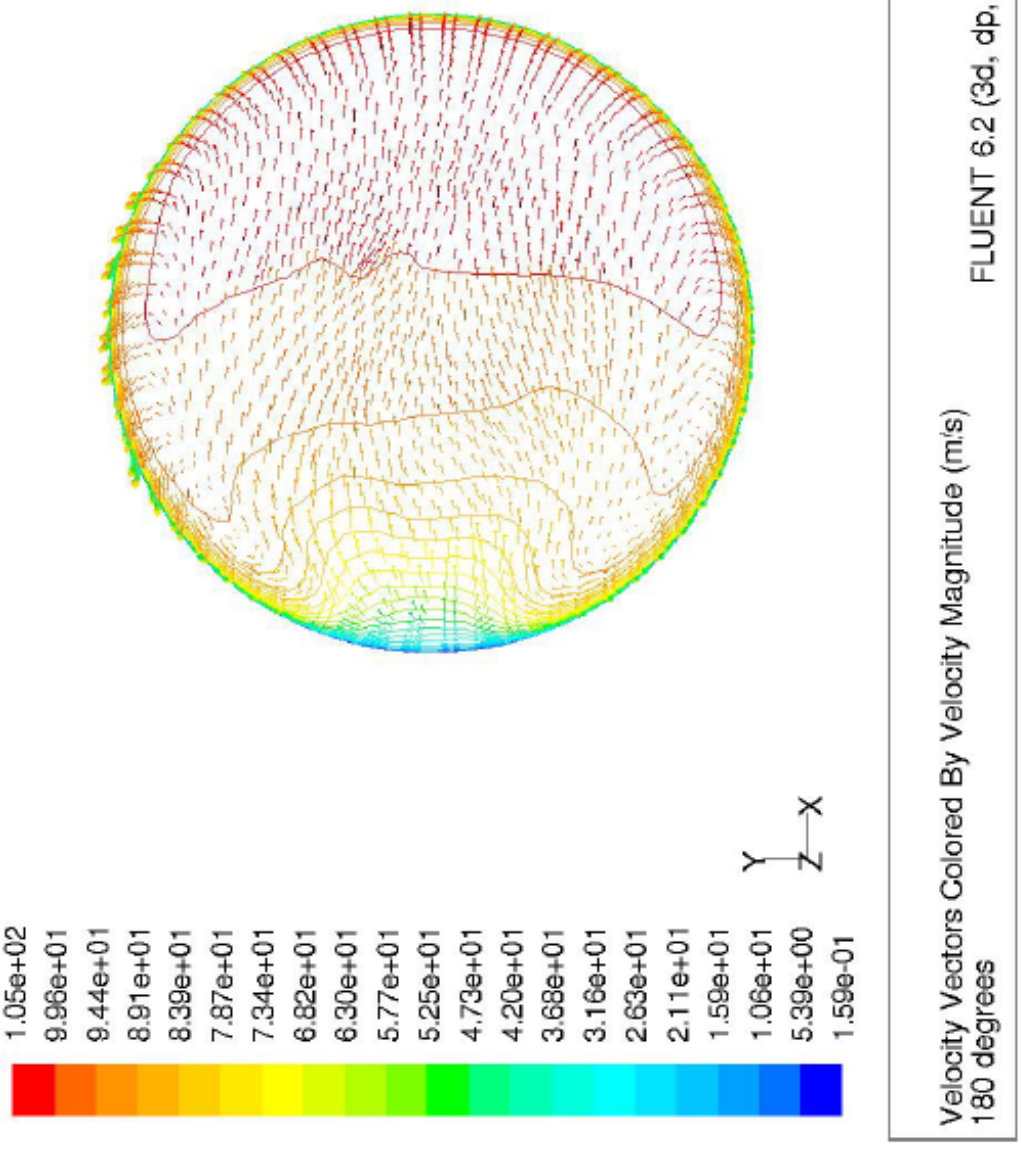

Velocity Vectors Colored By Velocity Magnitude (m/s)
Dec 27, 2006
FLUENT 6.2 (3d, dp, segregated, rke)
9.88e+01
9.40e+01
9.08e+01
8.76e+01
8.44e+01
8.12e+01
7.80e+01
7.48e+01
7.16e+01
6.84e+01
6.53e+01
6.21e+01
5.89e+01
5.57e+01
5.25e+01
4.93e+01
4.61e+01
4.29e+01
Z
X
Y

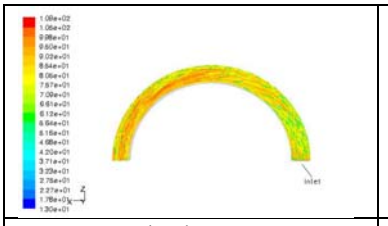
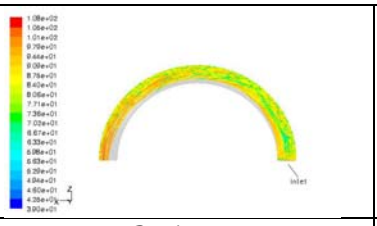
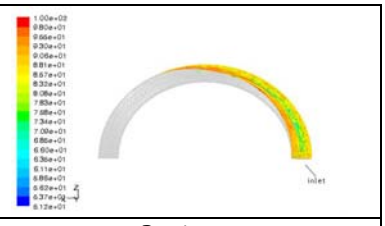
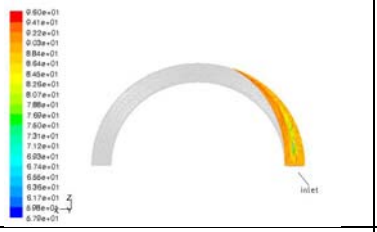
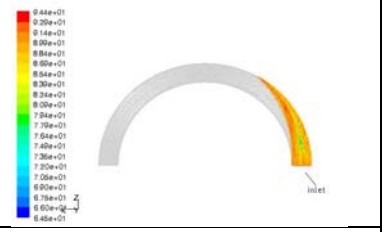
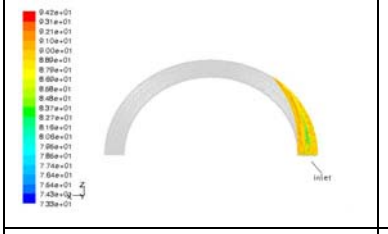
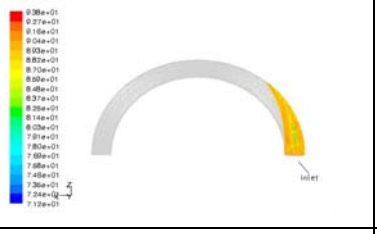
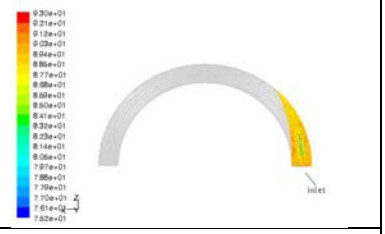
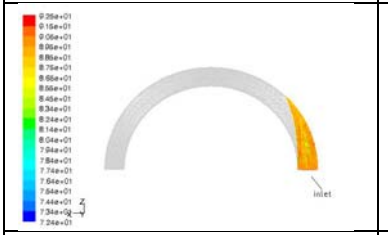
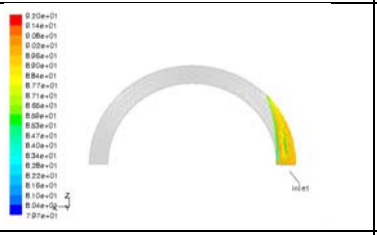

Velocity Vectors Colored By Velocity Magnitude (m/s)
90 degrees
Dec 27, 2006
FLUENT 6.2 (3d, dp, segregated, skw)
9.87e+01
9.39e+01
8.90e+01
8.41e+01
7.92e+01
7.43e+01
6.94e+01
6.46e+01
5.97e+01
5.48e+01
4.99e+01
4.50e+01
4.01e+01
3.53e+01
3.04e+01
2.55e+01
2.06e+01
1.57e+01
1.08e+01
5.95e+00
1.07e+00
Z
X
Y

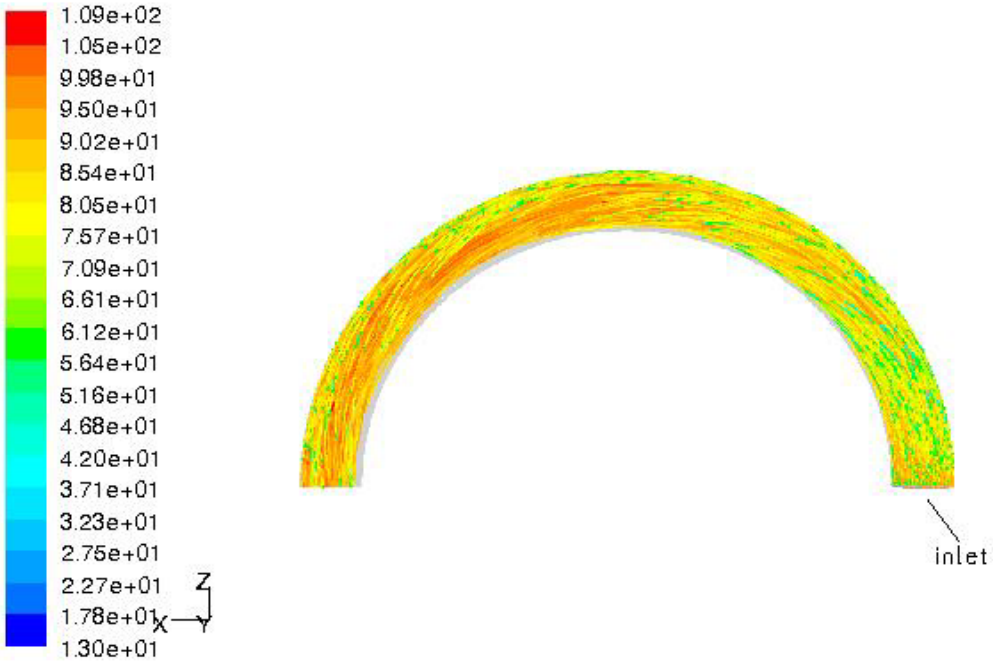

Velocity Vectors Colored By Velocity Magnitude (m/s)
Dec 27, 2006
FLUENT 6.2 (3d, dp, segregated, rke)
9.90e+01
9.66e+01
9.51e+01
9.36e+01
9.20e+01
9.05e+01
8.89e+01
8.74e+01
8.59e+01
8.43e+01
8.28e+01
8.12e+01
7.97e+01
7.82e+01
7.66e+01
7.51e+01
7.35e+01
7.20e+01
Y
X
Z


1.00e+02
9.52e+01
9.02e+01
8.53e+01
8.03e+01
7.53e+01
7.03e+01
6.53e+01
6.04e+01
5.54e+01
5.04e+01
4.54e+01
4.04e+01
3.55e+01
3.05e+01
2.55e+01
2.05e+01
1.55e+01
1.06e+01
5.58e+00
5.98e-01
Y
Z
X
Velocity Vectors Colored By Velocity Magnitude (m/s)
135 degrees
Dec 27, 2006
FLUENT 6.2 (3d, dp, segregated, skw)

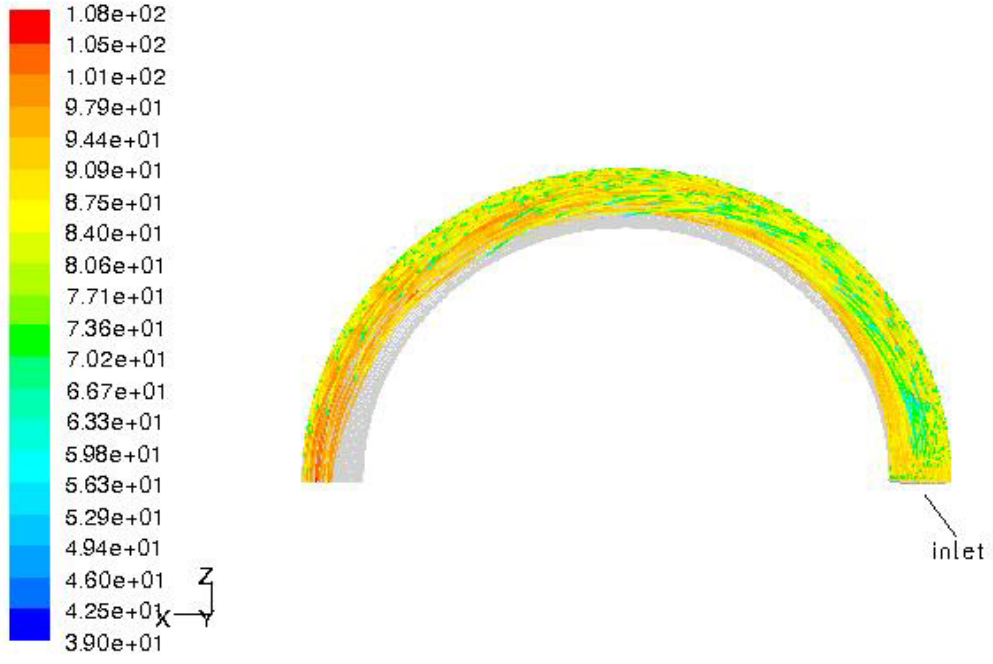

1.05e+02
9.83e+01
9.41e+01
8.98e+01
8.55e+01
8.13e+01
7.70e+01
7.28e+01
6.85e+01
6.43e+01
6.00e+01
5.57e+01
5.15e+01
4.72e+01
4.30e+01
3.87e+01
3.44e+01
3.02e+01
Y
Z
X
Velocity Vectors Colored By Velocity Magnitude (m/s)
Dec 27, 2006
FLUENT 6.2 (3d, dp, segregated, rke)


1.05e+02
9.96e+01
9.44e+01
8.91e+01
8.39e+01
7.87e+01
7.34e+01
6.82e+01
6.30e+01
5.77e+01
5.25e+01
4.73e+01
4.20e+01
3.68e+01
3.16e+01
2.63e+01
2.11e+01
1.59e+01
1.06e+01
5.39e+00
1.59e-01
Y
Z
X
Velocity Vectors Colored By Velocity Magnitude (m/s)
180 degrees
Dec 27, 2006
FLUENT 6.2 (3d, dp, segregated, skw)

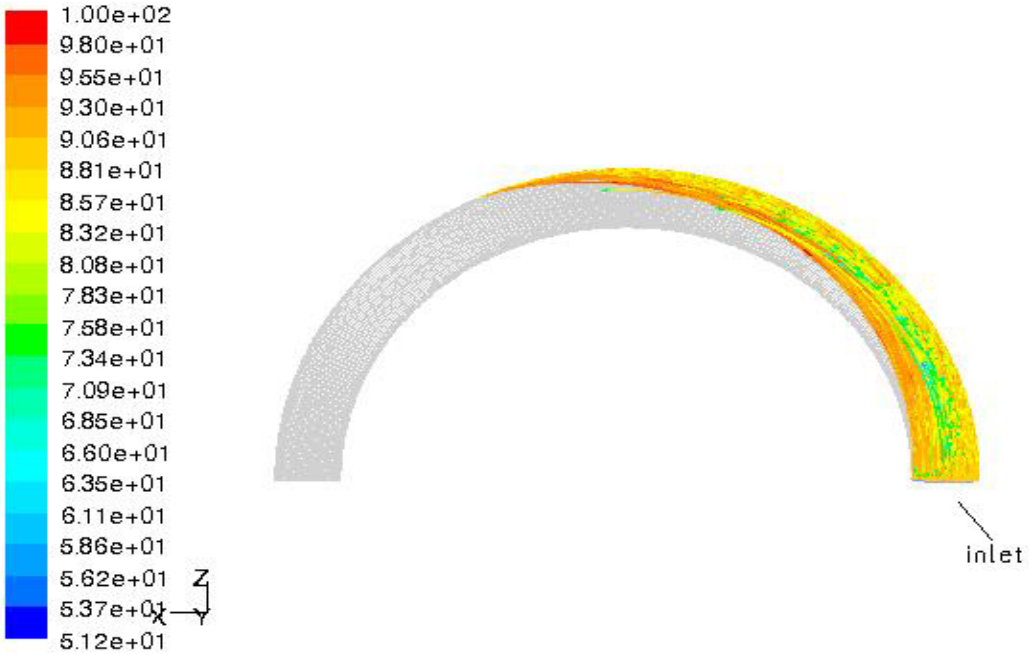
		
1 micron	2 micron	3 micron
		
4 micron	5 micron	6 micron
		
7 micron	8 micron	9 micron
		
10 micron	11 micron	12 micron



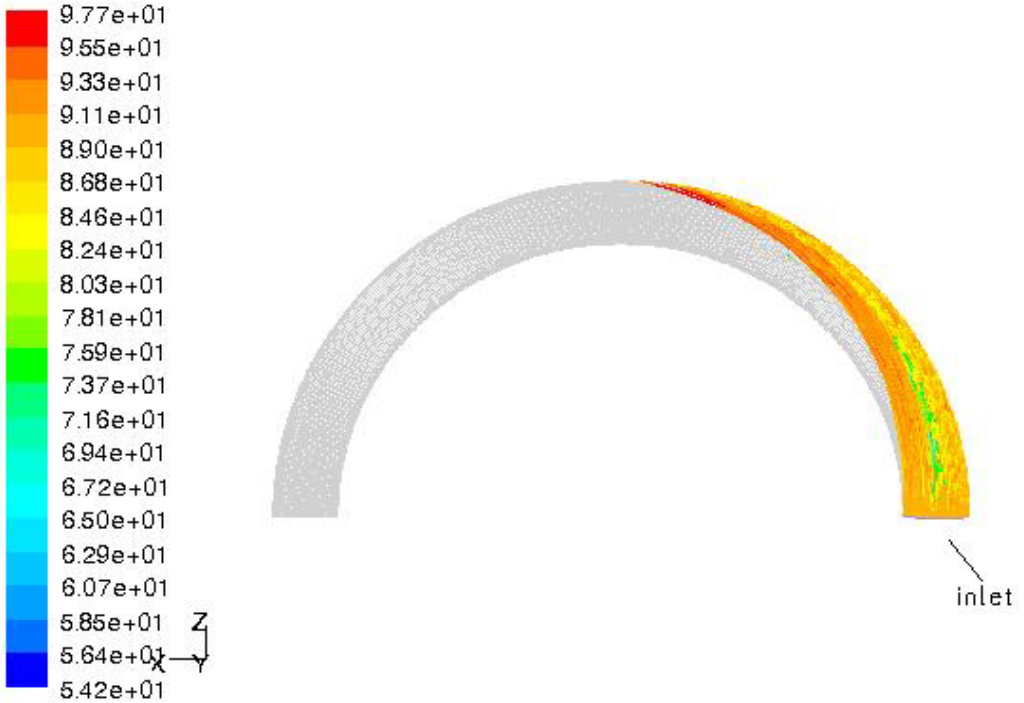
1 micron



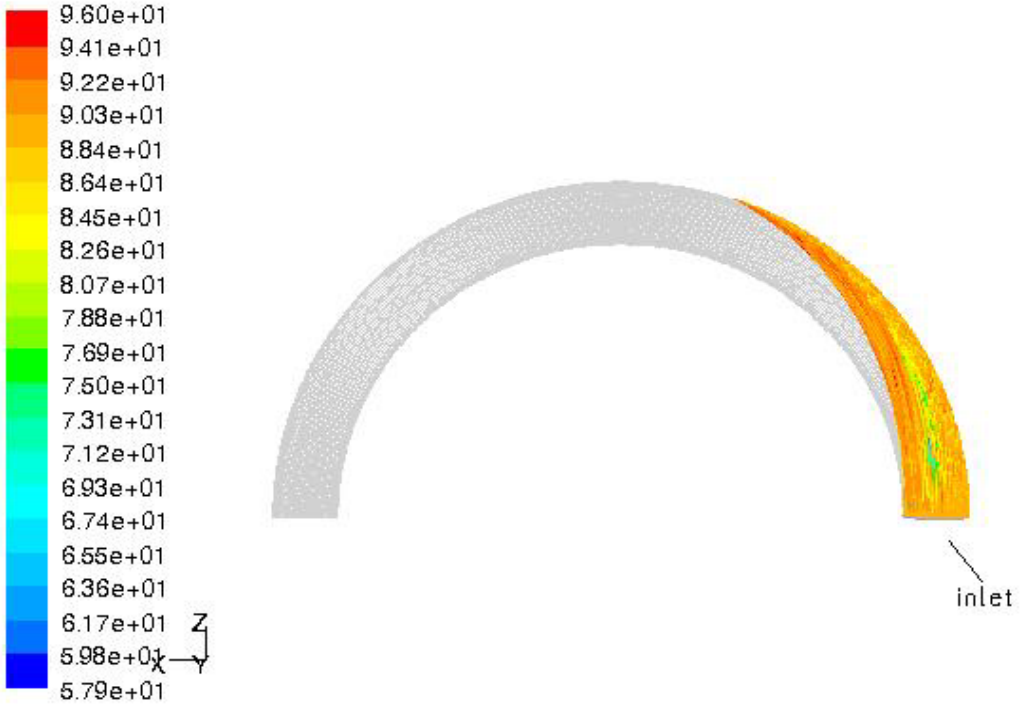
2 micron



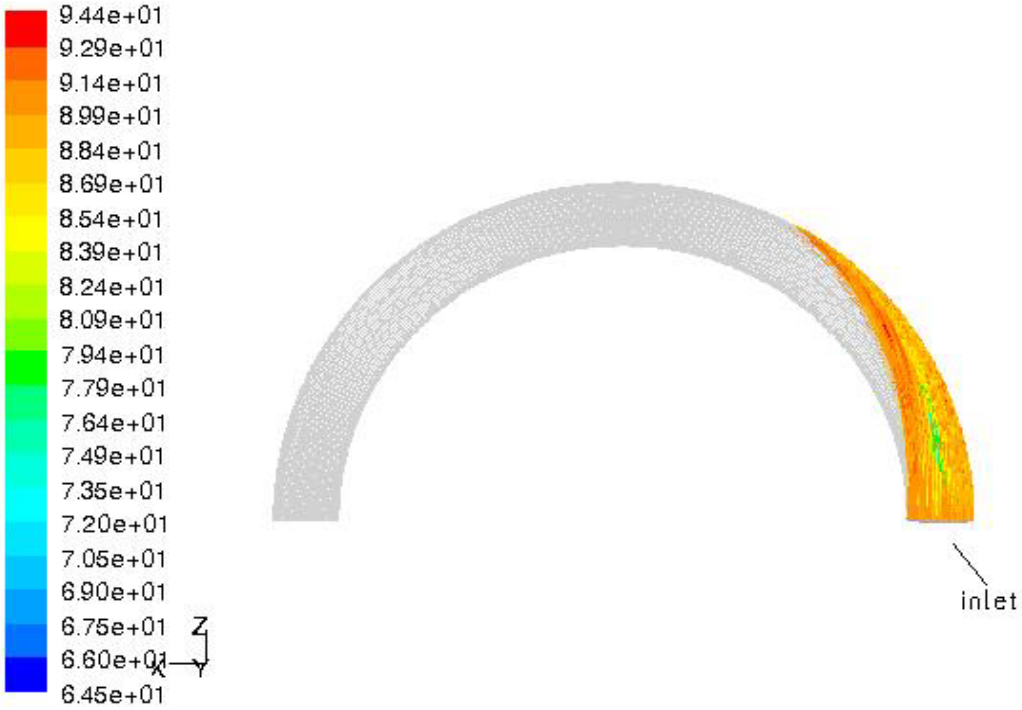
3 micron



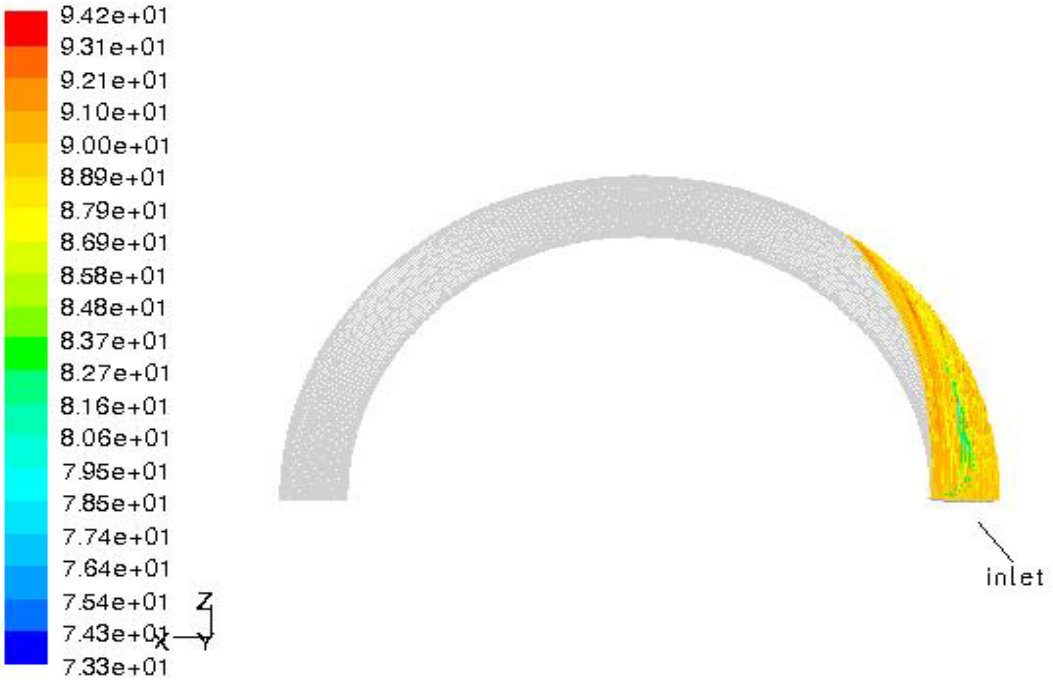
4 micron



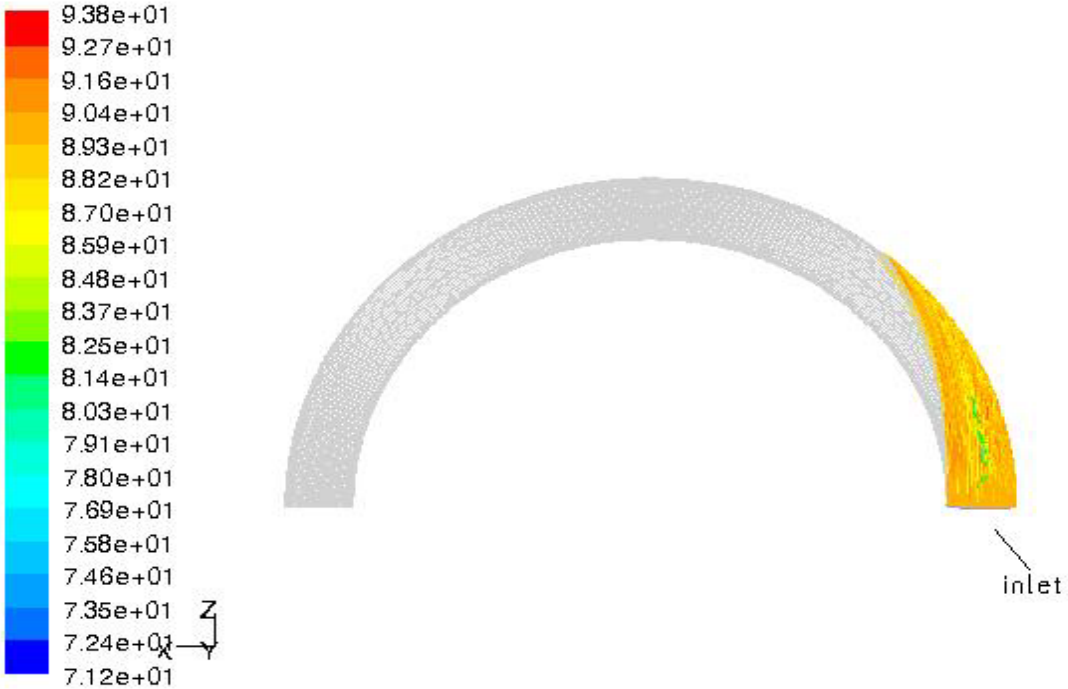
5 micron



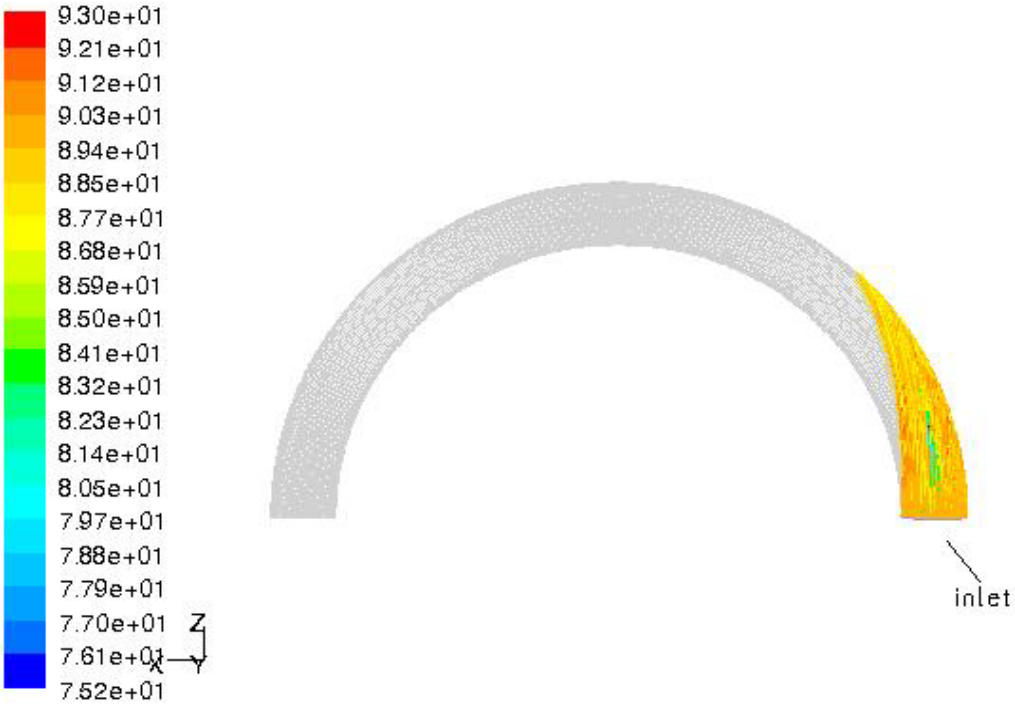
6 micron



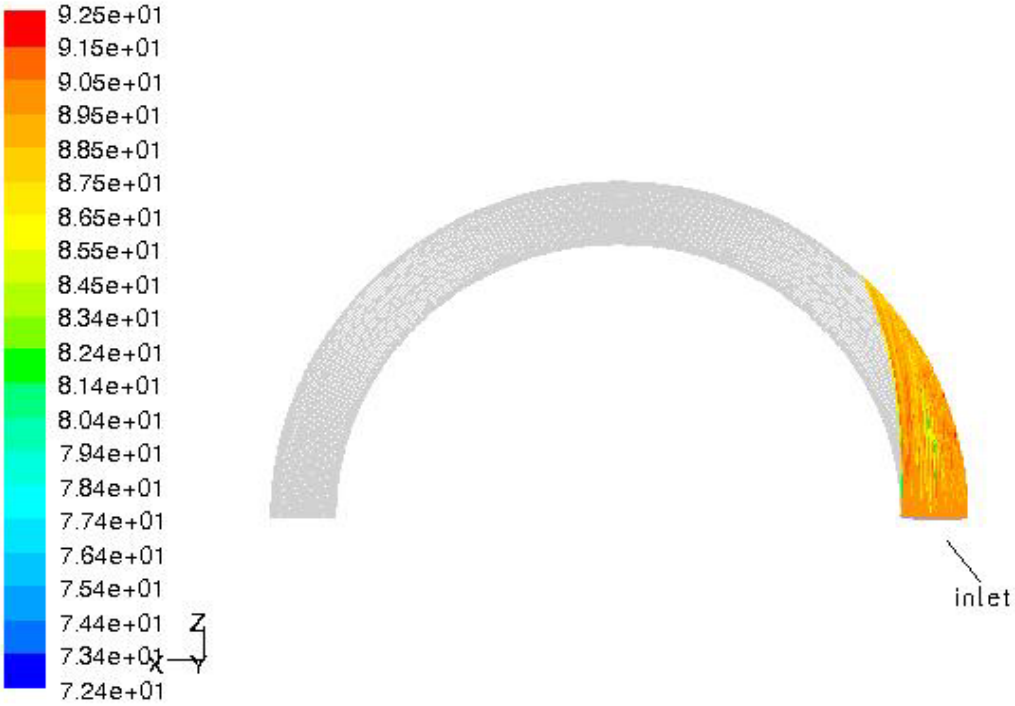
7 micron



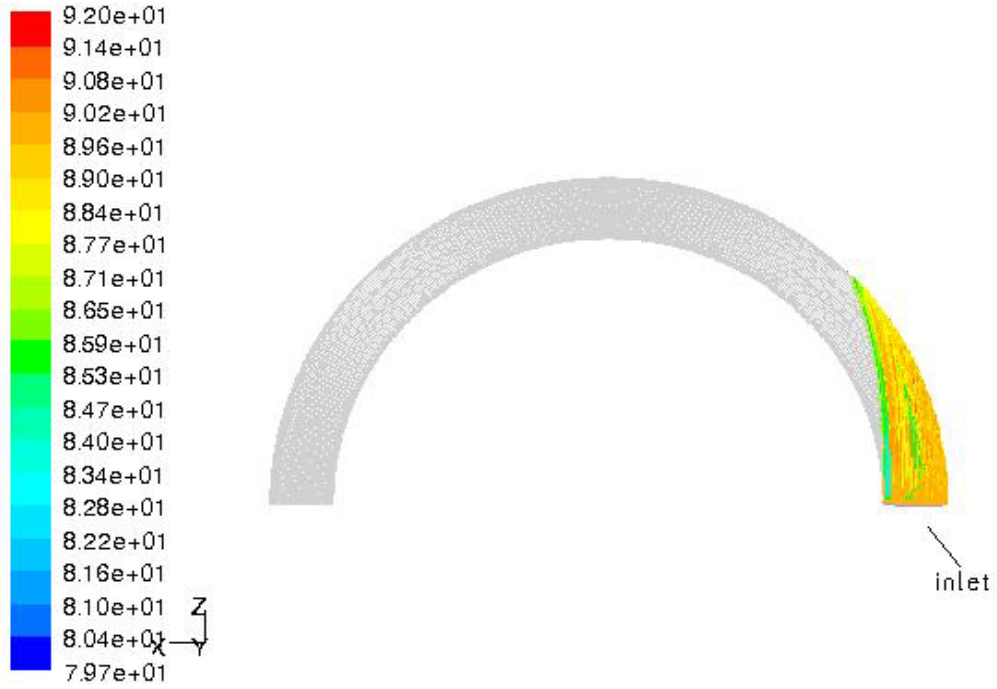
8 micron



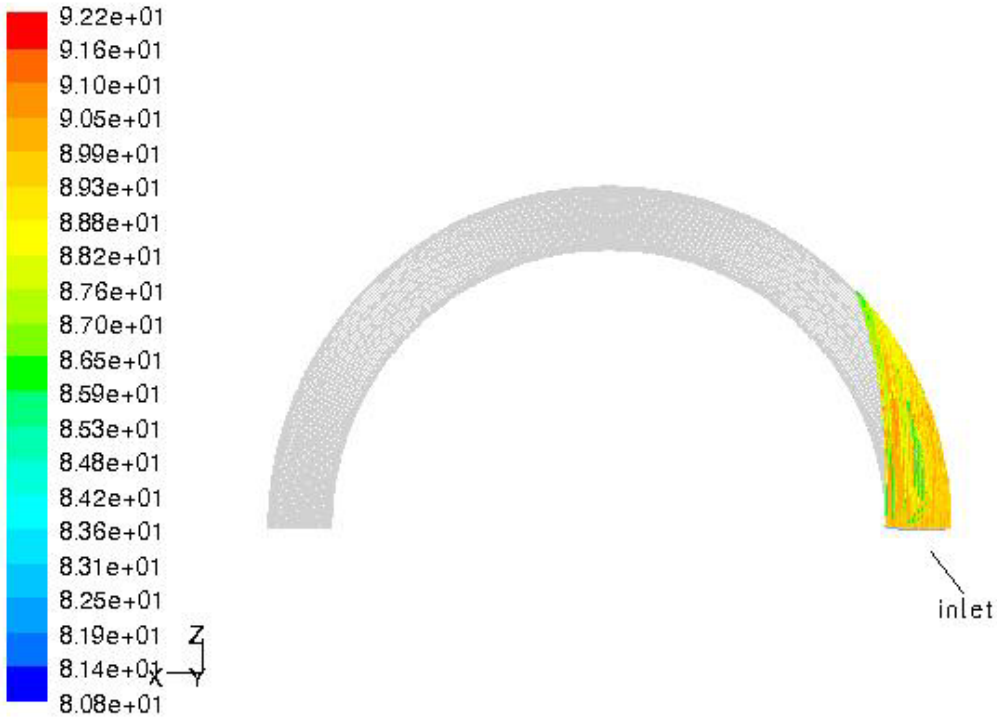
9 micron



10 micron



11 micron



12 micron

VITA

Pamela Longmire received a Bachelors of Science in Nuclear Engineering from the University of Cincinnati, Masters of Science in Engineering Geology (Geochemistry) from the University of Akron, and a Masters of Science in Nuclear Engineering from The Ohio State University. In January 1999, she enrolled at Texas A&M University located in College Station, Texas to pursue her Ph.D. in Nuclear Engineering. In June 2000, after completing the requisite course work for the doctorate she conducted research at the National Aeronautics and Space Administration (NASA), Glenn Research Center. Her research involved development of numerical models describing particle dynamics for use in Digital Particle Imaging Velocimetry. Remaining housed in the Nuclear Engineering Department, she switched to Interdisciplinary Engineering to accommodate her need to explore other areas of interest, which included for example: optics, biothermomechanics, respiratory physiology, and turbulence. To gain practical experience relative to numerical modeling of particle deposition in the human respiratory tract she accepted an invitation in June 2001 to conduct research at Lovelace Respiratory Research Institute located in Albuquerque, New Mexico. In October 2001, she joined Sandia National Laboratories where she continues employment in the Nuclear and Risk Technologies Center, Modeling & Analysis Department.

Address all correspondence to the author:

Pamela Longmire, Ph.D.
10704 Habanero Way SE
Albuquerque, NM 87123-4188