

Utah State University

DigitalCommons@USU

All Graduate Theses and Dissertations

Graduate Studies

12-2011

Evaluation of an Incompressible Energy-Vorticity Turbulence Model for Fully Rough Pipe Flow

Doug F. Hunsaker
Utah State University

Follow this and additional works at: <https://digitalcommons.usu.edu/etd>

 Part of the [Mechanical Engineering Commons](#)

Recommended Citation

Hunsaker, Doug F., "Evaluation of an Incompressible Energy-Vorticity Turbulence Model for Fully Rough Pipe Flow" (2011). *All Graduate Theses and Dissertations*. 1068.
<https://digitalcommons.usu.edu/etd/1068>

This Dissertation is brought to you for free and open access by the Graduate Studies at DigitalCommons@USU. It has been accepted for inclusion in All Graduate Theses and Dissertations by an authorized administrator of DigitalCommons@USU. For more information, please contact digitalcommons@usu.edu.



EVALUATION OF AN INCOMPRESSIBLE ENERGY-VORTICITY
TURBULENCE MODEL FOR FULLY ROUGH PIPE FLOW

by

Douglas F. Hunsaker

A dissertation submitted in partial fulfillment
of the requirements for the degree

of

DOCTOR OF PHILOSOPHY

in

Mechanical Engineering

Approved:

Dr. Warren F. Phillips
Major Professor

Dr. Robert E. Spall
Committee Member

Dr. Christine Hailey
Committee Member

Dr. Steve L. Folkman
Committee Member

Dr. Jan J. Sojka
Committee Member

Dr. Mark R. McLellan
Vice President for Research and
Dean of the School of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2011

Copyright © Douglas F. Hunsaker 2011

All Rights Reserved

ABSTRACT

Evaluation of an Incompressible Energy-Vorticity
Turbulence Model for Fully Rough Pipe Flow

by

Douglas F. Hunsaker, Doctor of Philosophy

Utah State University, 2011

Major Professor: Dr. Warren Phillips
Department: Mechanical and Aerospace Engineering

Traditional methods of closing the Boussinesq-based Reynolds-averaged Navier-Stokes equations are considered, and suggestions for improving two-equation turbulence models are made. The traditional smooth-wall boundary conditions are shown to be incorrect, and the correct boundary conditions are provided along with sample solutions to traditional models. The correct boundary condition at a smooth wall for dissipation-based turbulence models is that which forces both the turbulent kinetic energy and its first derivative to zero. Foundations for an energy-vorticity model suggested by Phillips are presented along with the near-smooth-wall behavior of the model. These results show that at a perfectly smooth wall, the turbulent kinetic energy may approach the wall at a higher order than is generally accepted. The foundations of this model are used in the development of a k - λ model for fully rough pipe flow. Closure coefficients for the model are developed through gradient-based optimization techniques. Results of the model are compared to results from the Wilcox 1998 and 2006 k - ω models as well as four eddy-viscosity models. The results show that the Phillips k - λ model is much more accurate than other models for predicting the relationship between Reynolds number and friction factor for fully rough pipe flow. However, the velocity profiles resulting from the model deviate noticeably from the law of the wall.

(445 pages)

ACKNOWLEDGMENTS

First I would like to sincerely thank my major professor, Dr. Warren F. Phillips, for his time and dedication to this work and also for his commitment to my learning. His one-on-one mentoring has been priceless and his expectations have required me to reach beyond my original aspirations. I will never forget the hours spent in his office drinking from his great well of knowledge and experience. I thank him for requiring genuine accomplishment and understanding from me throughout the learning process.

Although I cannot list all of the many students who have been of support here at USU, I'd like to mention just a few by name. I would like to thank Kyle Horne for his continual flow of computational support and fresh perspectives, Emilie Garbi for her hard work and dedication to this research, Michal Hradisky for being a sounding board of ideas, Zhuorui Song for his mathematical expertise, and many others for their friendship and support. I'd like to especially thank Nick Alley who encouraged me to pursue this opportunity without even knowing me. I would also like to extend special thanks to Dr. Deryl Snyder who believed in me enough to make this opportunity possible in the first place. Finally, I'd like to thank my supportive and loving family members and close friends who have encouraged and stood by me throughout the process.

Douglas F. Hunsaker

CONTENTS

	Page
ABSTRACT	iii
ACKNOWLEDGMENTS	iv
LIST OF TABLES	xiii
LIST OF FIGURES	xiv
LIST OF ACRONYMS	xx
NOMENCLATURE	xxi
CHAPTER	
1. TRADITIONAL TURBULENCE MODELING	1
I. Introduction	1
II. Governing Equations of Fluid Motion	5
A. Conservation of Mass: The Continuity Equation	6
B. Conservation of Momentum: Newton's Second Law	6
C. Conservation of Energy: First Law of Thermodynamics	8
D. Application of Newtonian and Continuum Assumptions	9
III. The Reynolds-Averaged Navier-Stokes Equations	11
IV. Traditional Turbulence Closure	14
A. Reynolds Stress Modeling	14
B. The Boussinesq Hypothesis	15
C. Turbulent Kinetic Energy Transport	18
D. Turbulent Dissipation Transport	20
E. Length and Velocity Scales	21
F. Resulting Energy-Dissipation Turbulence Models	23
1. The k - ε Model	23
2. The k - ω Model	24
3. The k - ζ Model	25
V. Closure Coefficient Evaluation	26
A. The Log Layer	27
B. Homogeneous, Isotropic Turbulence	29
C. The Log Layer at Separation	30
D. Shear-Free Mixing Layer	30
VI. Common Turbulence Model Evaluation Cases	30

	A.	Boundary Layer Flow	31	
	B.	Fully Developed Channel Flow	33	
	C.	Fully Developed Pipe Flow	34	
	D.	Plane Jet Flow	36	
	E.	Round Jet Flow	38	
VII.		Summary and Conclusions	39	
2.		POSSIBLE IMPROVEMENTS IN RANS-BASED TURBULENCE MODELING	41	
	I.	Introduction	41	
	II.	Traditional Smooth-Wall Boundary Conditions	42	
	III.	Turbulent Energy Transport	43	
	IV.	Turbulent Length Scale.....	47	
	V.	Phillips Energy-Vorticity Model	48	
		A.	RMS Turbulent Vorticity Closure: A $k-\tilde{\omega}$ Model.....	49
		B.	Solenoidal Dissipation Closure: A $k-\hat{\epsilon}$ Model.....	50
		C.	DNS Solenoidal Dissipation Closure: A $k-\hat{\epsilon}$ Model.....	51
		D.	General Enstrophy Closure: A $k-\zeta$ Model.....	51
		E.	Robinson Enstrophy-Transport Closure: A $k-\zeta$ Model	52
		F.	Mean Vortex Wavelength Closure: A $k-\lambda$ Model	53
	VI.	Closing Transport Equations	54	
	VII.	Rough-Wall Turbulence Modeling	55	
	VIII.	Summary and Conclusions	57	
3.		OUTLINE OF THE PRESENT WORK.....	58	
4.		COMPUTATIONAL FLUID DYNAMICS METHODS	60	
	I.	Introduction	60	
	II.	One-Dimensional Finite-Difference Formulation	60	
	III.	One-Dimensional Finite-Volume Formulation	62	
	IV.	Two-Dimensional Finite-Volume RANS Solver	63	
		A.	Coordinate System	63
		B.	Transport Equation Discretization	64
		C.	Pressure Coupling	64
		D.	Solution Procedure.....	65
		E.	Code Validation.....	66
			1. Laminar Lid-Driven Cavity	66
			2. Order of Convergence	69
			3. Turbulent Fully Developed Channel Flow	70
	V.	One-Dimensional Runge-Kutta Direct Integration.....	72	
	VI.	Summary and Conclusions	73	
5.		SMOOTH-WALL BOUNDARY CONDITIONS FOR DISSIPATION-BASED MODELS	74	
	I.	Introduction	74	

		vii
	II. Physics of Smooth-Wall Boundary Conditions	75
	III. Smooth-Wall Boundary Conditions for the $k-\varepsilon$ Model	76
	A. The Launder-Sharma $k-\varepsilon$ Model.....	77
	B. The Lam-Bremhorst $k-\varepsilon$ Model.....	80
	IV. Numerical Results from CFD Algorithms	87
	V. Smooth-Wall Boundary Conditions for the $k-\omega$ Model	94
	VI. Comments on CFD Algorithms and Solutions	99
	VII. Summary and Conclusions	101
6.	PHILLIPS TWO-EQUATION MODEL CHARACTERISTICS	102
	I. Introduction	102
	II. RMS Turbulent Vorticity Closure: A $k-\tilde{\omega}$ Model	102
	A. Homogeneous, Isotropic Turbulence	104
	B. Smooth-Wall Behavior.....	104
	C. A Closed-Form Analog.....	110
	III. Enstrophy Closure: A $k-\zeta$ Model	115
	A. Homogeneous, Isotropic Turbulence	116
	B. Smooth-Wall Behavior.....	116
	C. A Closed-Form Analog.....	119
	1. Perfectly Smooth Wall	119
	2. Rough Wall	123
	IV. Summary and Conclusions	124
7.	DEVELOPMENT OF THE PHILLIPS $k-\lambda$ MODEL FOR FULLY ROUGH PIPE FLOW ...	126
	I. Introduction	126
	II. Rough Pipe Flow	127
	A. Fundamental Relations and Definitions	127
	B. Mixing-Length Theory.....	129
	C. The Nikuradse Number	130
	D. The Colebrook Equation	133
	E. Velocity Profile	138
	F. Turbulent Eddy Viscosity	139
	III. The Phillips $k-\lambda$ Model for Fully Rough Pipe Flow	144
	A. Core Reynolds Number.....	145
	B. Mean Vortex Wavelength Distribution.....	146
	C. Model Summary	147
	D. Solution Process	149
	E. Initial Results.....	151
	IV. Summary and Conclusions	153

	viii
8. PHILLIPS k - λ MODEL CLOSURE COEFFICIENT EVALUATION	155
I. Introduction	155
II. The Influence of C_λ for Fully Rough Pipe Flow	155
III. Computer Optimization	157
A. Model Fitness	158
B. Common Optimization Algorithms.....	160
C. Broyden-Fletcher-Goldfarb-Shanno Update Method	162
D. An Example Optimization Run	164
E. Resulting Closure Coefficients	166
IV. Summary and Conclusions	173
9. PHILLIPS k - λ MODEL RESULTS AND CONCLUSIONS.....	175
I. Introduction	175
II. Model Results at Points of Interest	175
A. Low Value on the Flat: $\sigma_k = 0.1$	175
B. High Value on the Flat: $\sigma_k = 4$	177
C. Point of Interest on the Flat: $\sigma_k = 0.75$	179
D. Point of Minimum % RMS Error.....	182
III. Comparison to Other Models	185
A. Wilcox k - ω Model.....	185
B. Eddy-Viscosity Models.....	187
IV. Summary and Conclusions	192
REFERENCES	195
APPENDICES	202
A. MATHEMATICAL IDENTITIES.....	203
I. Vector Identities	203
II. Ensemble Averaging Identities	203
III. Flowfield Properties	203
B. TRADITIONAL TURBULENCE MODELS.....	205
I. The k - ε Model	205
A. The General k - ε Model Equations.....	205
B. Fully Developed Channel Flow	206
C. Fully Developed Pipe Flow.....	207
D. Sample Models.....	209
1. The Launder-Sharma k - ε Model	209
2. The Lam-Bremhorst k - ε Model	214

II.	The k - ω Model.....	221
	A. The General k - ω Model Equations.....	221
	B. Fully Developed Channel Flow	222
	C. Fully Developed Pipe Flow.....	223
	D. Sample Models.....	225
	1. The Traditional k - ω Model	225
	2. The Wilcox 1998 k - ω Model	229
C.	COMMON CASE STUDIES	235
	I. Boundary Layer Flow over an Infinite Flat Plate	235
	A. Case Description	235
	B. Laminar Flow	237
	C. Turbulent Flow	239
	II. Fully Developed Channel Flow	239
	A. Case Description	239
	B. The Continuity and RANS Equations	241
	C. Laminar Flow	244
	D. Turbulent Flow	245
	III. Fully Developed Pipe Flow	246
	A. Case Description	246
	B. The Continuity and RANS Equations	247
	C. Laminar Flow	251
	D. Turbulent Flow	253
	IV. Plane Jet Flow	253
	A. Case Description	253
	B. Laminar Flow	256
	V. Round Jet Flow	260
	A. Case Description	260
	B. Laminar Flow	261
D.	ONE-DIMENSIONAL FINITE-DIFFERENCE FORMULATIONS	266
	I. A Second-Order Formulation	266
	II. Higher-Order Formulations	273
	A. Motivation	273
	B. An n^{th} -Order First Derivative Approximation	274
	C. An n^{th} -Order Second Derivative Approximation	275
	D. Uniform-Grid Tabulated Approximations	278
	E. Example Results	282

	x
III. Sample Code	289
E. ONE-DIMENSIONAL FINITE-VOLUME FORMULATION	299
I. Governing Equations	299
II. Coordinate Transformation.....	299
III. Discretization	300
IV. Transport Equations	303
A. x -Momentum	303
B. Turbulent Kinetic Energy	305
C. Second Turbulence Variable, h	307
V. Solution Procedure.....	310
VI. Sample Results.....	310
A. Laminar Flow	310
B. A Second-Order Test Case.....	311
C. A Closed-Form k - ϵ Analogy	311
VII. Sample Code	312
F. COORDINATE TRANSFORMATIONS IN CARTESIAN COORDINATES.....	321
I. Transformation Definitions	321
II. Scalar Transport Coordinate Transformations	323
A. General Scalar Transport.....	323
B. Source Terms.....	323
1. Continuity	324
2. x -Momentum	325
3. y -Momentum	325
4. Strain-Rate Tensor	325
5. Rotation Tensor	326
G. COORDINATE TRANSFORMATIONS IN CYLINDRICAL COORDINATES.....	327
I. Transformation Definitions	327
II. Scalar Transport Coordinate Transformations	329
A. General Scalar Transport.....	329
B. Source Terms.....	330
1. Continuity	330
2. Momentum	330
3. Strain-Rate Tensor	332
H. FINITE-VOLUME DISCRETIZATION.....	333
I. Domain Discretization	333
II. Discretization Schemes	336

	xi
A.	Differencing 336
1.	Cell Centers 337
2.	Cell Faces 338
B.	Integration 340
C.	Approximations in Two Dimensions 340
1.	Cell Centers 340
2.	Cell Faces 341
3.	Integration 344
I.	FINITE-VOLUME METHOD IN CARTESIAN COORDINATES 345
I.	General Scalar Transport 345
II.	Boundary Condition Implementation 347
A.	The Dirichlet Boundary Condition 348
B.	The Neumann Boundary Condition 349
C.	Velocity Inlet 352
D.	Outlet 352
E.	Smooth, No-Slip Wall 353
F.	Symmetry 353
G.	Pressure 354
III.	Transport Equations 355
A.	Momentum 355
B.	Turbulent Kinetic Energy 357
C.	Dissipation 358
D.	Dissipation Frequency 358
J.	FINITE-VOLUME METHOD IN CYLINDRICAL COORDINATES 359
I.	General Scalar Transport 359
II.	Boundary Condition Implementation 361
A.	The Dirichlet Boundary Condition 362
B.	The Neumann Boundary Condition 363
III.	Transport Equations 368
K.	THE SIMPLE ALGORITHM IN CARTESIAN COORDINATES 371
I.	Development of the SIMPLE Algorithm 371
II.	Boundary Treatment 375
L.	THE SIMPLE ALGORITHM IN CYLINDRICAL COORDINATES 376
I.	Development of the SIMPLE Algorithm 376
II.	Boundary Treatment 380
M.	VORTICITY TRANSPORT 381

I.	Laminar Flow Algorithm	381
II.	Turbulent Flow Algorithm	382
	A. The Ensemble Average Vorticity Transport Equation.....	382
	B. The RANS Ensemble Average Vorticity Transport Equation	383
N.	ONE-DIMENSIONAL DIRECT INTEGRATION METHOD.....	385
	I. The k - ε Model Equations	385
	II. The k - ω Model Equations	387
O.	CODE FOR SOLVING THE PHILLIPS k - λ MODEL.....	391
P.	OPTIMIZATION CODE	403
Q.	CLOSURE COEFFICIENT EVALUATION ON THE FLAT	410
	CURRICULUM VITAE	411

LIST OF TABLES

Table	Page
8.1 Sample results for the Phillips k - λ fully rough pipe flow model given a set of non-optimal model constants.....	159
8.2 Example optimization iterations using the BFGS update method.....	165
D.1 Second-order approximation for the first derivative	278
D.2 Second-order approximation for the second derivative.....	278
D.3 Fourth-order approximation for the first derivative.....	279
D.4 Fourth-order approximation for the second derivative.....	279
D.5 Sixth-order approximation for the first derivative.....	280
D.6 Sixth-order approximation for the second derivative	280
D.7 Eighth-order approximation for the first derivative.....	281
D.8 Eighth-order approximation for the second derivative.....	282
F.1 Transport equation terms	324

LIST OF FIGURES

Figure	Page
1.1	Boundary layer case description 32
1.2	Fully developed channel flow case description 33
1.3	Fully developed pipe flow case description..... 35
1.4	Plane jet flow case description..... 37
1.5	Round jet flow case description..... 38
2.1	Correlation between friction coefficient and roughness Reynolds number by Nikuradse [59] 56
4.1	Grid convergence results for Wilcox 1998 $k-\omega$ model for fully developed pipe flow 61
4.2	Grid-converged results of codes written by Hunsaker and Wilcox for the Wilcox 1998 $k-\omega$ model for fully developed pipe flow 61
4.3	Grid-converged results of four models for the case of fully developed pipe flow 62
4.4	ICESS grid refinement results of the x -velocity profile along the vertical centerline of the lid-driven cavity at a Reynolds number of 100. 66
4.5	Fluent grid refinement results of the x -velocity profile along the vertical centerline of the lid-driven cavity at a Reynolds number of 100. 67
4.6	OpenFOAM grid refinement results of the x -velocity profile along the vertical centerline of the lid-driven cavity at a Reynolds number of 100. 68
4.7	Richardson extrapolation results for the x -velocity profile along the vertical centerline of the lid-driven cavity. 68
4.8	Finite volume algorithm RMS error as a function of average cell size squared 69
4.9	Grid resolution for the mean velocity predicted by the Wilcox 1998 $k-\omega$ model using the one-dimensional finite-volume method 70
4.10	Grid resolution for the mean velocity predicted by the Wilcox 1998 $k-\omega$ model using the two-dimensional finite-volume method 71
4.11	Comparison of results from the one- and two-dimensional finite-volume methods and a finite-difference method for the mean velocity predicted by the traditional $k-\omega$ model. 71
4.12	Comparison of results from the one- and two-dimensional finite-volume methods and a finite-difference method for the mean velocity predicted by the Wilcox 1998 $k-\omega$ model..... 72

		xv
5.1	Solutions to Eq. (5.31) with randomly selected wall boundary conditions	86
5.2	Solutions to Eq. (5.31) with no slip and no dissipation gradient at the wall.	87
5.3	Grid resolution for the mean velocity predicted from the Launder-Sharma $k-\varepsilon$ model.....	88
5.4	Grid resolution for the turbulent energy predicted from the Launder-Sharma $k-\varepsilon$ model.....	89
5.5	Grid resolution for the turbulent dissipation predicted from the Launder-Sharma $k-\varepsilon$ model	89
5.6	Effects of wall boundary conditions on turbulent energy predicted from the Lam-Bremhorst model.	90
5.7	Effects of wall boundary conditions on turbulent energy predicted from the Launder-Sharma model.....	90
5.8	Effects of wall boundary conditions on near-wall dissipation for Launder-Sharma model.....	92
5.9	Effects of wall boundary conditions on the mean velocity predicted from the Launder-Sharma model.....	92
5.10	Effects of wall boundary conditions on turbulent energy predicted from the Wilcox 1998 $k-\omega$ model	99
5.11	Effects of wall boundary conditions on the turbulent dissipation frequency predicted from the Wilcox 1998 $k-\omega$ model	99
6.1	Numerical solution for k in the closed-form analog to the $k-\tilde{\omega}$ model	114
6.2	Numerical solution for $\tilde{\omega}$ in the closed-form analog to the $k-\tilde{\omega}$ model.....	115
7.1	Nikuradse number as a function of roughness Reynolds number for experimental data of Nikuradse [59]	132
7.2	Nikuradse number as a function of roughness Reynolds number	134
7.3	The Moody chart with experimental data from Nikuradse and Shockling, et al.....	137
7.4	Velocity profiles in rough pipes at high Reynolds numbers.....	139
7.5	Eddy-viscosity profiles in fully developed pipe flow.....	141
7.6	Initial velocity profile results for the Phillips $k-\lambda$ model	152
7.7	Initial eddy-viscosity profile results for the Phillips $k-\lambda$ model.....	152
7.8	Initial friction factor results for the Phillips $k-\lambda$ model.....	153
8.1	Sample friction factor results for the Phillips $k-\lambda$ fully rough pipe flow model given a set of non-optimal model constants.....	160
8.2	Friction factor results for the Phillips $k-\lambda$ fully rough pipe flow model	

	with a set of optimal model constants given in Eq. (8.24)	165
8.3	% RMS error over a range of values for C_{r1} and σ_k	166
8.4	Optimal value for C_{r2} as a function of σ_k	168
8.5	Optimal value for a_{r2} as a function of σ_k	169
8.6	Optimal value for C_{r3} as a function of σ_k	169
8.7	Optimal value for a_{r3} as a function of σ_k	170
8.8	Optimal value for a_{r4} as a function of σ_k	170
8.9	% RMS error as a function of σ_k	171
8.10	% RMS error of the final model using coefficients evaluated from the flat over a range of values for C_{r1} and σ_k	172
9.1	Friction factor results for the Phillips k - λ model on the flat with $\sigma_k = 0.1$	176
9.2	Velocity results for the Phillips k - λ model on the flat with $\sigma_k = 0.1$ and $k_s^+ = 1000$	176
9.3	Eddy-viscosity results for the Phillips k - λ model on the flat with $\sigma_k = 0.1$ and $k_s^+ = 1000$	177
9.4	Friction factor results for the Phillips k - λ model on the flat with $\sigma_k = 4$	178
9.5	Velocity results for the Phillips k - λ model on the flat with $\sigma_k = 4$ and $k_s^+ = 1000$	178
9.6	Eddy-viscosity results for the Phillips k - λ model on the flat with $\sigma_k = 4$ and $k_s^+ = 1000$	179
9.7	Friction factor results for the Phillips k - λ model on the flat with $\sigma_k = 0.75$	179
9.8	Velocity results for the Phillips k - λ model on the flat with $\sigma_k = 0.75$ and $k_s^+ = 1000$	180
9.9	Eddy-viscosity results for the Phillips k - λ model on the flat with $\sigma_k = 0.75$ and $k_s^+ = 1000$	180
9.10	Friction factor results for the Phillips k - λ model at the point of minimum % RMS error	182
9.11	Velocity results for the Phillips k - λ model at the point of minimum % RMS error with $k_s^+ = 1000$	183
9.12	Eddy-viscosity results for the Phillips k - λ model at the point of minimum % RMS error with $k_s^+ = 1000$	183
9.13	Friction factor results for the Wilcox 1998 k - ω model	186

9.14	Friction factor results for the Wilcox 2006 k - ω model.....	186
9.15	Nikuradse number results of the Phillips k - λ model with optimum model constants	187
9.16	Velocity results for the eddy-viscosity model given in Eq. (7.35) with $k_s^+ = 1000$	188
9.17	Velocity results for the eddy-viscosity model given in Eq. (7.36) with $k_s^+ = 1000$	188
9.18	Velocity results for the eddy-viscosity model given in Eq. (7.42) with $k_s^+ = 1000$	189
9.19	Velocity results for the eddy-viscosity model given in Eq. (7.43) with $k_s^+ = 1000$	189
9.20	Friction factor results for the eddy-viscosity model given in Eq. (7.35)	190
9.21	Friction factor results for the eddy-viscosity model given in Eq. (7.36)	191
9.22	Friction factor results for the eddy-viscosity model given in Eq. (7.42)	191
9.23	Friction factor results for the eddy-viscosity model given in Eq. (7.43)	192
B.1	Nondimensional mean velocity results from the Launder-Sharma model	211
B.2	Nondimensional turbulent kinetic energy results from the Launder-Sharma model.....	212
B.3	Nondimensional dissipation results from the Launder-Sharma model	212
B.4	Nondimensional mean velocity results from the Lam-Bremhorst model.....	217
B.5	Nondimensional turbulent kinetic energy results from the Lam-Bremhorst model.....	217
B.6	Nondimensional dissipation results from the Lam-Bremhorst model.....	218
B.7	Nondimensional mean velocity results from the traditional k - ω model.....	227
B.8	Nondimensional turbulent kinetic energy results from the traditional k - ω model	228
B.9	Nondimensional dissipation frequency results from the traditional k - ω model	228
B.10	Nondimensional mean velocity results from the Wilcox 1998 model	232
B.11	Nondimensional turbulent kinetic energy results from the Wilcox 1998 model.....	232
B.12	Nondimensional dissipation frequency results from the Wilcox 1998 model.....	233

C.1	Boundary layer case description	236
C.2	Grid resolution results from ICESS for the laminar boundary layer	237
C.3	Grid resolution results from Fluent for the laminar boundary layer	238
C.4	Grid-resolved results from ICESS and Fluent for the laminar boundary layer	238
C.5	Fully developed channel flow case description	240
C.6	Normalized velocity profile for fully developed channel flow	245
C.7	Fully developed pipe flow case description	247
C.8	Normalized velocity profile for fully developed pipe flow	252
C.9	Plane jet flow case description	254
C.10	Numerical results for the nondimensional x -velocity profile for the plane laminar jet	258
C.11	Numerical results for the nondimensional plane laminar jet centerline velocity	259
C.12	Numerical results for the nondimensional plane laminar jet spread rate	259
C.13	Round jet flow case description	261
C.14	Numerical results for the nondimensional z -velocity profile for the round laminar jet	264
C.15	Numerical results for the nondimensional round laminar jet centerline velocity	264
C.16	Numerical results for the nondimensional round laminar jet spread rate	265
D.1	Near-wall results of a second-order finite-difference algorithm	283
D.2	Second-order results using double-precision computations	284
D.3	Second-order results using quad-precision computations	284
D.4	Fourth-order results using double-precision computations	285
D.5	Fourth-order results using quad-precision computations	285
D.6	Eighth-order results using double-precision computations	286
D.7	Eighth-order results using quad-precision computations	287
D.8	Eighth-order results of a tenth-order near-wall equation using quad-precision computations	287
D.9	Eighth-order finite-difference RMS error as a function of grid spacing	288
D.10	Negative near-wall results of an eighth-order approximation to a tenth-order near-wall solution	289

E.1	Comparison of physical and transformed domains	301
H.1	Physical domain discretization and nomenclature.....	334
H.2	Computational domain discretization and nomenclature	334
H.3	Physical domain of an example rectilinear grid with logarithmic spacing.....	335
H.4	Computational domain of an example rectilinear grid with logarithmic spacing	335

LIST OF ACRONYMS

BFGS	Broyden-Fletcher-Goldfarb-Shanno
CFD	Computational Fluid Dynamics
DNS	Direct Numerical Simulation
ICISS	Incompressible Computational Enstrophy Structured Solver
LES	Large Eddy Simulation
RANS	Reynolds-averaged Navier-Stokes
RMS	Root-Mean-Square
SIMPLE	Semi-Implicit Method for Pressure-Linked Equations

NOMENCLATURE

\mathbf{A}	=	acceleration vector
A	=	transport equation discretization coefficient
A	=	coefficient array
A	=	arbitrary constant
A^+	=	empirical constant dependent on wall roughness
$A_{0-\infty}$	=	Taylor-series expansion constants
A_{1-2}	=	closed-form analog model coefficients
A_k	=	constant defined in Eq. (7.75)
a	=	arbitrary constant
a_1	=	empirical constant, $a_1 = 0.15$
a_{r1-r4}	=	turbulence model closure coefficients
B	=	coefficient array
B	=	arbitrary constant
$B_{0-\infty}$	=	Taylor-series expansion constants
B_{1-2}	=	closed-form analog model coefficients
B_k	=	constant defined in Eq. (7.75)
b	=	arbitrary constant
C	=	coefficient array
C	=	arbitrary constant
$C_{0-\infty}$	=	Taylor-series expansion constants
C_{1-5}	=	constants of integration
C_{1-7}	=	turbulence model closure coefficients
C_f	=	friction coefficient defined in Eq. (1.115)
C_{k1-k2}	=	closed-form analog model coefficients

$C_{\ell 0}$	=	constant used in mixing-length-theory eddy-viscosity model, $C_{\ell 0} \approx 0.345$
C_{r1-r3}	=	turbulence model closure coefficients
$C_{\varepsilon 1-\varepsilon 2}$	=	turbulence model closure coefficients
$C_{\hat{\varepsilon} 1-\hat{\varepsilon} 4}$	=	turbulence model closure coefficients
$C_{\zeta 1-\zeta 6}$	=	turbulence model closure coefficients
C_{λ}	=	turbulence model closure coefficient
C_{μ}	=	turbulence model closure coefficient
C_{ν}	=	turbulence model closure coefficient
$C_{\omega 1-\omega 2}$	=	turbulence model closure coefficients
$C_{\tilde{\omega} 1-\tilde{\omega} 4}$	=	turbulence model closure coefficients
c_v	=	constant volume specific heat
D	=	discretization coefficient
D_h	=	hydraulic diameter
E	=	turbulence model damping function dependent on the model
E^+	=	wall-scaled damping function, $E^+ \equiv E \nu^2 / u_{\tau}^6$
e	=	specific energy, defined in Eq. (1.13)
\mathbf{F}	=	force vector
F	=	arbitrary function
F_L	=	deferred correction lower-order approximation
F_H	=	deferred correction higher-order approximation
F_{ε}	=	change of variables as in Eq. (5.33)
\mathbf{f}	=	force vector per unit volume
\mathbf{f}_b	=	body force vector per unit volume
\mathbf{f}_s	=	surface force vector per unit volume
f	=	arbitrary function
f_{1-2}	=	turbulence model damping function dependent on the model

f_D	=	Darcy friction factor
f_{h0}	=	near-wall approximation for h , dependent on the turbulence model
f_{h1-h3}	=	turbulence model damping functions dependent on the model
f_k	=	turbulence model damping function dependent on the model
f_{k1-k3}	=	turbulence model functions dependent on the model
f_u	=	x -momentum source term in one-dimensional finite-volume method
$f_{\varepsilon0}$	=	near-wall approximation for ε , dependent on the turbulence model
$f_{\varepsilon1-\varepsilon2}$	=	turbulence model functions dependent on the model
$f_{\omega0}$	=	near-wall approximation for ω , dependent on the turbulence model
$f_{\omega1-\omega2}$	=	turbulence model functions dependent on the model
f_μ	=	turbulence model damping function dependent on the model
f_ν	=	turbulence model damping function dependent on the model
G_ϕ^m	=	transport property production rate per unit volume
g_o	=	acceleration of gravity at standard sea level
$\bar{\bar{\mathbf{H}}}$	=	Hessian matrix
H	=	nondimensional length
h	=	arbitrary second variable in a two-equation turbulence model
h^+	=	wall-scaled second turbulence variable
h_0	=	near-wall approximation for h , dependent on the turbulence model
$\bar{\bar{\mathbf{J}}}$	=	Jacobian tensor of a vector field defined in Eq. (1.61)
J	=	coordinate transformation scalar, $J \equiv \xi_{,x}\eta_{,y} - \xi_{,y}\eta_{,x}$
K	=	specific axial momentum flux per unit width for plane jet
K	=	specific axial momentum flux for round jet
k	=	turbulent kinetic energy per unit mass defined in Eq. (1.55)
k^+	=	wall-scaled dimensionless turbulent-kinetic-energy, $k^+ \equiv k/u_\tau^2$
k_0	=	value for k at the wall

k_r	=	relative roughness, $k_r \equiv k_s/(2R)$
k_s	=	equivalent surface-roughness height
k_s^+	=	wall-scaled dimensionless surface-roughness height, $k_s^+ \equiv k_s u_\tau/\nu$
k_t	=	thermal conductivity
\hat{k}	=	closed-form analog model variable
L	=	flat plate length
L	=	channel half-width
l	=	channel half-width
l^+	=	wall-scaled dimensionless channel half-width, $l^+ \equiv l u_\tau/\nu$
ℓ	=	mixing length
ℓ^+	=	wall-scaled mixing length, $\ell^+ \equiv \ell u_\tau/\nu$
l_c	=	periodic wave length scale, $l_c \equiv V_c/\omega_c$
l_k	=	turbulent energy characteristic length defined in Eq. (2.21)
l_t	=	turbulent characteristic length
M	=	total momentum flux
\overline{M}_x	=	specific x -momentum flux defined in Eq. (1.123)
m	=	mass
m	=	arbitrary integer
\dot{m}	=	average volume flux
m_h	=	number of nodes implementing near-wall approximation of second transport equation
$\overline{\mathbf{N}}$	=	direction matrix used in optimization algorithm
N_i	=	Nikuradse number defined in Eq. (7.24)
n	=	arbitrary integer
O	=	on the order of
P	=	array of coefficients for pipe-flow algorithm
p	=	thermodynamic pressure

- p^+ = wall-scaled dimensionless pressure gradient, $p^+ \equiv (d\hat{p}/dx)(\nu/\rho u_\tau^3)$
 \hat{p} = pseudo mean pressure defined in Eq. (1.52)
 \hat{p} = pseudo hydrostatic pressure defined in Eq. (2.1)
 $\bar{\hat{p}}$ = mean pseudo hydrostatic pressure defined in Eq. (2.5)
 $\tilde{\hat{p}}$ = fluctuating pseudo hydrostatic pressure defined in Eq. (2.6)
 $\hat{\hat{p}}$ = term defined in Eq. (1.49)
 Q = volume flow rate
 Q_o = volume flow rate at physical origin of round jet
 Q^+ = change of variables
 \mathbf{q} = heat flux vector defined in Eq. (1.20)
 q^+ = change of variables
 R = pipe radius
 R_c = ratio of closure coefficients given in Eq. (6.76)
 R_c = length scale in core region of pipe flow defined in Eq. (7.51)
 R_e = Reynolds number
 R_{ec} = core Reynolds number defined in Eq. (7.52)
 R_L = Reynolds number based on plate length, defined in Eq. (1.114)
 R_t = turbulent Reynolds number, $R_t \equiv k^2/\nu\varepsilon$ for k - ε models, $R_t \equiv k/\nu\omega$ for k - ω models
 R_x = Reynolds number based on x -position
 R_y = cell Reynolds number, $R_y \equiv y\sqrt{k}/\nu$
 R_z = Reynolds number based on z -position
 R_τ = shear Reynolds number, $R_\tau \equiv Lu_\tau/\nu$ for channel flow, $R_\tau \equiv Ru_\tau/\nu$ for pipe flow
 r = radial coordinate measured normal to z -axis
 r^+ = wall-scaled radial coordinate, $r^+ \equiv ru_\tau/\nu$
 $r_{1\%}$ = radial coordinate of round jet at which axial velocity is 1% of the centerline velocity
 r_h = radial coordinate of round jet at which axial velocity is 50% of the centerline velocity

\hat{r}	= pipe-scaled radial coordinate, $\hat{r} \equiv r/R$
\hat{r}_o	= outer root of Eq. (7.41)
$\bar{\bar{\mathbf{S}}}$	= strain rate tensor defined in Eq. (1.19)
S	= transport equation source term
\hat{S}	= integral form of transport equation source term
S_c	= deferred correction source term
S_V^2	= squared magnitude of the strain-rate tensor, $S_V^2 \equiv \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})$
$\bar{\mathbf{s}}$	= search direction used in optimization algorithm
s	= arbitrary scalar
T	= thermodynamic temperature
T	= diagonal in tridiagonal matrix
t	= time
\mathbf{U}	= arbitrary vector
U	= magnitude of \mathbf{U}
U_e^m	= volumetric heating per unit volume
u	= x -component of velocity
u^+	= wall-scaled dimensionless velocity, $u^+ \equiv \bar{V}_x/u_\tau$
u_{bulk}^+	= wall-scaled bulk velocity, $u_{\text{bulk}}^+ \equiv \bar{V}_{\text{bulk}}/u_\tau$
u_c^+	= wall-scaled centerline velocity, $u_c^+ \equiv \bar{V}_c/u_\tau$
u_{max}^+	= wall-scaled maximum velocity, $u_{\text{max}}^+ \equiv \bar{V}_{\text{max}}/u_\tau$
u_e	= specific internal energy
u_τ	= friction velocity, $u_\tau \equiv \sqrt{\tau_w/\rho}$
\hat{u}	= closed-form analog model variable
\mathbf{V}	= velocity vector
V	= velocity magnitude
V_∞	= freestream velocity

\bar{V}_{bulk}	=	bulk velocity
V_c	=	periodic wave velocity scale
\bar{V}_c	=	centerline velocity
\bar{V}_{max}	=	maximum average velocity
V_t	=	turbulent flow field characteristic velocity
V_η	=	contravariant velocity component defined in Eq. (4.5)
V_ξ	=	contravariant velocity component defined in Eq. (4.5)
w	=	span of a plane jet in the third dimension
$\bar{\mathbf{x}}$	=	vector of design variables in optimization algorithm
x	=	coordinate measured along the x axis
x_{1-2}	=	example design variables
y	=	coordinate measured along the y axis
y^+	=	wall-scaled dimensionless distance, $y^+ \equiv yu_\tau/\nu$
$y_{1\%}$	=	normal coordinate of plane jet at which axial velocity is 1% of the centerline velocity
y_h	=	normal coordinate of plane jet at which axial velocity is 50% of the centerline velocity
\hat{y}	=	closed-form analog model variable
Z	=	geopotential altitude
z	=	coordinate measured along the z axis
z_o	=	physical origin of round jet
\hat{z}	=	axial position relative to physical origin of round jet
α	=	transport equation under-relaxation variable
α_p	=	pressure under-relaxation variable
β	=	grid-stretching factor used in coordinate transformations
β	=	blending factor
β_r	=	Bradshaw's constant, $\beta_r = 0.30$
Γ	=	diffusivity coefficient

Γ	=	blending factor
$\bar{\bar{\gamma}}$	=	tensor defined in Eq. (1.45)
$\bar{\gamma}$	=	vector used in optimization algorithm, defined in Eq. (8.12)
γ	=	similarity variable
γ	=	Nikuradse constant, traditionally $\gamma \approx 0.0334$
γ_k	=	trace of $\bar{\bar{\gamma}}$, defined in Eq. (1.57)
$\bar{\bar{\delta}}$	=	Kronecker delta
$\bar{\bar{\epsilon}}$	=	tensor defined in Eq. (1.46)
ϵ_k	=	trace of $\bar{\bar{\epsilon}}$, defined in Eq. (1.58)
ϵ_o	=	turbulence model damping function dependent on the model
ϵ	=	approximate turbulent kinetic energy dissipation rate defined in Eq. (1.64)
$\tilde{\epsilon}$	=	exact turbulent kinetic energy dissipation rate defined in Eq. (2.11)
$\hat{\epsilon}$	=	solenoidal dissipation defined in Eq. (2.16)
$\hat{\epsilon}$	=	closed-form analog model variable
ϵ^+	=	wall-scaled dimensionless dissipation, $\epsilon^+ \equiv \epsilon \nu / u_\tau^4$
ϵ_o^+	=	wall-scaled damping function, $\epsilon_o^+ \equiv \epsilon_o \nu / u_\tau^4$
ζ	=	turbulence enstrophy defined in Eq. (1.84)
ζ	=	turbulent Jacobian magnitude defined in Eq. (1.85)
ζ^+	=	wall-scaled turbulence enstrophy, $\zeta^+ \equiv \zeta \nu^2 / u_\tau^4$
η	=	computational domain coordinate
θ	=	azimuthal coordinate in cylindrical coordinate system
θ^+	=	change of variables
κ	=	von Kármán constant, traditionally $\kappa \approx 0.40$
λ	=	coefficient of bulk viscosity
λ	=	mean vortex wavelength defined in Eq. (2.47)
λ^+	=	wall-scaled mean vortex wavelength, $\lambda^+ \equiv \lambda u_\tau / \nu$

μ	=	dynamic viscosity
μ_t	=	dynamic eddy viscosity
ν	=	kinematic viscosity, $\nu \equiv \mu/\rho$
$\hat{\nu}$	=	pipe-scaled kinematic viscosity, $\hat{\nu} \equiv \nu/(u_\tau R)$
ν_t	=	kinematic eddy viscosity, $\nu_t \equiv \mu_t/\rho$
$\hat{\nu}_t$	=	pipe-scaled kinematic eddy viscosity, $\hat{\nu}_t \equiv \nu_t/(u_\tau R)$
ν^+	=	wall-scaled dimensionless viscosity, $\nu^+ \equiv \nu_t/\nu$
ξ	=	computational domain coordinate
$\bar{\bar{\pi}}$	=	tensor defined in Eq. (1.48)
π_k	=	trace of $\bar{\bar{\pi}}$, defined in Eq. (1.60)
ρ	=	fluid density
$\bar{\bar{\sigma}}$	=	fluid stress tensor defined in Eq. (1.9)
σ	=	stress
σ_h	=	turbulence model closure coefficient
σ_k	=	turbulence model closure coefficient
σ_ε	=	turbulence model closure coefficient
$\sigma_{\hat{\varepsilon}}$	=	turbulence model closure coefficient
σ_ζ	=	turbulence model closure coefficient
σ_ω	=	turbulence model closure coefficient
$\sigma_{\tilde{\omega}}$	=	turbulence model closure coefficient
ζ	=	arbitrary scalar
$\bar{\bar{\tau}}$	=	Reynolds stress tensor defined in Eq. (1.41)
τ_{xy}	=	Reynolds shear stress, $\tau_{xy} \equiv -\overline{\rho \tilde{V}_x \tilde{V}_y}$
τ_w	=	wall shear stress
ϕ	=	arbitrary transport property
$\bar{\bar{\chi}}$	=	tensor defined in Eq. (1.47)

χ_k	=	trace of $\bar{\bar{\chi}}$, defined in Eq. (1.59)
ψ	=	arbitrary scalar
ψ_k	=	turbulence model damping function
ψ^+	=	change of variables
φ	=	arbitrary scalar
$\bar{\bar{\Omega}}$	=	rotation tensor defined in Eq. (A.11)
$\mathbf{\Omega}$	=	curl of the velocity vector, $\mathbf{\Omega} \equiv \nabla \times \mathbf{V}$
\mathcal{Q}	=	magnitude of $\mathbf{\Omega}$
\mathcal{Q}	=	relaxation factor
Ω_V^2	=	squared magnitude of the rotation tensor, $\Omega_V^2 \equiv \bar{\bar{\Omega}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\Omega}}(\bar{\mathbf{V}})$
ω	=	turbulence dissipation frequency defined in Eq. (1.78)
ω^+	=	wall-scaled turbulent dissipation frequency, $\omega^+ \equiv \omega v / u_\tau^2$
ω_c	=	periodic wave angular velocity
ω_t	=	turbulent flow field characteristic angular velocity
$\tilde{\omega}$	=	turbulent fluctuating vorticity defined in Eq. (2.14)
$\tilde{\omega}^+$	=	wall-scaled turbulent fluctuating vorticity, $\tilde{\omega}^+ \equiv \tilde{\omega} v / u_\tau^2$
$\hat{\omega}$	=	closed-form analog model variable

Superscripts

*	=	(star) guessed value
'	=	(prime) correction factor
T	=	transpose
o	=	previous solution

Subscripts

0	=	initial condition or value at the wall
---	---	--

D	=	center diagonal
E	=	cell to the east of the current cell
e	=	east face of current cell
h	=	associated with the turbulence variable, h
i	=	index variable
j	=	index variable
k	=	associated with the turbulence variable, k
L	=	lower diagonal
N	=	cell to the north of the current cell
n	=	north face of current cell
nb	=	neighboring boundaries
NE	=	cell to the north-east of the current cell
NW	=	cell to the north-west of the current cell
old	=	previous solution
P	=	current cell
r	=	associated with the r -component or axis
S	=	cell to the south of the current cell
s	=	south face of current cell
SE	=	cell to the south-east of the current cell
SW	=	cell to the south-west of the current cell
U	=	upper diagonal
u	=	associated with the x -velocity variable, u
W	=	cell to the west of the current cell
w	=	west face of current cell
x	=	associated with the x -component or axis
y	=	associated with the y -component or axis

z	=	associated with the z -component or axis
ε	=	associated with the turbulence variable, ε
θ	=	associated with the θ -component
ω	=	associated with the turbulence variable, ω

Other notation

,	=	(comma) denotes differentiation
-	=	(overbar) ensemble-averaged value
\sim	=	(tilde) fluctuating value
.	=	(dot) denotes differentiation with time
'	=	(prime) denotes differentiation
bold	=	denotes an array or vector

CHAPTER 1

TRADITIONAL TURBULENCE MODELING

I. Introduction

Fluid mechanics has been a topic of study for hundreds of years and has fascinated many of the greatest minds of history. Many applications today are dependent on correctly understanding and predicting the motion of fluids and much research has been conducted to that end. The laws that govern fluid motion have been understood since the mid 1800s and the vector mathematics needed to fully analyze three-dimensional fluid mechanics was sufficiently understood only a few decades later. See for example the work of Navier [1], Stokes [2], Hamilton [3–5], and Boyer and Merzback [6]. Since that time, these laws of motion and mathematical properties have been studied by countless researchers, and much progress in the physical understanding of both laminar and turbulent flows has been achieved. However, the complexity presented in completely understanding and correctly predicting fluid mechanics leaves room for much improvement.

Many analytical solutions exist for problems with simple geometries because the governing equations can be simplified and analytically applied. However, for more complex geometries and flow fields, the boundary conditions and vector equations are more complex and numerical means must be employed. Computational Fluid Dynamics (CFD) is a method of fluid flow calculation that uses a gridded domain to predict the flow field in or about a given geometry. This solution method has grown in popularity as computing power has increased, making solutions to complex flow problems more readily achievable. However, even with modern computational power, accurate solutions to flow problems can take days or weeks to converge.

The difficulty of modeling fluid mechanics is greatly increased when turbulent flow is considered. Indeed, the most intriguing and complex flow solutions are those for turbulent flow fields. Analytical

solutions for turbulent flow fields are much more difficult to develop than those for laminar flow fields. Therefore, turbulent flow modeling is left almost entirely to numerical methods.

The governing principles of mass and momentum conservation apply to turbulent flow and can be employed directly through CFD methods. However, the grid refinement required to capture the small scales of turbulence using CFD techniques is so extreme that solutions based on this technique can be fairly computationally expensive. Such CFD techniques are called Direct Numerical Simulation (DNS) methods and are currently employed mainly for simple domains and relatively low Reynolds numbers in which the small scales of turbulence are large in comparison to the grid element size of the domain of interest. For a review of DNS, see Moin and Mahesh [7].

In order to model the turbulent characteristics of larger geometries without the grid refinement required for DNS, additional relationships are commonly combined with the governing equations of fluid motion. These additional relationships should be based on the physics of turbulent motion to provide accurate simulations. The physical characteristics of turbulence have been studied in detail during the past century. Much of our current understanding of turbulence has been constructed through the results of countless experiments performed in the past two centuries. Although these findings have greatly expanded our understanding of turbulent flow characteristics, they are somewhat preliminary in nature and have not proven to be fully representative of turbulent behavior. In other words, there is still much to be learned about turbulence. To this extent, notable authors have commented.

“I am an old man now, and when I die and go to Heaven there are two matters on which I hope for enlightenment. One is quantum electrodynamics, and the other is the turbulent motion of fluids. And about the former I am really rather optimistic.” -Sir Horace Lamb [8]

“Turbulence...[was] probably invented by the Devil on the seventh day of Creation (when the Good Lord wasn't looking).” -P. Bradshaw [9]

“... less is known about the fine scale of turbulence ... than about the structure of atomic nuclei. Lack of basic knowledge about turbulence is holding back progress in fields as diverse as

cosmology, meteorology, aeronautics and biomechanics. Understanding the hierarchically organized complexity of turbulence may well provide a paradigm for understanding a variety of problems at the frontiers of physics research.” -U. Frisch and S. Orszag [10]

“Turbulence is the last great unsolved problem of classical physics. Remarks of this sort have been variously attributed to Sommerfeld, Einstein, and Feynman, although no one seems to know precise references, and searches of some likely sources have been unproductive. Of course, the allegation is a matter of fact, not much in need of support by a quotation from a distinguished author. However, it would be interesting to know when the matter was first recognised.” -P.J. Holmes, G. Berkooz, and J.L. Lumley [11]

Although the physical phenomenon of turbulence is not fully understood, the basic features of turbulent motion are known. First, turbulence is caused by inertial forces overpowering viscous forces within a fluid. At low Reynolds numbers, viscous forces dominate producing laminar flow. However, as the Reynolds number increases, the inertial forces of the fluid increase and eventually overcome the viscous forces. At this point, velocity and pressure fluctuations develop and the flow becomes irregular. This leads to the second most fundamentally understood characteristic of turbulence. Turbulent flow is comprised of fluctuations in pressure, velocity, and temperature making it impossible to reproduce the exact fluctuations from consecutive experiments although the average flow field is recreated.

A definition for turbulence commonly accepted today was given by Hinze [12]:

“Turbulent fluid motion is an irregular condition of flow in which the various quantities show a random variation with time and space coordinates, so that statistically distinct average values can be discerned.”

A definition that is perhaps more mathematically precise is that given by Phillips [13]:

“Turbulent fluctuations are irregular variations in certain quantities of a flow field (such as pressure, temperature and velocity) that are not predictably repeatable from one experiment to another.”

The definition by Phillips is more accurate for two reasons. First, his definition specifies the averaging method required to correctly deduce statistical information about a turbulent flow field. It can be shown and is well understood that ensemble averaging must be used instead of spatial and temporal averaging in order to accurately distinguish the average flow field from turbulent fluctuations in common unsteady turbulent flow fields. For a detailed discussion on averaging see Phillips [13] or Wilcox [14]. Second, Phillips uses the term “irregular” as opposed to “random” in referring to the flow fluctuations. Strictly speaking, although the fluctuations may appear to be random, they must at all times satisfy the governing equations of fluid flow. Therefore, the term “irregular” is a more correct way of describing the flow fluctuations.

The complex nature of turbulent flow is not yet fully understood. However, research conducted during the past century has allowed for certain properties of mean turbulent flow to be quantified, and traditional governing equations of mean turbulent flow have been developed. The challenge of forming equations truly representative of mean turbulent flow has inspired much research that has resulted in varying degrees of success. The most widely used models for internal flows include the $k-\varepsilon$ model based on the development of Jones and Launder [15], and variations of the $k-\omega$ model originally developed by Kolmogorov [16]. Commonly used aerodynamic (external) flow models include models developed by Spalart and Allmaras [17] and Baldwin and Barth [18]. Although these models are widely used, correct turbulence modeling is anything but a closed subject at this point in computational fluid dynamics and several concerns about the traditional methods have been identified.

Retracing the derivations of traditional turbulence modeling equations reveals seemingly minor yet possibly significant assumptions which may have hindered the validity of the models in the past. Recent re-examination of these equations by Phillips [13] has led to alternative developments of the well-known and explored $k-\varepsilon$, $k-\omega$, and $k-\zeta$ turbulence models. It is possible that the adjusted equations will produce more accurate results for turbulent flow calculations.

Durbin [19] expressed concerns about the implementation of smooth-wall boundary conditions for dissipation-based turbulence models nearly 20 years ago. However, even today many of the implementations of these models lack the correct boundary conditions. This discrepancy has not been

widely acknowledged in the literature likely because Durbin’s warning has not been well understood by the traditional CFD community. If significant improvements in turbulence modeling are to be made, the correct boundary conditions for traditional models must be understood.

Additionally, most turbulence models are developed to model the flow about smooth surfaces while few models are capable of modeling the effects of rough walls. In reality, there is no such thing as a perfectly smooth surface, and what we term a “smooth” or “hydraulically smooth” surface is a surface for which the roughness effects are seemingly negligible. The fact that no surface is perfectly smooth may shed some light as to why traditional models have had difficulty matching experimental data over a wide range of Reynolds numbers.

This dissertation focuses on addressing a few identifiable concerns with traditional turbulence modeling including the correct implementation of smooth-wall boundary conditions for dissipation-based turbulence models, the modeling of rough-wall flows, and the implementation of a turbulent-kinetic-energy equation suggested by Phillips [13]. To set this work in perspective to previous work, an overview of traditional turbulence model development and evaluation is included here.

II. Governing Equations of Fluid Motion

The Eulerian study of fluid mechanics is accomplished by understanding the basic laws and relationships that govern the motion of a differential fluid element. Such a fluid element, that is large enough for the molecular structure and the molecular motions to be ignored, can be studied by understanding its transport properties. A general transport equation for any transportable property can be written as

$$\frac{\partial(\rho\phi)}{\partial t} + \nabla \cdot (\rho\mathbf{V}\phi) = G_\phi''' \quad (1.1)$$

where ϕ is the property of transport per unit mass, and G_ϕ''' is the production rate per unit volume of the transport property. Three fundamental laws of physics can be applied to the fluid element by specifying three different transport properties in Eq. (1.1). These three laws are the conservation of mass, the

conservation of momentum, and the conservation of energy and comprise the governing equations of fluid mechanics.

A. Conservation of Mass: The Continuity Equation

The conservation of mass equation (the continuity equation) is developed from Eq. (1.1) by defining mass as the property of transport. Since mass per unit mass is unity, ϕ is unity. Assuming Newtonian physics, the mass of the continuum cannot be created nor destroyed and the generation term is zero. Therefore, the continuity equation is written as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{V}) = 0 \quad (1.2)$$

For incompressible flow, the first term can be dropped and the equation simplifies to

$$\nabla \cdot \mathbf{V} = 0 \quad (1.3)$$

B. Conservation of Momentum: Newton's Second Law

The conservation of momentum equation is developed from Eq. (1.1) by defining momentum as the property of transport. In this case, ϕ is the velocity vector since momentum per unit mass is velocity. The generation of momentum per unit volume is the sum of the forces per unit volume acting on the differential element. Therefore, the conservation of momentum equation is written as

$$\frac{\partial(\rho \mathbf{V})}{\partial t} + \nabla \cdot (\rho \mathbf{V} \mathbf{V}) = \mathbf{f}_b + \mathbf{f}_s \quad (1.4)$$

where \mathbf{f}_b and \mathbf{f}_s are the body and surface forces respectively. It is important to note that the conservation of momentum equation is the same as Newton's second law which states that force exerted on an object is equal to its mass times the acceleration of the object. Multiplying Eq. (1.4) by volume simplifies to the familiar expression

$$\mathbf{F} = m\mathbf{A} \quad (1.5)$$

Because velocity is a vector, the conservation of momentum equation is a set of three scalar equations. The vector product $\mathbf{V}\mathbf{V}$ is a tensor and can be written

$$\mathbf{VV} \equiv \begin{bmatrix} V_x V_x & V_x V_y & V_x V_z \\ V_y V_x & V_y V_y & V_y V_z \\ V_z V_x & V_z V_y & V_z V_z \end{bmatrix} \quad (1.6)$$

Using the vector identity given in Eq. (A.1) and applying the chain rule and the continuity equation, the left-hand side of Eq. (1.4) can be written as

$$\begin{aligned} \frac{\partial(\rho\mathbf{V})}{\partial t} + \nabla \cdot (\rho\mathbf{VV}) &= \rho \frac{\partial\mathbf{V}}{\partial t} + (\rho\mathbf{V} \cdot \nabla)\mathbf{V} + \mathbf{V} \frac{\partial\rho}{\partial t} + \mathbf{V}[\nabla \cdot (\rho\mathbf{V})] \\ &= \rho \frac{\partial\mathbf{V}}{\partial t} + (\rho\mathbf{V} \cdot \nabla)\mathbf{V} \end{aligned} \quad (1.7)$$

The surface forces can be written as the divergence of the fluid stress tensor

$$\mathbf{f}_s = \nabla \cdot \bar{\bar{\boldsymbol{\sigma}}} \quad (1.8)$$

where the tensor, $\bar{\bar{\boldsymbol{\sigma}}}$, includes both viscous normal and shear stresses.

$$\bar{\bar{\boldsymbol{\sigma}}} \equiv \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix} \quad (1.9)$$

The components of the fluid stress tensor are written in the form σ_{ij} where the subscript i represents the direction normal to the surface on which the stress is acting, and the subscript j represents the direction of the stress. It is important to note that compressive pressures included in the fluid stress tensor would have a minus sign as tensile stress is denoted positive.

The body forces include all other external forces (ie. gravitational and electromagnetic forces). If gravity is the only body force, this term can be written as

$$\mathbf{f}_b = -\rho g_o \nabla Z \quad (1.10)$$

where g_o is the acceleration of gravity at standard sea level and Z is the geopotential altitude.

Using Eqs. (1.7), (1.8), and (1.10) in Eq. (1.4) gives a common form of the conservation of momentum equation for fluid mechanics

$$\rho \left[\frac{\partial\mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla)\mathbf{V} \right] = \nabla \cdot \bar{\bar{\boldsymbol{\sigma}}} - \rho g_o \nabla Z \quad (1.11)$$

C. Conservation of Energy: First Law of Thermodynamics

The conservation of energy equation is developed from Eq. (1.1) by defining energy as the property of transport. The generation of energy per unit volume can result from heat, work done on the fluid element from the fluid stress tensor, or other volumetric heating resulting from nuclear or chemical reactions. Therefore, the conservation of energy can be written as

$$\frac{\partial(\rho e)}{\partial t} + \nabla \cdot (\rho \mathbf{V} e) = -\nabla \cdot \mathbf{q} + \nabla \cdot (\bar{\bar{\boldsymbol{\sigma}}} \cdot \mathbf{V}) + U_e''' \quad (1.12)$$

where e is the specific energy, \mathbf{q} is the heat flux vector, and U_e''' includes any other volumetric heating. The energy per unit mass is

$$e \equiv u_e + \frac{1}{2}V^2 + g_o Z \quad (1.13)$$

where u_e is the specific internal energy and V is the magnitude of the velocity vector of the fluid element. Using Eq. (1.13) in Eq. (1.12), applying the vector identities in Eqs. (A.2) and (A.3) as well as the continuity equation, and realizing that the geopotential altitude is not a function of time, the conservation of energy equation can be written in the form

$$\rho \left[\frac{\partial}{\partial t} (u_e + \frac{1}{2}V^2) + (\mathbf{V} \cdot \nabla) (u_e + \frac{1}{2}V^2 + g_o Z) \right] = -\nabla \cdot \mathbf{q} + \bar{\bar{\boldsymbol{\sigma}}} \cdot (\nabla \mathbf{V}) + \mathbf{V} \cdot (\nabla \cdot \bar{\bar{\boldsymbol{\sigma}}}) + U_e''' \quad (1.14)$$

A less cumbersome transport equation can be obtained by first developing the mechanical energy equation. Taking the dot product of the conservation of momentum equation, Eq. (1.11), with the velocity vector and applying the vector identity in Eq. (A.4) to the left-hand side gives

$$\rho \left[\frac{\partial}{\partial t} (\frac{1}{2}V^2) + (\mathbf{V} \cdot \nabla) (\frac{1}{2}V^2) \right] = \mathbf{V} \cdot (\nabla \cdot \bar{\bar{\boldsymbol{\sigma}}}) - \rho g_o (\mathbf{V} \cdot \nabla) Z \quad (1.15)$$

This can be rearranged to yield the well-known mechanical energy equation

$$\rho \left[\frac{\partial}{\partial t} (\frac{1}{2}V^2) + (\mathbf{V} \cdot \nabla) (\frac{1}{2}V^2 + g_o Z) \right] = \mathbf{V} \cdot (\nabla \cdot \bar{\bar{\boldsymbol{\sigma}}}) \quad (1.16)$$

Subtracting Eq. (1.16) from Eq. (1.14) yields a transport equation often called the thermal energy equation

$$\rho \left[\frac{\partial u_e}{\partial t} + (\mathbf{V} \cdot \nabla) u_e \right] = -\nabla \cdot \mathbf{q} + \bar{\bar{\boldsymbol{\sigma}}} \cdot (\nabla \mathbf{V}) + U_e''' \quad (1.17)$$

D. Application of Newtonian and Continuum Assumptions

Assuming Newtonian physics, the continuity equation, Newton's second law, and the thermal energy equation represent the general Eulerian equations of motion for a reference frame with only a single gravitational potential.

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{V}) = 0 \quad (1.2)$$

$$\rho \left[\frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla) \mathbf{V} \right] = \nabla \cdot \bar{\bar{\boldsymbol{\sigma}}} - \rho g_o \nabla Z \quad (1.11)$$

$$\rho \left[\frac{\partial u_e}{\partial t} + (\mathbf{V} \cdot \nabla) u_e \right] = -\nabla \cdot \mathbf{q} + \bar{\bar{\boldsymbol{\sigma}}} : (\nabla \mathbf{V}) + U_e''' \quad (1.17)$$

These equations provide a system of five scalar equations and 17 scalar unknowns. The unknowns are the three components of the heat transfer vector, the three components of the fluid velocity vector, the nine components of the fluid stress tensor, the specific internal energy of the fluid, and the fluid density. To complete the system of equations, additional relationships are needed.

At this point, two assumptions are applied. First, we assume that the fluid can be treated as a continuum (ie. the length scales are large compared to the molecular mean-free path). Second, we assume that the fluid is a Newtonian fluid. These assumptions allow the stress tensor to be related to the element deformation

$$\bar{\bar{\boldsymbol{\sigma}}} = (\lambda \nabla \cdot \mathbf{V} - p) \bar{\bar{\boldsymbol{\delta}}} + 2\mu \bar{\bar{\mathbf{S}}}(\mathbf{V}) \quad (1.18)$$

where p is the thermodynamic pressure, $\bar{\bar{\boldsymbol{\delta}}}$ is the Kronecker delta, λ is the coefficient of bulk viscosity, μ is the dynamic viscosity, and $\bar{\bar{\mathbf{S}}}(\mathbf{V})$ is the strain-rate tensor defined as

$$\bar{\bar{\mathbf{S}}}(\mathbf{V}) \equiv \frac{1}{2} \begin{bmatrix} \left(\frac{\partial V_x}{\partial x} + \frac{\partial V_x}{\partial x} \right) & \left(\frac{\partial V_x}{\partial y} + \frac{\partial V_y}{\partial x} \right) & \left(\frac{\partial V_x}{\partial z} + \frac{\partial V_z}{\partial x} \right) \\ \left(\frac{\partial V_y}{\partial x} + \frac{\partial V_x}{\partial y} \right) & \left(\frac{\partial V_y}{\partial y} + \frac{\partial V_y}{\partial y} \right) & \left(\frac{\partial V_y}{\partial z} + \frac{\partial V_z}{\partial y} \right) \\ \left(\frac{\partial V_z}{\partial x} + \frac{\partial V_x}{\partial z} \right) & \left(\frac{\partial V_z}{\partial y} + \frac{\partial V_y}{\partial z} \right) & \left(\frac{\partial V_z}{\partial z} + \frac{\partial V_z}{\partial z} \right) \end{bmatrix} \quad (1.19)$$

Fourier's law of thermal diffusion is commonly used to express the heat flux vector in terms of the thermodynamic temperature, T , as

$$\mathbf{q} \equiv -k_t \nabla T \quad (1.20)$$

For incompressible liquids and ideal gases, the specific internal energy can be related to the temperature through the thermodynamic property

$$du_e = c_v dT \quad (1.21)$$

where c_v is the constant volume specific heat. These relations allow the Eulerian equations of motion to be written for Newtonian fluids as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{V}) = 0 \quad (1.22)$$

$$\rho \left[\frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla) \mathbf{V} \right] = -\nabla(p + g_o \rho Z - \lambda \nabla \cdot \mathbf{V}) + \nabla \cdot [2\mu \bar{\bar{\mathbf{S}}}(\mathbf{V})] + g_o Z \nabla \rho \quad (1.23)$$

$$\rho c_v \left[\frac{\partial T}{\partial t} + (\mathbf{V} \cdot \nabla) T \right] = \nabla \cdot (k_t \nabla T) - p \nabla \cdot \mathbf{V} + \lambda (\nabla \cdot \mathbf{V})^2 + 2\mu \bar{\bar{\mathbf{S}}}(\mathbf{V}) \cdot \bar{\bar{\mathbf{S}}}(\mathbf{V}) + U_e''' \quad (1.24)$$

The three components of Eq. (1.23) are called the Navier-Stokes equations after Navier [1] and Stokes [2] who developed the equations independently. To complete the formulation from this point, the fluid properties ρ , λ , μ , c_v , and k_t must be known functions of pressure and temperature. An approximation commonly used for λ was suggested by Stokes and is $\lambda = -(2/3)\mu$. Note that for incompressible fluids, the value of λ is inconsequential in light of the incompressible continuity equation, Eq. (1.3). For incompressible Newtonian flow, Eqs. (1.22)–(1.24) can be simplified to

$$\nabla \cdot \mathbf{V} = 0 \quad (1.25)$$

$$\frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla) \mathbf{V} = -\nabla(p/\rho + g_o Z) + \nabla \cdot [2\nu \bar{\bar{\mathbf{S}}}(\mathbf{V})] \quad (1.26)$$

$$\frac{\partial T}{\partial t} + (\mathbf{V} \cdot \nabla) T = [\nabla \cdot (k_t \nabla T) / \rho + 2\nu \bar{\bar{\mathbf{S}}}(\mathbf{V}) \cdot \bar{\bar{\mathbf{S}}}(\mathbf{V}) + U_e''' / \rho] / c_v \quad (1.27)$$

where ν is the kinematic viscosity, μ/ρ . Careful examination of Eqs. (1.25)–(1.27) reveals that for incompressible Newtonian flow, the dependence of viscosity on temperature is the only term that couples

the Navier-Stokes equations to the thermal energy equation. Therefore, if the viscosity of the fluid is assumed independent of temperature, or the temperature gradients are small enough that the dependence can be ignored, the thermal energy equation can be decoupled from the other four equations and solved separately after the four equations, consisting of the continuity equation and the Navier-Stokes equations, have been solved.

The continuity equation and the Navier-Stokes equations provide a system of four equations and four unknowns and must be satisfied at every point in the flow. For laminar flow, these governing equations have been applied with much success both analytically and numerically. However, turbulent flow has irregular fluctuations in the flow velocity vector at extremely small scales which complicates the analytical solutions and requires extremely fine grid refinement for accurate CFD solutions. Direct Numerical Simulation relies on a gridded domain with elements fine enough to capture the smallest turbulent fluctuations. As a result, DNS simulations are commonly used to better understand the physics of the small scales of turbulence. In order to model larger turbulent flow fields, other means are often used that incorporate turbulent flow properties into the general equations of motion. The most common methods, and the method of choice for this research, are based on the Reynolds-averaged Navier-Stokes (RANS) equations.

III. The Reynolds-Averaged Navier-Stokes Equations

Turbulence models based on the RANS equations attempt to model the average fluid motion rather than trying to capture every fluctuation within the flow. By definition, at any given point and time, the continuity and the Navier-Stokes equations must be satisfied regardless of whether the flow is turbulent or laminar. Additionally, because the continuity equation is linear, it must be satisfied for the mean flow. The velocity and pressure at any given time in a turbulent flow can be expressed as the sum of the ensemble average of the property at that point and the fluctuating component. In the following notation, the over-bar and tilde represent the ensemble averages and fluctuating components respectively. Using this notation, the velocity vector and pressure scalar at a point in the flow can be written

$$\mathbf{V} = \bar{\mathbf{V}} + \tilde{\mathbf{V}} \quad (1.28)$$

$$p = \bar{p} + \tilde{p} \quad (1.29)$$

It is important to note that the average of the fluctuating component is zero by definition. (See Appendix A for properties of ensemble averaging.)

$$\overline{\tilde{\mathbf{V}}} \equiv \overline{\tilde{p}} \equiv 0 \quad (1.30)$$

Fluctuations in density can be ignored for flows which do not have high supersonic mean Mach numbers. Although the fluctuations in density due to turbulence are ignored, changes in the average density with time and space are retained.

Taking the ensemble average of the continuity equation yields

$$\overline{\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{V})} = 0 \quad (1.31)$$

which can be simplified by using rules of ensemble averaging to

$$\frac{\partial \bar{\rho}}{\partial t} + \nabla \cdot (\bar{\rho} \bar{\mathbf{V}}) = 0 \quad (1.32)$$

Using Eq. (1.28) in Eq. (1.2) gives

$$\frac{\partial \rho}{\partial t} + \nabla \cdot [\rho(\bar{\mathbf{V}} + \tilde{\mathbf{V}})] = \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \bar{\mathbf{V}}) + \nabla \cdot (\rho \tilde{\mathbf{V}}) = 0 \quad (1.33)$$

Applying Eq. (1.32) to Eq. (1.33) gives a form of the continuity equation for the fluctuations

$$\nabla \cdot (\rho \tilde{\mathbf{V}}) = 0 \quad (1.34)$$

Taking the ensemble average of the strain rate tensor and applying rules of ensemble averaging gives

$$\overline{\bar{\bar{\mathbf{S}}(\mathbf{V})}} = \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \quad (1.35)$$

Using the approximation given by Stokes for λ in Eq. (1.23) and taking the ensemble average gives

$$\overline{\rho \left[\frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla) \mathbf{V} \right]} = -\nabla(p + \rho g_o Z + \frac{2}{3} \mu \nabla \cdot \mathbf{V}) + \nabla \cdot [2\mu \bar{\bar{\mathbf{S}}}(\mathbf{V})] + g_o Z \nabla \rho \quad (1.36)$$

Applying rules of ensemble averaging and simplifying yields

$$\rho \left[\frac{\partial \bar{\mathbf{V}}}{\partial t} + \overline{(\mathbf{V} \cdot \nabla) \mathbf{V}} \right] = -\nabla(\bar{p} + \rho g_o Z + \frac{2}{3} \mu \nabla \cdot \bar{\mathbf{V}}) + \nabla \cdot [2\mu \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] + g_o Z \nabla \rho \quad (1.37)$$

Using Eqs. (1.28) and (1.30) and applying rules of ensemble averaging, the second term on the left-hand side of Eq. (1.37) can be written

$$\begin{aligned} \overline{(\mathbf{V} \cdot \nabla) \mathbf{V}} &= \overline{[(\bar{\mathbf{V}} + \tilde{\mathbf{V}}) \cdot \nabla](\bar{\mathbf{V}} + \tilde{\mathbf{V}})} \\ &= \overline{(\bar{\mathbf{V}} \cdot \nabla) \bar{\mathbf{V}}} + \overline{(\tilde{\mathbf{V}} \cdot \nabla) \tilde{\mathbf{V}}} \end{aligned} \quad (1.38)$$

Thus the RANS equations can then be written

$$\rho \left[\frac{\partial \bar{\mathbf{V}}}{\partial t} + \overline{(\bar{\mathbf{V}} \cdot \nabla) \bar{\mathbf{V}}} \right] = -\nabla(\bar{p} + \rho g_o Z + \frac{2}{3} \mu \nabla \cdot \bar{\mathbf{V}}) + \nabla \cdot [2\mu \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] + g_o Z \nabla \rho - \overline{\rho(\tilde{\mathbf{V}} \cdot \nabla) \tilde{\mathbf{V}}} \quad (1.39)$$

Notice that the last term on the right-hand side is the only term that involves the turbulent fluctuations. All other terms depend only on the average flow field. Using the vector identity in Eq. (A.1), this fluctuation term can be expanded to

$$\overline{\rho(\tilde{\mathbf{V}} \cdot \nabla) \tilde{\mathbf{V}}} = \overline{\nabla \cdot (\rho \tilde{\mathbf{V}} \tilde{\mathbf{V}})} - \overline{\tilde{\mathbf{V}} [\nabla \cdot (\rho \tilde{\mathbf{V}})]} \quad (1.40)$$

The second term on the right-hand side in this equation is identically zero from the turbulent fluctuation continuity equation given in Eq. (1.34). The first term on the right-hand side introduces the symmetric Reynolds stress tensor first suggested by Osborn Reynolds [20]

$$\bar{\bar{\boldsymbol{\tau}}} \equiv -\rho \overline{\tilde{\mathbf{V}} \tilde{\mathbf{V}}} = -\rho \begin{bmatrix} \overline{\tilde{V}_x \tilde{V}_x} & \overline{\tilde{V}_x \tilde{V}_y} & \overline{\tilde{V}_x \tilde{V}_z} \\ \overline{\tilde{V}_y \tilde{V}_x} & \overline{\tilde{V}_y \tilde{V}_y} & \overline{\tilde{V}_y \tilde{V}_z} \\ \overline{\tilde{V}_z \tilde{V}_x} & \overline{\tilde{V}_z \tilde{V}_y} & \overline{\tilde{V}_z \tilde{V}_z} \end{bmatrix} \quad (1.41)$$

thus

$$\overline{\rho(\tilde{\mathbf{V}} \cdot \nabla) \tilde{\mathbf{V}}} \equiv -\nabla \cdot \bar{\bar{\boldsymbol{\tau}}} \quad (1.42)$$

Using Eq. (1.42) in Eq. (1.39), the RANS equations can then be written as

$$\rho \left[\frac{\partial \bar{\mathbf{V}}}{\partial t} + \overline{(\bar{\mathbf{V}} \cdot \nabla) \bar{\mathbf{V}}} \right] = -\nabla(\bar{p} + \rho g_o Z + \frac{2}{3} \mu \nabla \cdot \bar{\mathbf{V}}) + \nabla \cdot [2\mu \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) + \bar{\bar{\boldsymbol{\tau}}}] + g_o Z \nabla \rho \quad (1.43)$$

Combining the continuity equation given in Eq. (1.32) with this general expression of the RANS equations provides four equations that contain ten unknowns. The unknowns include the three components

of the average velocity vector, the average pressure, and the six unknown terms in the symmetric Reynolds stress tensor. More relationships are needed to close this system of equations. This represents what is known as the turbulence closure problem and has resulted in a great deal of research. An overview of the most common approaches to the turbulence closure problem is discussed in the following section.

IV. Traditional Turbulence Closure

Two general methods are typically used to close the turbulence equations. The first method is referred to as Reynolds stress modeling and consists of modeling the Reynolds stress tensor in a tensor transport equation based on the mean flow. The second method relies on the Boussinesq hypothesis to relate the Reynolds stress tensor to the mean-strain-rate tensor. Only a brief overview of the Reynolds stress modeling method is included here as it is not studied in this work. The Boussinesq hypothesis method is discussed in more detail.

A. Reynolds Stress Modeling

The Reynolds stress tensor can be modeled by looking at the transport of the components of the tensor with the average flow. This transport equation is developed by first taking a moment of the Navier-Stokes equations by multiplying the equations by the fluctuating velocity. Taking the ensemble average of the result gives the Reynolds-stress-transport equation. The Reynolds-stress-transport equation can be written

$$\frac{\partial \overline{\tilde{\mathbf{V}}\tilde{\mathbf{V}}}}{\partial t} + (\overline{\mathbf{V}} \cdot \nabla) \overline{\tilde{\mathbf{V}}\tilde{\mathbf{V}}} = -\overline{\tilde{\gamma}(\overline{\mathbf{V}}, \tilde{\mathbf{V}})} - \nu \overline{\tilde{\boldsymbol{\varepsilon}}(\tilde{\mathbf{V}})} + \nu \nabla^2 \overline{\tilde{\mathbf{V}}\tilde{\mathbf{V}}} - \overline{\tilde{\chi}(\tilde{\mathbf{V}})} - \overline{\tilde{\boldsymbol{\pi}}(\tilde{\mathbf{V}}, \tilde{p})} \quad (1.44)$$

where

$$\overline{\tilde{\gamma}(\overline{\mathbf{V}}, \tilde{\mathbf{V}})} \equiv -\overline{\tilde{\mathbf{V}}[(\tilde{\mathbf{V}} \cdot \nabla)\overline{\mathbf{V}}]} - \left\{ \overline{\tilde{\mathbf{V}}[(\tilde{\mathbf{V}} \cdot \nabla)\overline{\mathbf{V}}]} \right\}^T \quad (1.45)$$

$$\overline{\tilde{\boldsymbol{\varepsilon}}(\tilde{\mathbf{V}})} \equiv 2 \begin{bmatrix} \overline{\nabla(\tilde{V}_x) \cdot \tilde{\nabla}(\tilde{V}_x)} & \overline{\nabla(\tilde{V}_x) \cdot \tilde{\nabla}(\tilde{V}_y)} & \overline{\nabla(\tilde{V}_x) \cdot \tilde{\nabla}(\tilde{V}_z)} \\ \overline{\nabla(\tilde{V}_y) \cdot \tilde{\nabla}(\tilde{V}_x)} & \overline{\nabla(\tilde{V}_y) \cdot \tilde{\nabla}(\tilde{V}_y)} & \overline{\nabla(\tilde{V}_y) \cdot \tilde{\nabla}(\tilde{V}_z)} \\ \overline{\nabla(\tilde{V}_z) \cdot \tilde{\nabla}(\tilde{V}_x)} & \overline{\nabla(\tilde{V}_z) \cdot \tilde{\nabla}(\tilde{V}_y)} & \overline{\nabla(\tilde{V}_z) \cdot \tilde{\nabla}(\tilde{V}_z)} \end{bmatrix} \quad (1.46)$$

$$\bar{\bar{\chi}}(\tilde{\mathbf{V}}) \equiv \begin{bmatrix} \nabla \cdot (\tilde{V}_x \tilde{V}_x \tilde{\mathbf{V}}) & \nabla \cdot (\tilde{V}_x \tilde{V}_y \tilde{\mathbf{V}}) & \nabla \cdot (\tilde{V}_x \tilde{V}_z \tilde{\mathbf{V}}) \\ \nabla \cdot (\tilde{V}_y \tilde{V}_x \tilde{\mathbf{V}}) & \nabla \cdot (\tilde{V}_y \tilde{V}_y \tilde{\mathbf{V}}) & \nabla \cdot (\tilde{V}_y \tilde{V}_z \tilde{\mathbf{V}}) \\ \nabla \cdot (\tilde{V}_z \tilde{V}_x \tilde{\mathbf{V}}) & \nabla \cdot (\tilde{V}_z \tilde{V}_y \tilde{\mathbf{V}}) & \nabla \cdot (\tilde{V}_z \tilde{V}_z \tilde{\mathbf{V}}) \end{bmatrix} \quad (1.47)$$

$$\bar{\bar{\pi}}(\tilde{\mathbf{V}}, \tilde{p}) \equiv \frac{1}{\rho} \overline{\tilde{\mathbf{V}} \nabla \tilde{p}} + \frac{1}{\rho} [\overline{\tilde{\mathbf{V}} \nabla \tilde{p}}]^T - \overline{(\nabla \cdot \tilde{\mathbf{V}}) \tilde{\mathbf{V}} \tilde{\mathbf{V}}} \quad (1.48)$$

and $\overline{\tilde{\mathbf{V}} \tilde{\mathbf{V}}}$ is the negative of the specific Reynolds stress tensor, $\bar{\bar{\tau}}/\rho$. In Eq. (1.48) we have used the definition

$$\hat{\tilde{p}} \equiv \tilde{p} - \frac{1}{3} \mu \nabla \cdot \tilde{\mathbf{V}} \quad (1.49)$$

This approach to turbulence modeling requires knowledge of the new scalar terms in the tensors on the right-hand side of Eq. (1.44). These terms are correlations of the velocity and pressure fluctuations and are unknown. More transport equations for these unknown scalar terms could be developed, but this repeated process would never produce enough equations to match the number of unknowns. At some point, the subsequent scalar terms obtained from modeling the stress transport must be related to the mean flow in order to “close” the formulation. The method of introducing transport equations for the Reynolds stress tensor has not shown to produce better results than relating the Reynolds stress tensor to the mean flow and will not be discussed further.

B. The Boussinesq Hypothesis

The most direct and commonly employed approach to relating the Reynolds stress tensor to the mean properties of the flow is based on an analogy between molecular and turbulent transport first suggested by Boussinesq [21]. Application of the Boussinesq hypothesis suggests that the Reynolds stress tensor might be modeled as a linear function of the mean-strain-rate tensor just as the total molecular stress tensor is modeled as a linear function of the total-strain-rate tensor. The Boussinesq hypothesis can be expressed as

$$\bar{\bar{\tau}} = 2\mu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \frac{2}{3}(\rho k + \mu_t \nabla \cdot \bar{\mathbf{V}}) \bar{\bar{\delta}} \quad (1.50)$$

where μ_t is the dynamic eddy viscosity, k is the turbulent kinetic energy per unit mass, and $\bar{\bar{\delta}}$ is the Kronecker delta. The dynamic eddy viscosity and turbulent kinetic energy per unit mass will be discussed

in more detail at a later point. Notice how Eq. (1.50) compares to Eq. (1.18). It should be noted that in the literature, this is often called the Boussinesq approximation. However, it is not actually an approximation. It is better termed the Boussinesq hypothesis or assumption. Applying the Boussinesq hypothesis to the RANS equations given in Eq. (1.43) yields

$$\rho \left[\frac{\partial \bar{\mathbf{V}}}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) \bar{\mathbf{V}} \right] = -\nabla[\bar{p} + \rho g_o Z + \frac{2}{3} \rho k + \frac{2}{3} (\mu + \mu_t) \nabla \cdot \bar{\mathbf{V}}] + \nabla \cdot [2(\mu + \mu_t) \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] + g_o Z \nabla \rho \quad (1.51)$$

A pseudo mean pressure can be defined by combining like terms on the right-hand side

$$\hat{p} \equiv \bar{p} + \rho g_o Z + \frac{2}{3} \rho k + \frac{2}{3} (\mu + \mu_t) \nabla \cdot \bar{\mathbf{V}} \quad (1.52)$$

and the Boussinesq RANS equations can be written as

$$\rho \left[\frac{\partial \bar{\mathbf{V}}}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) \bar{\mathbf{V}} \right] = -\nabla(\hat{p}) + \nabla \cdot [2(\mu + \mu_t) \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] + g_o Z \nabla \rho \quad (1.53)$$

In order to close the formulation with the Boussinesq hypothesis, the dynamic eddy viscosity, μ_t , and the turbulent kinetic energy per unit mass, k , must be related to the other flow properties. Several methods have been suggested for closing the formulation based on the Boussinesq hypothesis, and can be classified as zero, one, or two equation turbulence models.

Most zero-equation models (also known as algebraic models) are based on a hypothesis first made by Ludwig Prandtl [22]. This hypothesis is referred to as mixing-length theory. Prandtl hypothesized that the turbulence characteristics were related to a characteristic length and velocity scale associated with the turbulent fluctuations. Thus, the closing equations are simply algebraic relationships between the turbulent parameters based on characteristics of the flow known a priori. Zero-equation models are classified as incomplete models because they require properties of the turbulent flow field to be known a priori.

Most one-equation models are based on a subsequent hypothesis by Prandtl [23]. In this development, Prandtl hypothesized that the eddy viscosity was proportional to the product of the square root of the turbulent kinetic energy per unit mass and a characteristic length scale.

$$\mu_t \propto l_t \sqrt{k} \quad (1.54)$$

Prandtl used a modeled version of a turbulent kinetic energy transport equation and related the eddy viscosity to the turbulent kinetic energy algebraically. The length scale was also calculated algebraically from the mean flow. The distinguishing factors of most one-equation models is that they model the turbulent kinetic energy per unit mass by a differential equation, they express the eddy viscosity as a function of the turbulent kinetic energy per unit mass, and they calculate some type of length scale from the mean fluid velocity. Many one-equation models have been proposed including those by Emmons [24], Glushko [25], and Wolfshtein [26]. Bradshaw, Ferriss, and Atwell [27] also proposed a one-equation model based on the turbulent kinetic energy, but do not use the Boussinesq hypothesis. Other one-equation models that are based on some transport property other than turbulent kinetic energy include models by Nee and Kovaszny [28], Sekundov [29], Baldwin and Barth [18], Spalart and Allmaras [17], and Menter [30].

Most two-equation turbulence models are based on a method first suggested by Kolmogorov [16]. Kolmogorov proposed using a differential equation to model the turbulent kinetic energy per unit mass as Prandtl had. However, Kolmogorov also suggested using a supplementary differential transport equation for a scalar quantity known as the specific dissipation rate. This quantity represents a characteristic frequency related to the turbulent kinetic energy dissipation and was given the symbol ω . Algebraic relationships are then developed on the basis of dimensional analysis to calculate a length scale and express the eddy viscosity as a function of the two modeled variables and the length scale. Two-equation formulations are called complete models because they allow the mean turbulent flow to be modeled without any characteristics of the flow being known a priori. Much attention has been given to this method and several two-equation models have consequently been developed. The most common two-equation models can be classified as either k - ε or k - ω models where ε is the turbulent kinetic energy dissipation rate per unit mass. These models will be discussed in more detail in a subsequent section. Two-equation models that are based on transport properties other than ε or ω include models suggested by Rotta [31], Rotta [32], Zeierman and Wolfshtein [33], and Speziale, Abid, and Anderson [34].

C. Turbulent Kinetic Energy Transport

The turbulent kinetic energy is a measure of the kinetic energy of the fluid resulting from the turbulent fluctuations that are generated from velocity gradients in the flow. Turbulent kinetic energy is transported with the mean flow and is dissipated through molecular viscosity. The specific turbulent kinetic energy is commonly denoted as k and is defined to be one-half the mean square magnitude of the velocity fluctuations. This is equal to one-half the trace of the negative of the specific Reynolds stress tensor.

$$k \equiv \frac{1}{2} \overline{\tilde{\mathbf{V}} \cdot \tilde{\mathbf{V}}} = \frac{1}{2} \overline{\tilde{V}^2} = \frac{1}{2} \left(\overline{\tilde{V}_x^2} + \overline{\tilde{V}_y^2} + \overline{\tilde{V}_z^2} \right) = -\frac{1}{2} \text{trace} \left(\frac{\tilde{\boldsymbol{\tau}}}{\rho} \right) \quad (1.55)$$

Because of the relation of the turbulent kinetic energy to the Reynolds stress tensor, a transport equation for the turbulent kinetic energy can be taken from the Reynolds-stress-transport equation. Taking the trace of the Reynolds-stress-transport equation given in Eq. (1.44) gives

$$\frac{\partial k}{\partial t} + (\overline{\mathbf{V}} \cdot \nabla) k = \gamma_k - \nu \varepsilon_k + \nu \nabla^2 k - \chi_k - \pi_k \quad (1.56)$$

where the subscript k represents one-half the trace of the tensor. A close look at these terms reveals that they can each be written as functions of alternate quantities. The trace terms on the right-hand side of Eq. (1.56) can be written

$$\gamma_k \equiv \frac{\tilde{\boldsymbol{\tau}}}{\rho} \cdot \tilde{\mathbf{J}}(\overline{\mathbf{V}}) \quad (1.57)$$

$$\varepsilon_k \equiv \overline{\tilde{\mathbf{J}}(\tilde{\mathbf{V}}) \cdot \tilde{\mathbf{J}}(\tilde{\mathbf{V}})} \quad (1.58)$$

$$\chi_k \equiv \frac{1}{2} \nabla \cdot \overline{\tilde{V}^2 \tilde{\mathbf{V}}} \quad (1.59)$$

$$\pi_k \equiv \frac{1}{\rho} \left[\nabla \cdot \overline{\tilde{p}(\tilde{\mathbf{V}})} - \overline{\tilde{p}(\nabla \cdot \tilde{\mathbf{V}})} + \frac{1}{2} \overline{\tilde{V}^2 \tilde{\mathbf{V}} \cdot \nabla \rho} \right] \quad (1.60)$$

where the tensor $\tilde{\mathbf{J}}$ is the Jacobian tensor and can be expressed in Cartesian coordinates for an arbitrary flow field, \mathbf{V} , as

$$\bar{\bar{\mathbf{J}}}(\mathbf{V}) \equiv \begin{bmatrix} \frac{\partial V_x}{\partial x} & \frac{\partial V_x}{\partial y} & \frac{\partial V_x}{\partial z} \\ \frac{\partial V_y}{\partial x} & \frac{\partial V_y}{\partial y} & \frac{\partial V_y}{\partial z} \\ \frac{\partial V_z}{\partial x} & \frac{\partial V_z}{\partial y} & \frac{\partial V_z}{\partial z} \end{bmatrix} \quad (1.61)$$

Using Eqs. (1.49) and (1.57)–(1.60) in Eq. (1.56) gives the turbulent-kinetic-energy transport equation per unit mass

$$\begin{aligned} \frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)k &= \frac{\bar{\bar{\boldsymbol{\tau}}}}{\rho} \cdot \bar{\bar{\mathbf{J}}}(\bar{\mathbf{V}}) - \nu \overline{\bar{\mathbf{J}}}(\tilde{\mathbf{V}}) \cdot \bar{\bar{\mathbf{J}}}(\tilde{\mathbf{V}}) + \frac{1}{\rho} \overline{\tilde{p}(\nabla \cdot \tilde{\mathbf{V}})} - \frac{1}{3} \overline{\nu(\nabla \cdot \tilde{\mathbf{V}})^2} \\ &+ \nu \nabla^2 k - \frac{1}{\rho} \left\{ \nabla \cdot \left[\frac{1}{2} \rho \overline{\tilde{V}^2 \tilde{\mathbf{V}}} + \tilde{p} \tilde{\mathbf{V}} - \frac{1}{3} \mu \overline{(\nabla \cdot \tilde{\mathbf{V}}) \tilde{\mathbf{V}}} \right] \right\} \end{aligned} \quad (1.62)$$

Each of the terms on the right-hand side of Eq. (1.62) represents physical aspects of the turbulent-kinetic-energy transport. The first term on the right-hand side of Eq. (1.62) is the production term and is the rate that specific kinetic energy is transferred from the mean flow to the turbulent fluctuations. Using the Boussinesq hypothesis given in Eq. (1.50), the production term can be written

$$\begin{aligned} \frac{\bar{\bar{\boldsymbol{\tau}}}}{\rho} \cdot \bar{\bar{\mathbf{J}}}(\bar{\mathbf{V}}) &= 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{J}}}(\bar{\mathbf{V}}) - \frac{2}{3} (k + \nu_t \nabla \cdot \bar{\mathbf{V}}) \bar{\bar{\boldsymbol{\delta}}} \cdot \bar{\bar{\mathbf{J}}}(\bar{\mathbf{V}}) \\ &= 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \frac{2}{3} (k + \nu_t \nabla \cdot \bar{\mathbf{V}}) \nabla \cdot \bar{\mathbf{V}} \end{aligned} \quad (1.63)$$

where ν_t is the kinematic eddy viscosity, μ_t/ρ . Notice that the second term on the right-hand side of Eq. (1.63) is zero for incompressible flow. The second term on the right-hand side of Eq. (1.62) is often understood to represent the dissipation of k , or the rate that the kinetic energy of the turbulent fluctuations is dissipated to thermal energy through viscosity. It is commonly called the dissipation per unit mass and has been modeled in many different ways. For now, it is simply denoted as

$$\varepsilon \equiv \nu \varepsilon_k = \nu \overline{\bar{\mathbf{J}}}(\tilde{\mathbf{V}}) \cdot \bar{\bar{\mathbf{J}}}(\tilde{\mathbf{V}}) \quad (1.64)$$

The third and fourth terms on the right-hand side of Eq. (1.62) are called the dilation terms and represent the exchange of energy between the turbulent kinetic energy and the thermal energy of the fluid resulting from fluid compression or expansion. For compressible flow the dilation terms are usually small and often neglected in modeling. For incompressible flow the dilation terms are identically zero.

$$\frac{1}{\rho} \overline{\tilde{p}(\nabla \cdot \tilde{\mathbf{V}})} - \frac{1}{3} \overline{\nu(\nabla \cdot \tilde{\mathbf{V}})^2} = 0 \quad (1.65)$$

The fifth term on the right-hand side of Eq. (1.62) comes from molecular diffusion and is commonly understood to account for the diffusion of turbulent kinetic energy resulting from the fluid's molecular transport properties. Finally, the remaining terms on the right-hand side of Eq. (1.62) are usually referred to as the turbulent transport terms. These terms include the transport of turbulent kinetic energy caused by the turbulent fluctuations. The turbulent transport term is often modeled as a gradient diffusion process

$$\frac{1}{2} \rho \overline{\tilde{V}^2 \tilde{\mathbf{V}}} + \overline{\tilde{p} \tilde{\mathbf{V}}} - \frac{1}{3} \mu \overline{(\nabla \cdot \tilde{\mathbf{V}}) \tilde{\mathbf{V}}} = -(\mu_t / \sigma_k) \nabla k \quad (1.66)$$

where σ_k is a closure constant. Using Eqs. (1.63)–(1.66) in Eq. (1.62) gives a version of the turbulent-kinetic-energy-transport equation commonly used for modeling.

$$\frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) k = 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \frac{2}{3} (k + \nu_t \nabla \cdot \bar{\mathbf{V}}) \nabla \cdot \bar{\mathbf{V}} - \varepsilon + \nabla \cdot [(\nu + \nu_t / \sigma_k) \nabla k] \quad (1.67)$$

Combining Eq. (1.67) with the mean continuity equation and the RANS equations provides a system of five equations with seven unknowns. The unknowns are the three components of the mean velocity vector, the mean pressure, and the three turbulence variables, ν_t , k , and ε . In order to complete the system, we must provide additional relations between these variables. As stated previously, most one-equation models accomplish this by relating the turbulence variables through an algebraic expression. Most complete turbulence models, however, employ an additional transport equation and relate the kinematic eddy viscosity, ν_t , algebraically.

D. Turbulent Dissipation Transport

The modeled transport equation for the turbulent dissipation, ε , is usually developed by analogy with Eq. (1.67), and can be written as

$$\frac{\partial \varepsilon}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) \varepsilon = 2C_{\varepsilon 1} \nu_t \frac{\varepsilon}{k} \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - C_{\varepsilon 2} \frac{\varepsilon^2}{k} + \nabla \cdot [(\nu + \nu_t / \sigma_\varepsilon) \nabla \varepsilon] \quad (1.68)$$

where $C_{\varepsilon 1}$, $C_{\varepsilon 2}$, and σ_ε are closure constants. Note that the turbulent dissipation transport equation is traditionally constructed by dimensional analysis and is not developed rigorously from the Navier-Stokes

equations. Wilcox [35] makes mention of this concern and gives what he terms an “exact” equation for the turbulent dissipation, ε , by taking a moment of the Navier-Stokes equation using the definition of the turbulent dissipation given in Eq. (1.64). However, the resulting equation is extremely complicated and Wilcox submits that closure coefficients for the resulting differential equation are all but impossible to measure at this point. Thus the modeled version of the dissipation transport given in Eq. (1.68) is traditionally used for lack of a useful version of a more rigorously derived equation.

E. Length and Velocity Scales

The Boussinesq hypothesis allows the Reynolds stress tensor to be written in terms of two unknowns, the kinematic eddy viscosity and the specific turbulent kinetic energy. By definition, the specific turbulent kinetic energy is related to the Reynolds stress tensor, allowing a transport equation for the specific turbulent kinetic energy to be developed. This provides an additional equation but also introduces an additional unknown, the turbulent dissipation. By analogy to the turbulent kinetic energy transport equation, a transport equation for the turbulent dissipation can be found. In order to complete the closure of the Boussinesq RANS equations, a relation must be found that relates one of these unknowns to other parameters. This is accomplished by observing the length and velocity scales of turbulence.

The kinematic eddy viscosity is a diffusivity coefficient and can be better understood by first looking at the diffusivity coefficient resulting from the kinetic theory of gases, molecular viscosity. All kinematic diffusivity coefficients have units of length times velocity. Molecular viscosity has been shown to be proportional to the product of the molecular mean free path and the square root of the total specific molecular kinetic energy. This leads to the obvious claim that the kinematic eddy viscosity can be modeled as a product of a turbulent length scale and a turbulent velocity scale. These scales are not properties of the fluid, but properties of the flow. Perhaps the seemingly most natural velocity scale of a flow is the average velocity of the fluid. This relation was first used by Prandtl who hypothesized that the characteristic length and velocity of a turbulent flow could be related to the characteristic velocity and length of the average flow. However, it is important to note that the turbulent kinetic energy is independent of the average fluid

velocity. Rather, the turbulent kinetic energy is a measure of the energy contained in the velocity fluctuations where the average of the fluctuations is zero. Obviously, there is neither a single length scale nor a single velocity scale that is fully representative of turbulent fluctuations. Turbulence is the superposition of a myriad of velocity fluctuations each with differing characteristic lengths and energies. However, spectral analysis of turbulent fluctuations provides insight into velocity and length scales for a turbulent flow. Geoffrey I. Taylor [36] was the first to present such an analysis. Taylor developed a mathematical relation between the velocity fluctuations and the turbulence energy spectrum. Thus, a characteristic length of the flow could be found from some type of weighted average of the various eddy lengths of the turbulent flow. For a detailed discussion on the energy spectrum, see Hinze [37].

Prandtl [23] suggested that the square root of the specific kinetic energy of the fluctuating velocity field offers a valuable velocity scale for the flow field.

$$V_t = k^{1/2} \quad (1.69)$$

Realizing that periodic waves can be characterized by an angular velocity, ω_c , and a translational velocity, V_c , provides a relationship for a characteristic length scale, l_c . These two velocity scales of a periodic wave are related to the length scale of the wave through

$$l_c \equiv \frac{V_c}{\omega_c} \quad (1.70)$$

Therefore, the turbulent characteristic length scale should be proportional to the turbulent characteristic velocity divided by a turbulent characteristic angular velocity.

$$l_t \propto \frac{V_t}{\omega_t} \quad (1.71)$$

The characteristic angular velocity is often taken to be proportional to the approximate dissipation divided by the specific kinetic energy

$$\omega_t \propto \frac{\varepsilon}{k} \quad (1.72)$$

It is important to note that this definition links the characteristic length of the turbulent flow field to the characteristic length of turbulent dissipation, not to the characteristic length of the eddies in which the most

energy is found. This is an item of concern and will be addressed later. Using Eqs. (1.69), (1.71), and (1.72) to represent the kinematic eddy viscosity yields

$$\nu_t = C_\mu k^2 / \varepsilon \quad (1.73)$$

where C_μ is a closure constant. This relationship finally closes the first turbulence model of interest, known as the k - ε model. The other two models presented in the subsequent section can be constructed through a change of variables from the k - ε model as will be shown.

F. Resulting Energy-Dissipation Turbulence Models

Many two-equation turbulence models have been proposed on the basis of transport equations for the turbulent kinetic energy and the dissipation. In this section, three traditional models are presented that are based on these two transport equations. The purpose of this section is to provide the fundamental equations to these traditional models, not to provide a catalog of existing models. The models presented here are often thought to be fundamentally different. However, it will be shown that these models are all based on the transport of dissipation, and can be derived through a change of variables from the other models. For each of the models presented in this section, incompressible flow is assumed.

1. The k - ε Model

Combining Eqs. (1.67), (1.68), and (1.73) gives the traditional k - ε model first made popular by Jones and Launder [15].

$$\frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)k = 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \varepsilon + \nabla \cdot [(v + \nu_t / \sigma_k) \nabla k] \quad (1.74)$$

$$\frac{\partial \varepsilon}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)\varepsilon = 2C_{\varepsilon 1} \nu_t \frac{\varepsilon}{k} \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - C_{\varepsilon 2} \frac{\varepsilon^2}{k} + \nabla \cdot [(v + \nu_t / \sigma_\varepsilon) \nabla \varepsilon] \quad (1.75)$$

$$\nu_t = C_\mu k^2 / \varepsilon \quad (1.76)$$

The most widely used closure coefficients are attributed to Launder, Morse, Rodi, and Spalding [38] who suggested very similar coefficients after significant computer optimization. The commonly used coefficients are

$$C_\mu = 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3 \quad (1.77)$$

The k - ε model in its general form cannot be integrated to the wall. Generally wall functions or damping functions are incorporated into the model for near-wall treatment. For more detail and example results for the k - ε model, see Appendix B.

2. The k - ω Model

Using the relationship given in Eq. (1.72), the k - ε model can be reparameterized in terms of the turbulent dissipation frequency, ω , rather than the turbulent dissipation rate, ε . Note that ω is sometimes called the specific dissipation rate in the literature. Using C_μ as the proportionality constant gives

$$\omega \equiv \frac{\varepsilon}{C_\mu k} = \frac{2\nu \overline{\mathbf{J}(\tilde{\mathbf{V}}) \cdot \mathbf{J}(\tilde{\mathbf{V}})}}{C_\mu \overline{\tilde{\mathbf{V}} \cdot \tilde{\mathbf{V}}}} \quad (1.78)$$

Solving for ε and substituting the result into Eq. (1.73) gives

$$\nu_t = \frac{k}{\omega} = \frac{C_\mu \left(\overline{\tilde{\mathbf{V}} \cdot \tilde{\mathbf{V}}} \right)^2}{4\nu \overline{\mathbf{J}(\tilde{\mathbf{V}}) \cdot \mathbf{J}(\tilde{\mathbf{V}})}} \quad (1.79)$$

Using Eq. (1.78) in Eq. (1.67) gives the traditional k - ω model first attempted by Kolmogorov [16] and revised by many others. See for example Wilcox [39,40], Speziale, Abid, and Anderson [34], Menter [41], Peng, Davidson, and Holmberg [42], Kok [43], and Hellsten [44]. Thus, the turbulent-energy-transport equation is given by

$$\frac{\partial k}{\partial t} + (\overline{\mathbf{V}} \cdot \nabla)k = 2\nu_t \overline{\mathbf{S}(\overline{\mathbf{V}}) \cdot \mathbf{S}(\overline{\mathbf{V}})} - C_\mu k \omega + \nabla \cdot [(v + \nu_t / \sigma_k) \nabla k] \quad (1.80)$$

Through dimensional analysis and analogy with Eq. (1.80), the dissipation-frequency-transport equation is given by

$$\frac{\partial \omega}{\partial t} + (\overline{\mathbf{V}} \cdot \nabla)\omega = 2C_{\omega 1} \nu_t \frac{\omega}{k} \overline{\mathbf{S}(\overline{\mathbf{V}}) \cdot \mathbf{S}(\overline{\mathbf{V}})} - C_{\omega 2} \omega^2 + \nabla \cdot [(v + \nu_t / \sigma_\omega) \nabla \omega] \quad (1.81)$$

The algebraic relationship to close the system is

$$\nu_t = \frac{k}{\omega} \quad (1.82)$$

Many values for the closure coefficients have been suggested as the model has evolved over time.

Currently, the following coefficients are commonly used

$$C_\mu = 0.09, \quad C_{\omega 1} = 0.52, \quad C_{\omega 2} = 0.072, \quad \sigma_k = 2.0, \quad \sigma_\omega = 2.0 \quad (1.83)$$

It is also not uncommon for a form of wall damping functions to be applied to this model. The wall damping functions of Wilcox [45] are perhaps the most widely used to date. For more detail and example results for the k - ω model, see Appendix B.

3. The k - ζ Model

A less-common turbulence model based on enstrophy was first presented by Robinson, Harris, and Hassan [46] and further developed by Robinson and Hassan [47]. Enstrophy, ζ , is traditionally defined as the mean square magnitude of the fluctuating vorticity

$$\zeta \equiv \overline{(\nabla \times \tilde{\mathbf{V}}) \cdot (\nabla \times \tilde{\mathbf{V}})} \quad (1.84)$$

However, in their model, Robinson et al. defined an approximate turbulent energy dissipation term, $\varepsilon \equiv \zeta \nu$.

This is simply a change of variables defining the turbulent Jacobian magnitude

$$\zeta \equiv \frac{\varepsilon}{\nu} = \overline{\tilde{\mathbf{J}}(\tilde{\mathbf{V}}) \cdot \tilde{\mathbf{J}}(\tilde{\mathbf{V}})} \quad (1.85)$$

Using this in Eqs. (1.74) and (1.76) gives the turbulent-energy-transport equation

$$\frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)k = 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \nu \zeta + \nabla \cdot [(\nu/3 + \nu_t/\sigma_k)\nabla k] \quad (1.86)$$

and the closing algebraic equation

$$\nu_t = C_\mu \frac{k^2}{\nu \zeta} \quad (1.87)$$

The turbulent-enstrophy-transport equation used in the model is

$$\begin{aligned} \frac{\partial \zeta}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)\zeta = & [C_{\zeta 1} \zeta / \bar{\Omega}^2 - C_{\zeta 2} \nu_t^2 \bar{\Omega} / \nu k - C_{\zeta 3} \nu_t \zeta / k \bar{\Omega}] \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot (\bar{\Omega} \bar{\Omega}) \\ & + C_{\zeta 2} \nu_t \bar{\Omega}^3 / 3\nu + C_{\zeta 3} \zeta \bar{\Omega} / 3 - C_{\zeta 4} \nu \zeta^2 / k + C_{\zeta 5} (\nu_t \zeta / k) \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) \\ & - 4\nu_t \bar{\mathbf{S}}(\bar{\Omega}) \cdot \bar{\mathbf{S}}(\bar{\Omega}) - (\nabla \times \bar{\Omega}) \cdot \{2\nabla \cdot [\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}})] + \frac{1}{3} \nabla k\} \\ & + C_{\zeta 6} [(\nu_t / k) \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\Omega} - \frac{1}{3} \bar{\Omega}] \cdot (\nabla k \times \nabla \zeta) / \bar{\Omega}^2 + \nabla \cdot [(\nu + \nu_t / \sigma_\zeta) \nabla \zeta] \end{aligned} \quad (1.88)$$

where the definitions $\overline{\boldsymbol{\Omega}} \equiv \nabla \times \overline{\mathbf{V}}$ and $\overline{\Omega}^2 \equiv \overline{\boldsymbol{\Omega}} \cdot \overline{\boldsymbol{\Omega}}$ are used and the closure coefficients are

$$\begin{aligned} C_\mu = 0.09, \quad C_{\zeta_1} = 1.50, \quad C_{\zeta_2} = 0.40, \quad C_{\zeta_3} = 0.84, \quad C_{\zeta_4} = 2.37, \\ C_{\zeta_5} = 0.70, \quad C_{\zeta_6} = 9.20, \quad 1/\sigma_k = 1.8, \quad 1/\sigma_\zeta = 1.46 \end{aligned} \quad (1.89)$$

It should be noted that although Robinson et al. claim that their definition of ζ has the physical significance of enstrophy, their development does not support that claim mathematically. Rather, their definition of the enstrophy term is simply a change of variables which comes from dividing the dissipation term by the kinematic viscosity.

The k - ε , k - ω , and k - ζ models are often thought of as being fundamentally different. However, a close look at these models shows that they are not fundamentally different. Rather they differ through a change of variables and the closing differential equation for the chosen turbulence parameter.

V. Closure Coefficient Evaluation

Once the fundamental equations for a turbulence model have been developed, closure coefficients can be adjusted to allow the developer to tune the model. In general, closure coefficients are chosen which allow the turbulence model to match experimental results. A wide range of experimental data is available for many flow scenarios. The model developer is free to choose the actual flow scenarios used to evaluate the closure coefficients. However, the resulting model will be tuned to produce more accurate solutions for flows similar to those flow scenarios.

In the ideal situation, the coefficients could be evaluated by isolating each term of the equations and comparing the isolated terms to physical aspects of a given flow scenario. The terms in the governing equations do not appear in a completely isolated form in nature, so empirical relations are often developed by conducting wind tunnel tests, DNS simulations, Large Eddy Simulations (LES), or other computational methods that simulate isolated characteristics of turbulent flow. Once an estimate for the closure coefficients has been obtained for a given flow scenario, computer optimization is often used to refine the final values. Closure coefficients are often altered as new flow scenarios are studied.

Common methods for evaluating model coefficients are based on examination of the behavior of the model in the log layer and the behavior of the model in decaying homogeneous isotropic turbulence. Other flow scenarios sometimes used include vortex stretching, flow separation, and shear-free mixing. A brief overview of a few of these approaches is given here. The discussion will focus on how closure coefficients for the k - ω model could be obtained.

A. The Log Layer

The log layer is the portion of flow near a wall that is close enough to the wall for the velocity term normal to the wall to be ignored, but far enough from the wall that the molecular viscosity is negligible compared to the turbulent eddy viscosity. Extensive experimental data has shown that in the log layer, the nondimensional velocity profile satisfies what is known as the law-of-the-wall

$$\bar{V}_x = \frac{u_\tau}{\kappa} \ln(u_\tau y / \nu) + C \quad (1.90)$$

where the value of the constants are generally accepted as $\kappa = 0.41$ and $C = 5.0$, and $u_\tau \equiv \sqrt{\tau_w / \rho}$ is the conventional friction velocity. From the definition of the friction velocity and the assumption of the constant-stress layer,

$$\nu_t \frac{\partial \bar{V}_x}{\partial y} \cong u_\tau^2 \quad (1.91)$$

Experimental data by Townsend [48] indicate that in the log layer,

$$\frac{\tau_{xy}}{\rho} = -\bar{\tilde{V}}_x \bar{\tilde{V}}_y \cong a_1 \bar{\tilde{V}}^2 \quad (1.92)$$

where $a_1 = 0.15$ is the empirical constant. From the definition of the turbulent-kinetic-energy given in Eq. (1.55), Eq. (1.92) can be written as

$$-\bar{\tilde{V}}_x \bar{\tilde{V}}_y \cong 2a_1 k = \beta_r k \quad (1.93)$$

where $\beta_r = 0.30$ is often called Bradshaw's constant from Bradshaw, Ferris, and Atwell [27]. Applying the near-wall approximations to the Boussinesq hypothesis used to define eddy viscosity in Eq. (1.50) gives a definition for the eddy viscosity in the near-wall region

$$-\overline{\widetilde{V}_x \widetilde{V}_y} \equiv v_t \frac{\partial \overline{V}_x}{\partial y} \quad (1.94)$$

Using Eqs. (1.91), (1.93) and (1.94) gives the relation

$$k \cong \frac{u_\tau^2}{\beta_r} = \frac{u_\tau^2}{0.30} \quad (1.95)$$

This relationship suggests a constant nondimensional value for the turbulent-kinetic-energy in the log layer

$$k^+ \equiv \frac{k}{u_\tau^2} \cong \frac{1}{0.3} \quad (1.96)$$

This relationship can also be used to calibrate a turbulence model.

Applying the log-layer approximations to the k - ω model gives the equations for \overline{V}_x , k , and ω

$$v_t \frac{\partial \overline{V}_x}{\partial y} \cong u_\tau^2 \quad (1.97)$$

$$-\frac{\partial}{\partial y} \left(\frac{v_t}{\sigma_k} \frac{\partial k}{\partial y} \right) = v_t \left(\frac{\partial \overline{V}_x}{\partial y} \right)^2 - C_\mu k \omega \quad (1.98)$$

$$-\frac{\partial}{\partial y} \left(\frac{v_t}{\sigma_\omega} \frac{\partial \omega}{\partial y} \right) = C_{\omega 1} v_t \frac{\omega}{k} \left(\frac{\partial \overline{V}_x}{\partial y} \right)^2 - C_{\omega 2} \omega^2 \quad (1.99)$$

where the turbulent eddy viscosity is

$$v_t = \frac{k}{\omega} \quad (1.100)$$

We wish the solution of this model to match Eq. (1.90). From Eq. (1.90) we find

$$\frac{\partial \overline{V}_x}{\partial y} = \frac{u_\tau}{\kappa y} \quad (1.101)$$

Using Eqs. (1.95), (1.100), and (1.101) in Eq. (1.98) gives

$$\omega = \frac{u_\tau}{\kappa y \sqrt{C_\mu}} \quad (1.102)$$

Using Eqs. (1.100), (1.101), (1.102), and (1.95) in Eq. (1.97) gives

$$C_\mu = \beta_r^2 = (0.30)^2 = 0.09 \quad (1.103)$$

Using these same expressions along with Eq. (1.103) in Eq. (1.99) gives the relation

$$\sigma_\omega = \frac{\kappa^2}{\beta_r(C_{\omega 2}/C_\mu - C_{\omega 1})} \cong \frac{0.683}{(C_{\omega 2}/0.09 - C_{\omega 1})} \quad (1.104)$$

An almost identical development can be followed for the k - ε model to develop relationships for the closure coefficients. Methods for applying the law of the wall to find closure coefficients are presented by Durbin and Pettersson Reif [49] and Wilcox [35].

B. Homogeneous, Isotropic Turbulence

Applying the properties of homogeneous isotropic turbulence to the governing equations is another commonly used tool for finding closure coefficients. Experiments show that the decay of turbulent kinetic energy can be expressed in the form

$$k = \frac{C}{(t+B)^a}, \quad \frac{\partial k}{\partial t} = -\frac{aC}{(t+B)^{a+1}} \quad (1.105)$$

where $a = 1.2 \pm 0.25$. Notice the considerable range in the data. For homogeneous, isotropic flow, there are no spatial gradients of any of the mean flow properties. Applying this assumption to the traditional k - ω model allows Eqs. (1.80) and (1.81) to be reduced to

$$\begin{aligned} \frac{\partial k}{\partial t} &= -C_\mu k \omega \\ \frac{\partial \omega}{\partial t} &= -C_{\omega 2} \omega^2 \end{aligned} \quad (1.106)$$

These equations can be solved to yield

$$\begin{aligned} k &= t^{-C_\mu/C_{\omega 2}} \\ \omega &= \frac{1}{C_{\omega 2} t} \end{aligned} \quad (1.107)$$

Applying this result to Eq. (1.105) gives the relationship $C_\mu/C_{\omega 2} = a$. Using a value of $a = 1.2$ and applying Eq. (1.103) gives

$$C_{\omega 2} = 0.075 \quad (1.108)$$

Note that the traditional value is $C_{\omega 2} = 0.072$. Using Eq. (1.108) in Eq. (1.104) gives

$$\sigma_\omega \cong \frac{0.683}{.833 - C_{\omega 1}} \quad (1.109)$$

Applying the traditional closure coefficient $C_{\omega 1} = 0.52$ gives $\sigma_{\omega} = 2.18$. Applying the traditional closure coefficient $\sigma_{\omega} = 2.0$ gives $C_{\omega 1} = 0.49$.

C. The Log Layer at Separation

The log layer for differential pressure boundary layers can also be examined. Measurements [48] indicate that near separation, the law of the wall is replaced by

$$V_x \rightarrow \frac{1}{0.24} \sqrt{\frac{y}{\rho} \frac{dp}{dx}} \quad (1.110)$$

as $y \rightarrow 0$. Additionally, measurements from Clark [50] and Laufer [51] suggest that empirical relationships can be used to approximate the kinetic energy and velocity near the center of a channel. Zeierman and Wolfshtein [33] use this information in the development of closure coefficients for a turbulence model.

D. Shear-Free Mixing Layer

Another approach to calculating closure coefficients is that used by Briggs, Ferziger, Koseff, and Monismith [52] by examining LES results of a shear-free mixing layer. In this scenario, diffusion terms offset dissipation terms in the governing equations. The authors conclude that the asymptotic behavior of k and ν_t can be described as

$$k \sim ay^{-2.45} \quad (1.111)$$

$$\nu_t \sim by^{-0.42} \quad (1.112)$$

as $y \rightarrow \infty$ where a and b are constants.

It has been shown here that there are several possible methods for evaluating closure coefficients. The method of choice is generally dependent on the purpose of the turbulence model. During the development of any turbulence model, benchmarking cases are used to evaluate the accuracy of the model. This is the topic of the following section.

VI. Common Turbulence Model Evaluation Cases

Common cases used for evaluating a turbulence model include the cases of boundary layer flow, channel flow, pipe flow, plane-jet flow, and round-jet flow. This set of flow scenarios allows a model to be tested for both wall-bounded and free-shear flows. An overview of these testing scenarios is given here. For a more in-depth discussion of each of these flow scenarios as well as sample computations from a CFD algorithm, see Appendix C.

A. Boundary Layer Flow

Boundary layer flow is often one of the first scenarios considered for evaluating a turbulence model. The test case consists of an infinite flat plate placed in-line with the velocity vector of a uniform flow. As the flow advances along the plate, the momentum deficit from the skin drag on the plate causes a boundary layer to develop. The profile of this boundary layer has been studied in great detail and much experimental data exists for this case. Experimental results show that the boundary layer profile is self-similar and a function of the skin friction along the plate.

Because of the behavior of the boundary layer profile, certain approximations can be applied to the RANS equations and the resulting simplified equations can be written in a similarity form. The nature of the boundary layer equations allows a solution to be obtained by directly integrating the equations in space. This solution method allows for the effect of turbulence model closure coefficients to be quickly assessed because any one solution requires only a matter of seconds to compute with modern computers. However, this method suffers from the fact that approximations about the flowfield must be assumed in order to develop the boundary layer equations.

The boundary layer flow case in a two-dimensional CFD solver can be constructed on a rectangular domain as shown in Fig. 1.1. On the west side of the domain a velocity inlet boundary condition is specified. At this boundary condition, all freestream properties of the flow must be specified. In this case, uniform properties of the flow are specified across the inlet boundary. Along the south side of the domain, a no-slip wall boundary condition is specified. Along the north and east sides of the domain, pressure outlet

boundary conditions are specified. Once the solution has reached a converged state, there should be no flow entering the domain from the north and east boundaries.

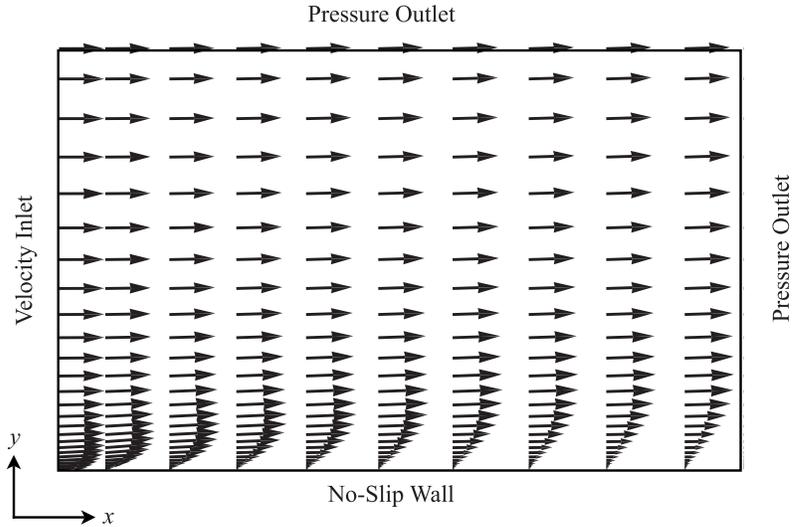


Fig. 1.1 Boundary layer case description.

Two of the most significant parameters of interest in flat-plate boundary layer flow are the prediction of the skin friction along the plate, and the prediction of the nondimensional velocity profile. According to convention, R_x is defined as the Reynolds number at any x -position along the plate where the x -position is measured from the leading edge of the plate along the direction of flow. This Reynolds number can be expressed as

$$R_x \equiv \bar{V}_\infty x / \nu \quad (1.113)$$

Here we define R_L as the Reynolds number based on the length of the plate, L , which can be expressed as

$$R_L \equiv \bar{V}_\infty L / \nu \quad (1.114)$$

The friction coefficient is defined as

$$C_f \equiv 2\tau_w / (\rho \bar{V}_\infty^2) \quad (1.115)$$

Plots of the nondimensional velocity profile, u^+ , as a function of y^+ where $u^+ \equiv \bar{V}_x / u_\tau$ and $y^+ \equiv y u_\tau / \nu$ in comparison with experimental data are often included. There is a wealth of experimental data for boundary

layer flows. For example, results of boundary layer flow can be compared to experimental data by Klebanoff [53], Bradshaw [54], Marusic and Hutchins [55], and Guala, Metzger, and McKeon [56].

B. Fully Developed Channel Flow

Fully developed flow in a channel is a common case for evaluating a turbulence model. This case is a two-dimensional flow which is identical to the three-dimensional flow between two infinite flat plates. As flow enters a channel, a boundary layer develops on each channel wall. As the flow moves downstream, the two boundary layers eventually meet, and the flow reaches the fully developed state.

This flow case in two dimensions can be constructed on a rectangular domain as shown in Fig. 1.2. Along the south side of the domain, a no-slip wall boundary condition is applied. Because this case is symmetrical along the centerline of the channel, only half of the channel is modeled, and a symmetry boundary condition is applied along the north side of the domain. The east and west sides of the domain are then specified as periodic boundary conditions. This boundary condition forces the properties of the flow exiting one side of the domain to be equal to those entering the other side of the domain. The use of periodic boundary conditions requires either a pressure drop across the domain to be specified, or a mass flux across the periodic boundary to be specified.

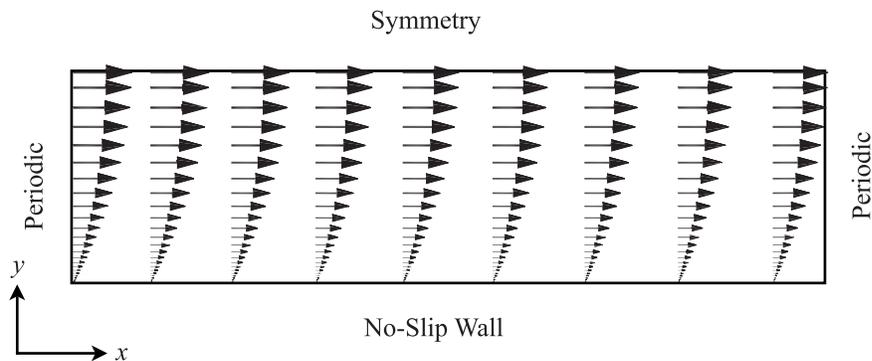


Fig. 1.2 Fully developed channel flow case description.

For fully developed channel flow, gradients in the flow properties with respect to the flow direction disappear, and the profiles of flow properties become dependent only on the coordinate normal to the wall. Therefore, the governing equations can be simplified to a one-dimensional problem and the solution can be

obtained numerically very quickly on modern computers. The formulation can be discretized into a banded system of equations and solved quite rapidly where the full two-dimensional problem may take a significantly longer amount of time. Therefore, this one-dimensional test case is ideal for the optimization process of closure coefficients.

The most significant parameters of interest for the channel flow case are the ability of a model to predict the correct relation between flow Reynolds number and the friction coefficient, and the ability of the model to predict the correct nondimensional velocity profile. The channel Reynolds number is defined as

$$R_e \equiv \bar{V}_{\text{bulk}} D_h / \nu \quad (1.116)$$

where

$$\bar{V}_{\text{bulk}} \equiv \frac{1}{L} \int_{y=0}^L \bar{V}_x dy \quad (1.117)$$

is the bulk velocity, and

$$D_h = 4L \quad (1.118)$$

is the hydraulic diameter based on the channel half-width, L . The Fanning friction factor is defined as

$$C_f \equiv 2\tau_w / (\rho \bar{V}_{\text{bulk}}^2) \quad (1.119)$$

Results of fully developed channel flow can be compared to experimental data by Laufer [57] and Zanoun, Durst, and Nagib [58].

C. Fully Developed Pipe Flow

The fully developed pipe flow case is very closely related to the fully developed channel flow case. As flow enters a pipe, a boundary layer develops along the pipe wall. As the flow moves downstream, the boundary layer eventually fills the entire pipe, and the flow reaches the fully developed state. In this state, there are no gradients in the azimuthal direction of the pipe, and the case can be simplified to a two-dimensional flow in cylindrical coordinates.

This flow case in two dimensions can be constructed on a rectangular domain as shown in Fig. 1.3. Along the north side of the domain, a no-slip wall boundary condition is applied. Because this case is symmetrical along the centerline of the pipe, only half of the pipe is modeled, and a symmetry boundary condition is applied along the south side of the domain. This side of the domain is also the axis of rotation of the pipe. The east and west sides of the domain are then specified as periodic boundary conditions where either a pressure drop across the domain or a mass flux across the periodic boundary must be specified.

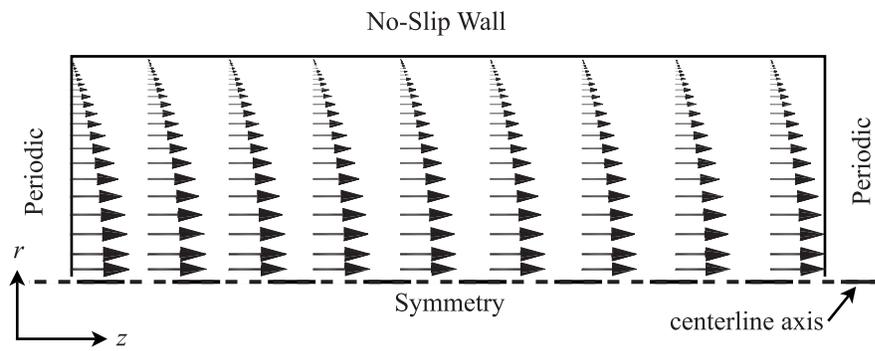


Fig. 1.3 Fully developed pipe flow case description.

For fully developed pipe flow, gradients in the flow properties with respect to the flow direction disappear, and the profiles of flow properties become dependent only on the coordinate normal to the axis of symmetry. Therefore, the governing equations can be simplified to a one-dimensional problem. Because solutions to this case can be quickly obtained, this test case is often used for evaluating closure coefficients.

The most significant parameters of interest for the pipe flow case are the ability of a model to predict the correct relation between flow Reynolds number and the friction coefficient, and the ability of the model to predict the correct nondimensional velocity distribution. The pipe Reynolds number is defined as

$R_e \equiv \bar{V}_{\text{bulk}} D_h / \nu$ where

$$\bar{V}_{\text{bulk}} \equiv \frac{2}{R^2} \int_{-R}^R \bar{V}_z r dr \quad (1.120)$$

is the bulk velocity, and

$$D_h = 2R \quad (1.121)$$

is the hydraulic diameter based on the pipe radius, R . Again, the Fanning friction factor is defined as $C_f \equiv 2\tau_w / (\rho \bar{V}_{\text{bulk}}^2)$. Results of fully developed pipe flow can be compared to experimental data by Nikuradse [59], Laufer [60], and Shockling, Allen, and Smits [61].

D. Plane Jet Flow

The plane jet flow case is a valuable case for testing the ability of the model to predict shear flows. The case consists of a two-dimensional jet of fluid entering a quiescent fluid. As the jet of fluid advances into the quiescent fluid, the momentum of the jet is diffused outward normal to the jet axis. The jet centerline velocity decreases as the flow moves downstream, and the width of the jet grows. Eventually, the core of the jet profile becomes self-similar. The boundary layer equations are often applied to this case in order to develop a similarity solution for the jet profile. However, these equations are based on the assumption that the fluid velocity normal to the jet axis is much smaller than the fluid velocity in the direction of the jet axis. This assumption holds near the center of the jet, but is obviously not correct far from the jet centerline. In fact, in the regions far from the jet centerline, the fluid velocity normal to the axis of the jet is much greater than the fluid velocity in the direction of the jet axis. This is caused from the fluid entrainment surrounding the jet. The fact that the boundary layer equations don't hold far from the jet centerline is seldom mentioned in the literature, and the similarity solution for the jet is often used to evaluate the performance of a turbulence model for the plane jet case. The advantage of using the similarity solution is that it provides a system of equations that can be quickly solved compared to a full two-dimensional CFD model.

This flow case in two dimensions can be constructed on a rectangular domain as shown in Fig. 1.4. Along the west side of the domain, an initial profile for the jet is specified along with inlet conditions for any flow parameters. This initial profile can be taken from the similarity solution for the jet or from experimental data. Along the south side of the domain, a symmetry boundary condition is applied. The east and north sides of the domain are specified as pressure boundary conditions. Figure 1.4 shows the setup for this case in Cartesian coordinates.

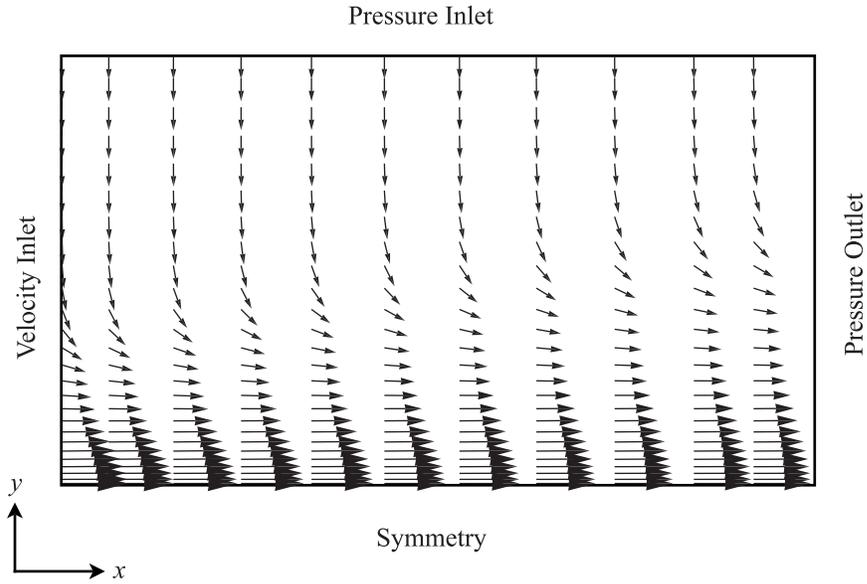


Fig. 1.4 Plane jet flow case description.

Two of the most significant parameters of interest in plane jet flow are the prediction of the jet velocity along the centerline of the jet, and the prediction of the spread rate of the jet. Here we define R_x as the Reynolds number at any x -position measured along the centerline of the jet. This Reynolds number can be expressed as

$$R_x \equiv \left(\frac{36 \bar{M}_x x}{\nu^2} \right)^{1/3} \quad (1.122)$$

where \bar{M}_x is the specific x -momentum flux initiated in the quiescent fluid from the jet and is defined as

$$\bar{M}_x \equiv \int_{-\infty}^{\infty} \bar{V}_x^2 dy \quad (1.123)$$

This value must remain constant across each x cross section of the flow. In order to define a spread rate for the jet, the width of the jet must be defined. Here we define the jet width, y_h , as the y -coordinate at which the velocity in the direction of the jet centerline is equal to half the velocity along the jet centerline

$$\bar{V}_x(x, y_h) = 0.5 \bar{V}_x(x, 0) \quad (1.124)$$

The spread rate can then be evaluated by plotting the jet width as a function of distance along the jet centerline. Results for the plane jet can be compared to data by Bradbury [62] and Heskestad [63].

E. Round Jet Flow

The round jet flow case is very similar to the plane jet, but examines an axisymmetric jet rather than a two-dimensional jet. Just like the plane jet, as the jet of fluid advances into the quiescent fluid, the momentum of the jet is diffused outward normal to the jet axis. The jet centerline velocity decreases as the flow moves downstream, and the radius of the jet grows. Eventually, the core of the jet profile becomes self-similar. Because the jet is axisymmetric, the case can be simplified to a two-dimensional problem in cylindrical coordinates. The boundary layer equations are often applied to this case in order to develop a similarity solution for the jet profile. However, just like the case of the plane jet, these equations do not hold far from the jet centerline.

This flow case in two dimensions can be constructed on a rectangular domain as shown in Fig. 1.5. Along the west side of the domain, an initial profile for the jet is specified along with inlet conditions for any flow parameters. This initial profile can be taken from the similarity solution for the jet or from experimental data. Along the south side of the domain, a symmetry boundary condition is applied. The south side of the domain is the axis of rotation for the case. The east and north sides of the domain are specified as pressure boundary conditions. Figure 1.5 shows the setup for this case in cylindrical coordinates.

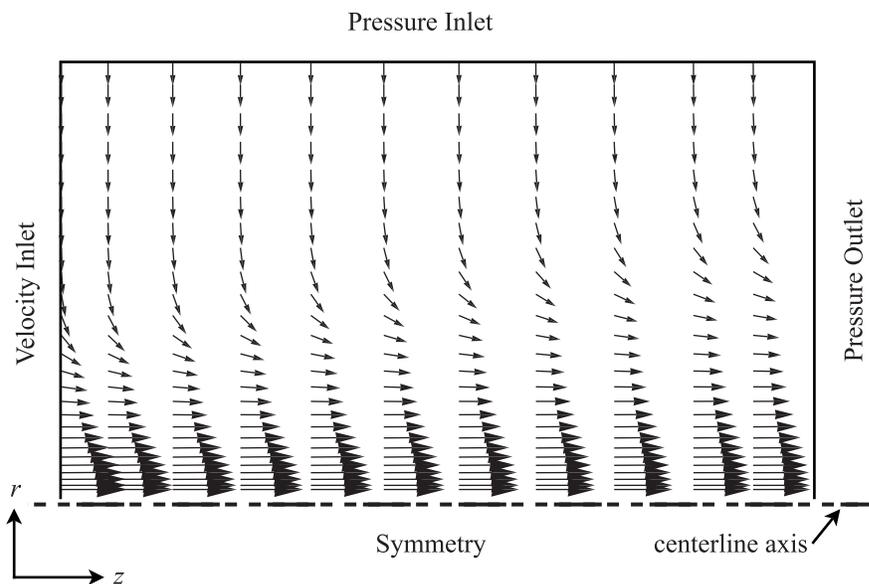


Fig. 1.5 Round jet flow case description.

Two of the most significant parameters of interest in plane jet flow are the prediction of the jet velocity along the centerline of the jet, and the prediction of the spread rate of the jet. Here we define R_z as the Reynolds number at any z -position measured along the centerline of the jet. Results of a turbulence model sometimes include plots of the centerline velocity, \bar{V}_c , as a function of R_z . In order to define a spread rate for the jet, the width of the jet must be defined. Again we define the jet width, r_h , as the radial coordinate at which the velocity in the direction of the jet centerline is equal to half the velocity along the jet centerline

$$\bar{V}_z(z, r_h) = 0.5\bar{V}_z(z, 0) \quad (1.125)$$

The spread rate can then be evaluated by plotting the jet width as a function of distance along the jet centerline. Results for the round jet can be compared to data by Wygnanski and Fiedler [64] and Rodi [65].

Many turbulence models predict that the round jet spreads more quickly than the plane jet. However, this goes against the measured physical characteristics of the jets. The tendency towards this error in turbulence models is known as the round-jet/plane-jet anomaly. Some have tried to correct this anomaly in current turbulence models. For example, Pope [66] suggested a modification to the dissipation equation which alleviates this shortcoming in the k - ϵ model. The round-jet/plane-jet anomaly is an important assessment of a turbulence model.

VII. Summary and Conclusions

Although the laws of Newtonian fluid motion have been understood for quite some time, turbulence modeling is still an unsolved problem. The fundamental equations for laminar flow are the continuity and Navier-Stokes equations which can be extended to the RANS equations for turbulent flow. The RANS equations cannot be closed without modeling the unknown components of the Reynolds stress tensor, and this difficulty presents what is known as the turbulence closure problem. Various methods for closing the RANS equations have been proposed, and the discussion included in this chapter focused on those closure methods that are based on the Boussinesq hypothesis. The Boussinesq hypothesis assumes the Reynolds stress tensor can be modeled as a function of the turbulent kinetic energy and the eddy viscosity of the flow. This replaces the six unknowns in the Reynolds stress tensor with only two unknowns. A transport

equation for the turbulent kinetic energy is traditionally developed from the Reynolds-stress-transport equation. This transport equation introduces one additional unknown, the dissipation of turbulent kinetic energy. Energy-dissipation turbulence models solve for the dissipation using a transport equation that is developed by direct analogy to the turbulent-kinetic-energy equation. The turbulent kinetic energy and dissipation are then related algebraically to the turbulent eddy viscosity, closing the system of equations. The most popular dissipation-based models are the k - ε , k - ω , and k - ζ models.

Closure coefficients for turbulence models have traditionally been developed by tuning the model to match experimental data for specific flow scenarios. Perhaps the most widely used experimental data sets used for this purpose are data sets taken in the log layer and data taken on the decay of turbulent kinetic energy. Once initial estimates for closure coefficients are developed, models are commonly tested for a wide range of flow cases including free-shear flows and wall-bounded flows. These cases allow the versatility of the model to be tested and closure coefficients to be refined.

Although energy-dissipation models such as the k - ε and k - ω models have been widely used for many years, it is well known that these models are sometimes inconsistent and are often unable to capture the true behavior of turbulent flow. This can likely be attributed to the fact that these models have only partially been developed from fundamental physical phenomenon and have been patched and tuned over time to exhibit the proper behavior for specific flows. Several concerns with the development of the traditional energy-dissipation models have been noted by turbulence model developers. These concerns and some possible remedies for these concerns are the topic of the following chapter.

CHAPTER 2

POSSIBLE IMPROVEMENTS IN RANS-BASED TURBULENCE MODELING

I. Introduction

Even though RANS-based turbulence modeling has been a topic of research for many years, it is anything but a closed subject. Several concerns with traditional turbulence modeling have been identified by various researchers, and many concerns have yet to be fully addressed. Some of the most significant concerns with the mathematics and fundamentals of turbulence modeling are listed here along with references to researchers who have noted these concerns.

1. The traditional smooth-wall boundary conditions implemented for dissipation-based models are mathematically incorrect [19,67].
2. The dissipation per unit mass used in the traditional turbulence models is not the true dissipation of turbulent kinetic energy per unit mass [13,35].
3. The traditional dissipation per unit mass actually includes a portion of the total molecular transport term. Therefore, the molecular transport term in the traditional turbulent-kinetic-energy equation neglects a portion of the molecular transport of turbulent kinetic energy [13].
4. Because a portion of the molecular transport is neglected in the traditional turbulent-kinetic-energy equation, subsequent application of Boussinesq's analogy between turbulent and molecular transport results in neglecting a portion of the turbulent transport of turbulent kinetic energy [13].
5. The length scale used to close traditional turbulence models is a dissipation-based length scale associated with the smaller turbulent eddies which have higher strain rates. However, the larger turbulent eddies carry more energy and are primarily responsible for the transport of momentum and energy in a fluid [9,13,35].
6. The dissipation-based length scale used in the development of Eqs. (1.76), (1.82), and (1.87) results in a modeled Reynolds stress tensor that is inversely proportional to the fluid molecular viscosity.

However, the modeled version of the Reynolds stress tensor should not be directly dependent on molecular viscosity, as seen by the definition given in Eq. (1.41) [13].

7. The closing transport equations given by Eqs. (1.75), (1.81), and (1.88) were developed by simple analogy and dimensional analysis and were not developed rigorously from the Navier-Stokes equations [13,35].
8. Few turbulence models are capable of modeling turbulence near a rough wall. However, it is impossible to manufacture a perfectly smooth wall. Therefore, no experimental data exists for perfectly smooth walls, and the use of a model that has been designed for perfectly smooth-walls in the analysis of a real-life scenario is questionable.

The following sections discuss a few of these concerns in more detail and present some preliminary work by others to address them. Note that in subsequent chapters and sections, these concerns may be referenced as a concern number ranging from Concern #1 to Concern #8.

II. Traditional Smooth-Wall Boundary Conditions

The smooth-wall boundary conditions commonly employed on traditional dissipation-based turbulence models are mathematically incorrect. Durbin [19] was the first to point this out and stated, “*These conditions must violate the energy balance...*” However, this statement has seldom been acknowledged in the literature perhaps because it has not been well understood. Today, 20 years later, even commercial implementations of k - ϵ , k - ω , and k - ζ turbulence models still employ smooth-wall boundary conditions that are mathematically incorrect and do not enforce energy conservation. Failure to impose proper wall boundary conditions results in an indeterminate system of equations with an infinite number of solutions. Results obtained for dissipation-based turbulence models using the traditional smooth-wall boundary conditions are dependent on the numerical algorithms used to implement the equations. Such solutions can be highly implementation dependent.

Perhaps the most fundamental flaw in the development of traditional smooth-wall boundary conditions is the extraction of a boundary condition from the differential equation itself. Many of the most widely

implemented models including those of Lam and Bremhorst [68], Launder and Sharma [69], and Wilcox [45] derive boundary conditions for the second turbulence variable, ε or ω , by examining the near-wall behavior of the differential equations. Such an approach is mathematically incorrect because a boundary condition can never be developed from the differential equation itself. A boundary condition is a condition that is imposed on a differential equation. No amount of analysis will ever result in a boundary condition being derived from the differential equation. The near-wall approximation of any transport equation is valid and can be used as an expression appropriately close to a wall. However, it is not a boundary condition and should not be used as such.

This common mistake has led to the implementation of many turbulence models in an indeterminate form because what is referred to as a boundary condition is actually a near-wall approximation of the differential equations. This misconception must be understood by the CFD community if correct turbulence modeling is to be achieved.

III. Turbulent Energy Transport

The turbulent-kinetic-energy transport equation has traditionally been derived from the specific Reynolds stress tensor. In contrast to the traditional development, the turbulent-energy transport equation can also be developed from the mechanical energy equation which is formed by taking the dot product of the velocity vector with the Navier-Stokes equations. This approach alleviates a few of the concerns mentioned regarding the traditional turbulent-kinetic-energy equation and was first taken by Phillips [13]. This new development of the turbulent-kinetic-energy equation has not previously been implemented in a turbulence model. Because this equation forms the foundation of the turbulence model presented in this work, an overview of the development of this equation is included here.

Defining a pseudo hydrostatic pressure that includes the static pressure, hydrostatic head, and viscous normal stresses,

$$\hat{p} \equiv p + \rho g_o Z + \frac{2}{3} \mu \nabla \cdot \mathbf{V} \quad (2.1)$$

the Navier-Stokes equations can be written in vector form as

$$\rho \left[\frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla) \mathbf{V} \right] = -\nabla \hat{p} + \nabla \cdot [2\mu \bar{\bar{\mathbf{S}}}(\mathbf{V})] + g_o Z \nabla \rho \quad (2.2)$$

Taking the dot product of the velocity vector with the Navier-Stokes equations, rearranging, and using mathematical identities gives the mechanical energy equation for a Newtonian fluid

$$\begin{aligned} \rho \left[\frac{\partial}{\partial t} \left(\frac{1}{2} V^2 \right) + \mathbf{V} \cdot \nabla \left(\frac{1}{2} V^2 \right) \right] &= \nabla \cdot \{ \mu [\nabla \left(\frac{1}{2} V^2 \right) + (\mathbf{V} \cdot \nabla) \mathbf{V}] \} \\ &\quad - \mathbf{V} \cdot [\nabla \hat{p} - g_o Z \nabla \rho] - 2\mu \bar{\bar{\mathbf{S}}}(\mathbf{V}) \cdot \bar{\bar{\mathbf{S}}}(\mathbf{V}) \end{aligned} \quad (2.3)$$

Applying Eq. (1.28) and Eq. (1.29) to this general form of the mechanical energy equation and taking the ensemble average of the resulting formulation gives

$$\begin{aligned} \rho \left[\frac{\partial}{\partial t} \left(\frac{1}{2} \bar{V}^2 + k \right) + \bar{\mathbf{V}} \cdot \nabla \left(\frac{1}{2} \bar{V}^2 + k \right) + \overline{\bar{\mathbf{V}} \cdot \nabla (\bar{\mathbf{V}} \cdot \bar{\mathbf{V}})} + \overline{\bar{\mathbf{V}} \cdot \nabla \left(\frac{1}{2} \tilde{V}^2 \right)} \right] \\ = \nabla \cdot \{ \mu [\nabla \left(\frac{1}{2} \bar{V}^2 + k \right) + (\bar{\mathbf{V}} \cdot \nabla) \bar{\mathbf{V}} + \overline{(\tilde{\mathbf{V}} \cdot \nabla) \tilde{\mathbf{V}}}] \} \\ - \bar{\mathbf{V}} \cdot [\nabla \bar{p} - g_o Z \nabla \rho] - \tilde{\mathbf{V}} \cdot \nabla \tilde{p} - 2\mu [\bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) + \overline{\bar{\mathbf{S}}(\tilde{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\tilde{\mathbf{V}})}] \end{aligned} \quad (2.4)$$

where k is defined from Eq. (1.55) as the specific turbulent kinetic energy and the two pressure terms are the mean and fluctuating hydrostatic pressure terms respectively

$$\bar{p} \equiv \bar{p} + \rho g_o Z + \frac{2}{3} \mu \nabla \cdot \bar{\mathbf{V}} \quad (2.5)$$

$$\tilde{p} \equiv \tilde{p} + \frac{2}{3} \mu \nabla \cdot \tilde{\mathbf{V}} \quad (2.6)$$

Using these definitions for pressure terms, the RANS equations in vector form can be written as

$$\rho \left[\frac{\partial \bar{\mathbf{V}}}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) \bar{\mathbf{V}} + \overline{(\tilde{\mathbf{V}} \cdot \nabla) \tilde{\mathbf{V}}} \right] = -\nabla \bar{p} + \nabla \cdot [2\mu \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] + g_o Z \nabla \rho \quad (2.7)$$

Taking the dot product of the mean velocity vector with this form of the Reynolds-averaged Navier-Stokes equations, rearranging, and using mathematical identities gives the mean mechanical energy equation for a Newtonian fluid

$$\begin{aligned} \rho \left[\frac{\partial}{\partial t} \left(\frac{1}{2} \bar{V}^2 \right) + \bar{\mathbf{V}} \cdot \nabla \left(\frac{1}{2} \bar{V}^2 \right) + \bar{\mathbf{V}} \cdot \overline{(\tilde{\mathbf{V}} \cdot \nabla) \tilde{\mathbf{V}}} \right] &= \nabla \cdot \{ \mu [\nabla \left(\frac{1}{2} \bar{V}^2 \right) + (\bar{\mathbf{V}} \cdot \nabla) \bar{\mathbf{V}}] \} \\ &\quad - \bar{\mathbf{V}} \cdot [\nabla \bar{p} - g_o Z \nabla \rho] - 2\mu \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \end{aligned} \quad (2.8)$$

Applying mathematical identities to this equation, subtracting the result from Eq. (2.4), and applying more mathematical identities gives an alternate form of the turbulent-kinetic-energy-transport equation

$$\begin{aligned} \rho \left[\frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) k \right] &= \bar{\bar{\boldsymbol{\tau}}} \cdot \mathbf{J}(\bar{\mathbf{V}}) - 2\mu \overline{[\bar{\bar{\mathbf{S}}}(\tilde{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\tilde{\mathbf{V}}) - \frac{1}{3}(\nabla \cdot \tilde{\mathbf{V}})^2]} + \overline{\tilde{p}(\nabla \cdot \tilde{\mathbf{V}})} \\ &+ \nabla \cdot (\mu \nabla k - \nu \nabla \cdot \bar{\bar{\boldsymbol{\tau}}}) - \nabla \cdot \left[\frac{1}{2} \rho \overline{\tilde{V}^2 \tilde{\mathbf{V}}} + \overline{\tilde{p} \tilde{\mathbf{V}}} + \frac{2}{3} \mu \overline{(\nabla \cdot \tilde{\mathbf{V}}) \tilde{\mathbf{V}}} \right] \end{aligned} \quad (2.9)$$

A close look at this equation reveals that the terms on the right-hand side are the true volumetric production, viscous dissipation, pressure dilation, molecular transport, and the volumetric turbulent transport of turbulent kinetic energy. Note that the only approximation used to develop this equation was that for a Newtonian fluid. Also note that the assumption of constant dynamic viscosity was not made in the development of this governing equation as was done in the development of Eq. (1.62). Applying the Boussinesq hypothesis to this equation gives the Boussinesq-based turbulent-energy-transport equation.

$$\begin{aligned} \rho \left[\frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) k \right] &= 2\mu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \frac{2}{3}(\rho k + \mu_t \nabla \cdot \bar{\mathbf{V}}) \nabla \cdot \bar{\mathbf{V}} \\ &- 2\mu \overline{[\bar{\bar{\mathbf{S}}}(\tilde{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\tilde{\mathbf{V}}) - \frac{1}{3}(\nabla \cdot \tilde{\mathbf{V}})^2]} + \overline{\tilde{p}(\nabla \cdot \tilde{\mathbf{V}})} \\ &+ \nabla \cdot ((\nu + \nu_t / \sigma_k) \{ \rho \nabla k + \frac{2}{3} \nabla(\rho k + \mu_t \nabla \cdot \bar{\mathbf{V}}) - 2 \nabla \cdot [\mu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] \}) \end{aligned} \quad (2.10)$$

Defining $\tilde{\varepsilon}$ as the exact dissipation per unit mass,

$$\tilde{\varepsilon} \equiv 2\nu \overline{[\bar{\bar{\mathbf{S}}}(\tilde{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\tilde{\mathbf{V}}) - \frac{1}{3}(\nabla \cdot \tilde{\mathbf{V}})^2]} \quad (2.11)$$

and neglecting the pressure dilation term

$$\overline{\tilde{p}(\nabla \cdot \tilde{\mathbf{V}})} \cong 0 \quad (2.12)$$

gives a version of the turbulent-energy-transport equation that can be used in a traditional RANS-based turbulence model.

$$\begin{aligned} \rho \left[\frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) k \right] &= 2\mu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \frac{2}{3}(\rho k + \mu_t \nabla \cdot \bar{\mathbf{V}}) \nabla \cdot \bar{\mathbf{V}} \\ &- \rho \tilde{\varepsilon} + \nabla \cdot ((\nu + \nu_t / \sigma_k) \{ \rho \nabla k + \frac{2}{3} \nabla(\rho k + \mu_t \nabla \cdot \bar{\mathbf{V}}) - 2 \nabla \cdot [\mu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] \}) \end{aligned} \quad (2.13)$$

Note that the dissipation term used in this differential equation represents the exact dissipation per unit mass. However, this equation can be used interchangeably with other turbulence models that include a

modeled dissipation term. A close look at Eq. (2.13) reveals that the molecular transport term is not simply a pure gradient diffusion process as is assumed with traditional developments. Therefore, it is probable that this version of the turbulent-energy-transport equation will be more accurate than the traditional equation.

Before leaving the topic of turbulent-energy transport, it is important to note that the turbulent-energy-transport equation can alternatively be written in another significant form. The square of the root-mean-square (RMS) fluctuating vorticity is defined as

$$\tilde{\omega}^2 \equiv \overline{(\nabla \times \tilde{\mathbf{V}}) \cdot (\nabla \times \tilde{\mathbf{V}})} \quad (2.14)$$

A change of variables can be applied to express the exact dissipation term in terms of the RMS fluctuating vorticity

$$\begin{aligned} \tilde{\varepsilon} &\equiv 2\nu[\overline{\tilde{\mathbf{S}}(\tilde{\mathbf{V}}) \cdot \tilde{\mathbf{S}}(\tilde{\mathbf{V}})} - \frac{1}{3}\overline{(\nabla \cdot \tilde{\mathbf{V}})^2}] \\ &= \nu\tilde{\omega}^2 + 4\nu\nabla \cdot (\{\frac{1}{3}\nabla(\rho k + \mu_t \nabla \cdot \bar{\mathbf{V}}) - \nabla \cdot [\mu_t \tilde{\mathbf{S}}(\bar{\mathbf{V}})]\} / \rho) \end{aligned} \quad (2.15)$$

The first term on the second line of the left-hand side of Eq. (2.15) is known as the solenoidal dissipation and will be given the symbol

$$\hat{\varepsilon} \equiv \nu\tilde{\omega}^2 \quad (2.16)$$

Huang, Coleman, and Bradshaw [70] have shown that the dilatational terms in Eq. (2.15) are negligibly small for mean velocities under Mach three. Neglecting the dilatational terms, the turbulent-energy-transport equation can be written in terms of the RMS fluctuating vorticity as

$$\begin{aligned} \rho \left[\frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)k \right] &= 2\mu_t \overline{\tilde{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \tilde{\mathbf{S}}(\bar{\mathbf{V}})} - \frac{2}{3}(\rho k + \mu_t \nabla \cdot \bar{\mathbf{V}})\nabla \cdot \bar{\mathbf{V}} \\ &\quad - \mu\tilde{\omega}^2 - 4\mu\nabla \cdot (\{\frac{1}{3}\nabla(\rho k + \mu_t \nabla \cdot \bar{\mathbf{V}}) - \nabla \cdot [\mu_t \tilde{\mathbf{S}}(\bar{\mathbf{V}})]\} / \rho) \\ &\quad + \nabla \cdot ((\nu + \nu_t / \sigma_k) \{\rho \nabla k + \frac{2}{3}\nabla(\rho k + \mu_t \nabla \cdot \bar{\mathbf{V}}) - 2\nabla \cdot [\mu_t \tilde{\mathbf{S}}(\bar{\mathbf{V}})]\}) \end{aligned} \quad (2.17)$$

Phillips submits that the advantage of applying this change of variables is two-fold. First, vorticity is a well-understood transportable property of a fluid flow field while the turbulent dissipation is not. Second, the divergence of the vorticity of any flow field is always zero. This mathematical property could be beneficial in the development of a vorticity-based-transport equation which is not available in the development of a turbulent-dissipation-transport equation.

A significant difference between traditional two-equation turbulence models and the Phillips turbulence model can be seen in the differences in the modeled turbulent-kinetic-energy equation. The traditional turbulent-kinetic-energy transport equation can be written for steady, incompressible flow as

$$(\bar{\mathbf{V}} \cdot \nabla)k = 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \varepsilon + \nabla \cdot [(\nu + \nu_t / \sigma_k) \nabla k] \quad (2.18)$$

The Phillips turbulent-kinetic-energy transport equation can be written for steady-state, incompressible flow in terms of the RMS fluctuating vorticity as

$$\begin{aligned} (\bar{\mathbf{V}} \cdot \nabla)k = & 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \nu(\bar{\omega}^2 + 4\nabla \cdot \{\frac{1}{3} \nabla k - \nabla \cdot [\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})]\}) \\ & + \nabla \cdot ((\nu + \nu_t / \sigma_k) \{\frac{5}{3} \nabla k - 2\nabla \cdot [\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})]\}) \end{aligned} \quad (2.19)$$

It is instructive to note that the traditional turbulent-kinetic-energy equation was developed by taking the trace of the Reynolds stress transport equation. In that development, the fluid viscosity is assumed constant, which puts use of the Boussinesq hypothesis in question. On the other hand, Phillips's equation was derived with only the assumption of a Newtonian fluid and includes terms that are neglected in the traditional development on grounds that they were small for molecular transport. It is also important to note that the dissipation term in the Phillips development is directly proportional to the fluid viscosity. However, the dissipation term in the traditional development is modeled as being independent of the fluid viscosity. It seems intuitive that the dissipation of turbulent kinetic energy should physically be directly related to the fluid viscosity. These two insights into the differences of the two modeled versions of the turbulent-kinetic-energy equation suggest that the Phillips equation should model the physical aspects of turbulent flow better than the traditional equation.

IV. Turbulent Length Scale

As mentioned in Chapter 1, the length scale commonly employed in the development of traditional turbulence models is that associated with the turbulent dissipation. However, the larger energy-bearing eddies are primarily responsible for the majority of the transport of turbulent energy in a flow. An alternate length scale suggested by Phillips [13] is formulated by examining the fluctuating vorticity of a flow field.

It is important to note that the angular velocity of a fluid element is one-half the local vorticity. For a fluctuating flowfield with a mean velocity of zero, the mean kinetic energy per unit mass is equal to one-half the mean square of the translational velocity. Phillips suggests that the mean kinetic energy per unit mass for this scenario can also be defined as one-half the mean square of the angular velocity multiplied by the square of some length scale. This gives the relation

$$k \equiv \frac{1}{2} \overline{\tilde{\mathbf{V}} \cdot \tilde{\mathbf{V}}} = \frac{1}{2} \Gamma \frac{1}{4} \overline{(\nabla \times \tilde{\mathbf{V}}) \cdot (\nabla \times \tilde{\mathbf{V}})]} l_k^2 \quad (2.20)$$

where l_k is the energy-weighted length scale. This length scale should be an important length scale associated with the turbulent energy. From the definition of the fluctuating vorticity, $\tilde{\omega}^2 \equiv \overline{(\nabla \times \tilde{\mathbf{V}}) \cdot (\nabla \times \tilde{\mathbf{V}})}$, Phillips defines this length scale as

$$l_k \equiv \frac{\sqrt{8k}}{\tilde{\omega}} \quad (2.21)$$

This approach to the important length scale of turbulent flow could provide a significant improvement to the accuracy of a turbulence model. The use of the symbol $\tilde{\omega}$ in these relations should not be confused with the ω used for the traditional k - ω model. Note that ω is related to the dissipation as shown in Eq. (1.78) whereas $\tilde{\omega}$ is the fluctuating vorticity defined in Eq. (2.14). These symbols cannot be used interchangeably.

V. Phillips Energy-Vorticity Model

From the development of Eqs. (2.19) and (2.21), Phillips [13] constructs the core of an energy-vorticity turbulence model. Using Eq. (2.21) as the length scale in the definition of the turbulent eddy viscosity, and combining this new definition of turbulent eddy viscosity with Eq. (2.19) gives the two fundamental equations of Phillips' energy-vorticity model

$$\nu_t = C_\nu \frac{k}{\tilde{\omega}} \quad (2.22)$$

$$\begin{aligned} \frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)k &= 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \nu(\tilde{\omega}^2 + 4\nabla \cdot \{\frac{1}{3}\nabla k - \nabla \cdot [v_t \bar{\mathbf{S}}(\bar{\mathbf{V}})]\}) \\ &+ \nabla \cdot ((\nu + \nu_t/\sigma_k) \{\frac{5}{3}\nabla k - 2\nabla \cdot [v_t \bar{\mathbf{S}}(\bar{\mathbf{V}})]\}) \end{aligned} \quad (2.23)$$

where C_ν and σ_k are closure constants.

The most important contributions of the Phillips development are first, the inclusion of the last term in Eq. (2.23) and second, the proposition of using the local energy-weighted turbulent length scale found in Eq. (2.21) to define the eddy viscosity. These contributions have not been included in other turbulence models and could potentially improve the accuracy of RANS-based turbulence modeling.

The reader may note that the model proposed by Phillips lacks a closing relation for the RMS fluctuating vorticity. For most two-equation turbulence models, the final closing equation is developed by analogy with the turbulent-kinetic-energy transport equation but is not rigorously developed from physics. Several options exist for closing the Phillips k -vorticity model, and a short explanation of some of these methods is included here.

A. RMS Turbulent Vorticity Closure: A k - $\tilde{\omega}$ Model

Perhaps the simplest approach to closing the Phillips vorticity model is to model the RMS turbulent vorticity in terms of the mean fluid velocity, the specific turbulent kinetic energy, and the turbulent eddy viscosity. The RMS fluctuating vorticity is defined by Eq. (2.14). By analogy with Eq. (2.23), a turbulent-vorticity transport equation can be obtained. This results in a model defined by

$$\nu_t = C_\nu \frac{k}{\tilde{\omega}} \quad (2.24)$$

$$\begin{aligned} \frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)k = & 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \nu(\tilde{\omega}^2 + 4\nabla \cdot \{\frac{1}{3}\nabla k - \nabla \cdot [v_t \bar{\mathbf{S}}(\bar{\mathbf{V}})]\}) \\ & + \nabla \cdot ((\nu + \nu_t/\sigma_k) \{\frac{5}{3}\nabla k - 2\nabla \cdot [v_t \bar{\mathbf{S}}(\bar{\mathbf{V}})]\}) \end{aligned} \quad (2.25)$$

$$\frac{\partial \tilde{\omega}}{\partial t} + \bar{\mathbf{V}} \cdot \nabla \tilde{\omega} = 2C_{\tilde{\omega}1} \nu_t \frac{\tilde{\omega}}{k} \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - C_{\tilde{\omega}2} \tilde{\omega}^2 + \nabla \cdot [(\nu + \nu_t/\sigma_{\tilde{\omega}})\nabla \tilde{\omega}] \quad (2.26)$$

where C_ν , σ_k , $C_{\tilde{\omega}1}$, $C_{\tilde{\omega}2}$, and $\sigma_{\tilde{\omega}}$ are the closure constants for the model and need to be evaluated.

This formulation can be directly recast in terms of enstrophy. The enstrophy is equal to the square of the fluctuating vorticity.

$$\zeta \equiv \tilde{\omega}^2 \quad (2.27)$$

Using this change of variables, the formulation can be written

$$\nu_t = C_\nu k / \zeta^{1/2} \quad (2.28)$$

$$\begin{aligned} \frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)k &= 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \nu(\zeta + 4\nabla \cdot \{ \frac{1}{3} \nabla k - \nabla \cdot [\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}})] \}) \\ &+ \nabla \cdot ((\nu + \nu_t / \sigma_k) \{ \frac{5}{3} \nabla k - 2\nabla \cdot [\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}})] \}) \end{aligned} \quad (2.29)$$

$$\begin{aligned} \frac{\partial \zeta}{\partial t} + \bar{\mathbf{V}} \cdot \nabla \zeta &= 4C_{\omega 1} \nu_t \frac{\zeta}{k} \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - 2C_{\omega 2} \zeta^{3/2} \\ &+ \nabla \cdot [(\nu + \nu_t / \sigma_\omega) \nabla \zeta] - \frac{1}{2} (\nu + \nu_t / \sigma_\omega) \nabla \zeta \cdot \nabla \zeta / \zeta \end{aligned} \quad (2.30)$$

where the closure coefficients are the same as those needed for Eqs. (2.24)–(2.26).

B. Solenoidal Dissipation Closure: A k - $\hat{\varepsilon}$ Model

Another approach to closing the formulation is to model the solenoidal dissipation by relating it to the enstrophy. A modeled version of the turbulent dissipation is given in Eq. (1.68). By direct analogy with this traditional turbulent dissipation equation, a transport equation for the solenoidal dissipation can be written

$$\frac{\partial \hat{\varepsilon}}{\partial t} + \bar{\mathbf{V}} \cdot \nabla \hat{\varepsilon} = 2C_{\hat{\varepsilon} 1} \nu_t \frac{\hat{\varepsilon}}{k} \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - C_{\hat{\varepsilon} 2} \frac{\hat{\varepsilon}^2}{k} + \nabla \cdot [(\nu + \nu_t / \sigma_{\hat{\varepsilon}}) \nabla \hat{\varepsilon}] \quad (2.31)$$

The enstrophy is related to the solenoidal dissipation through the relation

$$\hat{\varepsilon} = \nu \zeta \quad (2.32)$$

Using this relation, the solenoidal dissipation transport equation can be recast in terms of the enstrophy.

This results in the model

$$\nu_t = C_\nu k / \zeta^{1/2} \quad (2.33)$$

$$\begin{aligned} \frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)k &= 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \nu(\zeta + 4\nabla \cdot \{ \frac{1}{3} \nabla k - \nabla \cdot [\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}})] \}) \\ &+ \nabla \cdot ((\nu + \nu_t / \sigma_k) \{ \frac{5}{3} \nabla k - 2\nabla \cdot [\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}})] \}) \end{aligned} \quad (2.34)$$

$$\frac{\partial \zeta}{\partial t} + \bar{\mathbf{V}} \cdot \nabla \zeta = 2C_{\zeta 1} \nu_t \frac{\zeta}{k} \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - C_{\zeta 2} \nu \frac{\zeta^2}{k} + \nabla \cdot [(\nu + \nu_t / \sigma_\zeta) \nabla \zeta] \quad (2.35)$$

where C_ν , σ_k , $C_{\zeta 1}$, $C_{\zeta 2}$, and σ_ζ are the closure constants for the model and may need to be reevaluated from those used in the traditional dissipation transport equation because the solenoidal dissipation, $\hat{\varepsilon}$, differs from the traditional definition of dissipation, ε .

C. DNS Solenoidal Dissipation Closure: A k - $\hat{\varepsilon}$ Model

Another possibility to closing the Phillips vorticity-based model is to use the results of a recently developed DNS-based solenoidal-dissipation model by Kreuzinger, Friedrich, and Gatski [71] that has provided good agreement with DNS results. The solenoidal-dissipation-transport equation for incompressible flow is given in this study as

$$\begin{aligned} \frac{\partial \hat{\varepsilon}}{\partial t} + \bar{\mathbf{V}} \cdot \nabla \hat{\varepsilon} = & 2C_{\varepsilon 1} \nu_t \frac{\hat{\varepsilon}}{k} \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - C_{\varepsilon 2} \frac{1}{k} (\hat{\varepsilon} + 4\nu \nabla \cdot \{\frac{1}{3} \nabla k - \nabla \cdot [v_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})]\})^2 \\ & - \nu \nu_t \{C_{\varepsilon 3} (\nabla^2 \bar{\mathbf{V}}) + (C_{\varepsilon 4}/k) [(\nabla k) \cdot \nabla] \bar{\mathbf{V}}\} \cdot (\nabla^2 \bar{\mathbf{V}}) + \nabla \cdot [(v + \nu_t/\sigma_{\varepsilon}) \nabla \hat{\varepsilon}] \end{aligned} \quad (2.36)$$

This equation can be recast using the change of variables from solenoidal dissipation to enstrophy, and the resulting formulation is written as

$$\nu_t = C_\nu k / \zeta^{1/2} \quad (2.37)$$

$$\begin{aligned} \frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) k = & 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \nu (\zeta + 4\nabla \cdot \{\frac{1}{3} \nabla k - \nabla \cdot [v_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})]\}) \\ & + \nabla \cdot ((v + \nu_t/\sigma_k) \{\frac{5}{3} \nabla k - 2\nabla \cdot [v_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})]\}) \end{aligned} \quad (2.38)$$

$$\begin{aligned} \frac{\partial \zeta}{\partial t} + \bar{\mathbf{V}} \cdot \nabla \zeta = & 2C_{\zeta 1} \nu_t \frac{\zeta}{k} \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - C_{\zeta 2} \frac{\nu}{k} (\zeta + 4\nabla \cdot \{\frac{1}{3} \nabla k - \nabla \cdot [v_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})]\})^2 \\ & - \nu_t \{C_{\zeta 3} (\nabla^2 \bar{\mathbf{V}}) + (C_{\zeta 4}/k) [(\nabla k) \cdot \nabla] \bar{\mathbf{V}}\} \cdot (\nabla^2 \bar{\mathbf{V}}) + \nabla \cdot [(v + \nu_t/\sigma_\zeta) \nabla \zeta] \end{aligned} \quad (2.39)$$

where C_ν , σ_k , $C_{\zeta 1}$, $C_{\zeta 2}$, $C_{\zeta 3}$, $C_{\zeta 4}$ and σ_ζ are the closure constants for the model and need to be evaluated. The closure coefficients $C_{\zeta 1}$, $C_{\zeta 2}$, $C_{\zeta 3}$, $C_{\zeta 4}$ and σ_ζ could possibly be taken from those given in the study.

D. General Enstrophy Closure: A k - ζ Model

A close look at the previous three closure methods suggests a general model which would encompass the three models. This formulation is written

$$\nu_t = C_\nu k / \zeta^{1/2} \quad (2.40)$$

$$\begin{aligned} \frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) k = & 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \nu (\zeta + 4\nabla \cdot \{\frac{1}{3} \nabla k - \nabla \cdot [v_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})]\}) \\ & + \nabla \cdot ((v + \nu_t/\sigma_k) \{\frac{5}{3} \nabla k - 2\nabla \cdot [v_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})]\}) \end{aligned} \quad (2.41)$$

$$\begin{aligned}
\frac{\partial \zeta}{\partial t} + \bar{\mathbf{V}} \cdot \nabla \zeta &= C_1 \nu_t \frac{\zeta}{k} \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - C_2 \zeta^{3/2} \\
&\quad - C_3 \frac{\nu}{k} (\zeta + C_4 \nabla \cdot \{ \frac{1}{3} \nabla k - \nabla \cdot [\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] \})^2 \\
&\quad - \nu_t \{ C_5 (\nabla^2 \bar{\mathbf{V}}) + (C_6/k) [(\nabla k) \cdot \nabla] \bar{\mathbf{V}} \} \cdot (\nabla^2 \bar{\mathbf{V}}) \\
&\quad + \nabla \cdot [(\nu + \nu_t / \sigma_\zeta) \nabla \zeta] - C_7 (\nu + \nu_t / \sigma_\zeta) \nabla \zeta \cdot \nabla \zeta / \zeta
\end{aligned} \tag{2.42}$$

where C_ν , σ_k , σ_ζ , and C_1 through C_7 are the closure constants for the model and need to be evaluated. Notice that Eq. (2.30) is a special case of Eq. (2.42) when $C_1 = 4C_{\bar{\omega}1}$, $C_2 = 2C_{\bar{\omega}2}$, $C_7 = 0.5$, $\sigma_\zeta = \sigma_{\bar{\omega}}$, and $C_3 = C_4 = C_5 = C_6 = 0$. Equation (2.35) is a special case of Eq. (2.42) when $C_1 = 2C_{\zeta1}$, $C_3 = C_{\zeta2}$, and $C_2 = C_4 = C_5 = C_6 = C_7 = 0$. Equation (2.39) is a special case of Eq. (2.42) when $C_1 = 2C_{\zeta1}$, $C_3 = C_{\zeta2}$, $C_4 = 4.0$, $C_5 = C_{\zeta3}$, $C_6 = C_{\zeta4}$, and $C_2 = C_7 = 0$.

E. Robinson Enstrophy-Transport Closure: A k - ζ Model

Another approach to closing the vorticity-based model is to implement the closing formulation given by Robinson et al. [46,47]. Using Eq. (1.88) as the closing equation gives the model

$$\nu_t = C_\nu k / \zeta^{1/2} \tag{2.43}$$

$$\begin{aligned}
\frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) k &= 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \nu (\zeta + 4\nabla \cdot \{ \frac{1}{3} \nabla k - \nabla \cdot [\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] \}) \\
&\quad + \nabla \cdot ((\nu + \nu_t / \sigma_k) \{ \frac{5}{3} \nabla k - 2\nabla \cdot [\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] \})
\end{aligned} \tag{2.44}$$

$$\begin{aligned}
\frac{\partial \zeta}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) \zeta &= [C_{\zeta1} \zeta / \bar{\bar{\Omega}}^2 - C_{\zeta2} \nu_t^2 \bar{\bar{\Omega}} / \nu k - C_{\zeta3} \nu_t \zeta / k \bar{\bar{\Omega}}] \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot (\bar{\bar{\Omega}}) \\
&\quad + C_{\zeta2} \nu_t \bar{\bar{\Omega}}^3 / 3\nu + C_{\zeta3} \zeta \bar{\bar{\Omega}} / 3 - C_{\zeta4} \nu \zeta^2 / k + C_{\zeta5} (\nu_t \zeta / k) \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \\
&\quad - 4\nu_t \bar{\bar{\mathbf{S}}}(\bar{\bar{\Omega}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\bar{\Omega}}) - (\nabla \times \bar{\bar{\Omega}}) \cdot \{ 2\nabla \cdot [\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] + \frac{1}{3} \nabla k \} \\
&\quad + C_{\zeta6} [(\nu_t / k) \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\Omega}} - \frac{1}{3} \bar{\bar{\Omega}}] \cdot (\nabla k \times \nabla \zeta) / \bar{\bar{\Omega}}^2 + \nabla \cdot [(\nu + \nu_t / \sigma_\zeta) \nabla \zeta]
\end{aligned} \tag{2.45}$$

where C_ν , σ_k , $C_{\zeta1}$, $C_{\zeta2}$, $C_{\zeta3}$, $C_{\zeta4}$, $C_{\zeta5}$, $C_{\zeta6}$, and σ_ζ are the closure constants for the model and would need to be evaluated. The closure coefficients in the turbulent-enstrophy-transport equation may initially be taken from those given in the published model but would likely need to be tuned to the current model.

F. Mean Vortex Wavelength Closure: A k - λ Model

Another possible approach to closing the Phillips energy-vorticity model is based on the analogy between turbulent diffusion and molecular diffusion. Just as the molecular mean-free path is used to model molecular viscosity, a mean vortex wavelength can be used to model the turbulent eddy viscosity. Experimental research by Anderson, Kays, and Moffat [72] has shown that the turbulent mixing length varies nearly linearly in the near-wall region of a flow, and is nearly constant in the bulk flow region for pipe flow. Therefore, for the case of pipe flow, an algebraic expression for the mean vortex wavelength could be developed which varies linearly near a wall and reaches a constant in the bulk flow region. In a sense, this closure method is similar to that taken by Prandtl in the development of his mixing-length model. Prandtl hypothesized that the mixing length was proportional to the distance from the wall, and this hypothesis led to a very successful model for near-wall turbulence that has been widely used.

Using the mean vortex wavelength, λ , as the length scale in the turbulent eddy viscosity gives

$$v_t = \lambda \sqrt{k} \quad (2.46)$$

Setting this relationship equal to that given in Eq. (2.22) gives a relationship between the mean vortex wavelength and the RMS fluctuating vorticity

$$\lambda \equiv C_v \frac{\sqrt{k}}{\bar{\omega}} = C_v \sqrt{\frac{\overline{\tilde{\mathbf{V}} \cdot \tilde{\mathbf{V}}}}{2(\nabla \times \tilde{\mathbf{V}}) \cdot (\nabla \times \tilde{\mathbf{V}})}} \quad (2.47)$$

Using this relationship in Eq. (2.23) gives the turbulent kinetic energy equation in terms of λ

$$\begin{aligned} \frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)k &= 2v_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \nu(C_\lambda k/\lambda^2 + 4\nabla \cdot \{\frac{1}{3}\nabla k - \nabla \cdot [v_t \bar{\mathbf{S}}(\bar{\mathbf{V}})]\}) \\ &+ \nabla \cdot ((\nu + v_t/\sigma_k)\{\frac{5}{3}\nabla k - 2\nabla \cdot [v_t \bar{\mathbf{S}}(\bar{\mathbf{V}})]\}) \end{aligned} \quad (2.48)$$

where C_λ is a closure coefficient. For the case of pipe flow, a closing relation for the mean vortex wavelength could take an algebraic form

$$\lambda = f(y) \quad (2.49)$$

Such a method is described in more detail in Chapter 7.

VI. Closing Transport Equations

It is widely acknowledged that the closing transport equations given in Eqs. (1.75), (1.81), and (1.88) are not derived rigorously from the Navier-Stokes equations. Instead, closing equations are generally developed by analogy with the turbulent-kinetic-energy equation and/or by dimensional analysis. In the ideal case, a turbulence model would be written entirely in terms of turbulence properties for which the governing equations were known and well understood. However, this has not yet been achieved, and a second equation must generally be developed in order to close a RANS-based turbulence model. Energy-dissipation models such as the traditional k - ε , k - ω , and k - ζ models relate the turbulent eddy viscosity to the dissipation of turbulent kinetic energy algebraically, and then attempt to model the dissipation property as a transport property of the flow. Because a transport equation for the dissipation would be difficult at best to develop, this closing equation is generally developed by dimensional analysis with the turbulent-kinetic-energy equation. This presents two drawbacks. First, the equation used to close the model is not rigorously derived from physics. Second, because dissipation is not a transportable property, attempting to model the dissipation using a transport equation that is analogous to the transport of turbulent kinetic energy is a grievous error.

Although the work of Phillips presented in the previous sections does not circumvent the difficulty in developing a closing equation for the second turbulence variable, the models suggested by Phillips attempt to model transportable properties with transport equations. In other words, the closing transport equations suggested by Phillips are no more rigorously derived from physics than those used in the traditional dissipation-based models. However, the turbulent parameters which the transport equations are designed to model can be expected to behave according to a transport equation. Therefore, from a physical standpoint, the models suggested by Phillips are likely more viable than traditional dissipation-based models.

VII. Rough-Wall Turbulence Modeling

It appears that most turbulence models have been developed initially for smooth walls and that the rough-wall case has traditionally been seen as a more difficult case to which the model could be extended once the smooth-wall behavior of the model was well understood. In practice, the smooth-wall equations and constants considered to be the fundamental model are commonly modified to match experimental data for roughened walls. However, it may be that this approach has hindered fundamental understanding of turbulence near a wall.

Nikuradse [59] conducted a phenomenal amount of experimental research exploring the effects of roughness on pipe flow. He correlated his results according to a relation which will be called the Nikuradse number, $N_i \equiv 2.0 \log_{10}(7.4R/k_s) - 1/\sqrt{4C_f}$, in this work. The Nikuradse number is a function of the friction coefficient, C_f , and roughness height, k_s , and was correlated with a parameter called the roughness Reynolds number, $k_s^+ \equiv u_\tau k_s / \nu$. It was shown that at low roughness Reynolds numbers, the experimental data for the Nikuradse number asymptotically approaches a logarithmic relationship with the roughness Reynolds number. At high roughness Reynolds numbers, the Nikuradse number is independent of roughness Reynolds number. Figure 2.1 shows a plot of his experimental data along with lines representing the asymptotes at low and high roughness Reynolds numbers.

These results show that for nearly smooth walls or flows at low Reynolds numbers, the friction coefficient is a strong function of Reynolds number and roughness, while for very rough walls or flows at high Reynolds numbers, the friction coefficient is independent of the Reynolds number. This is also easily observed in the well-known Moody diagram, which was based on the findings of Nikuradse [59] and Colebrook [73].

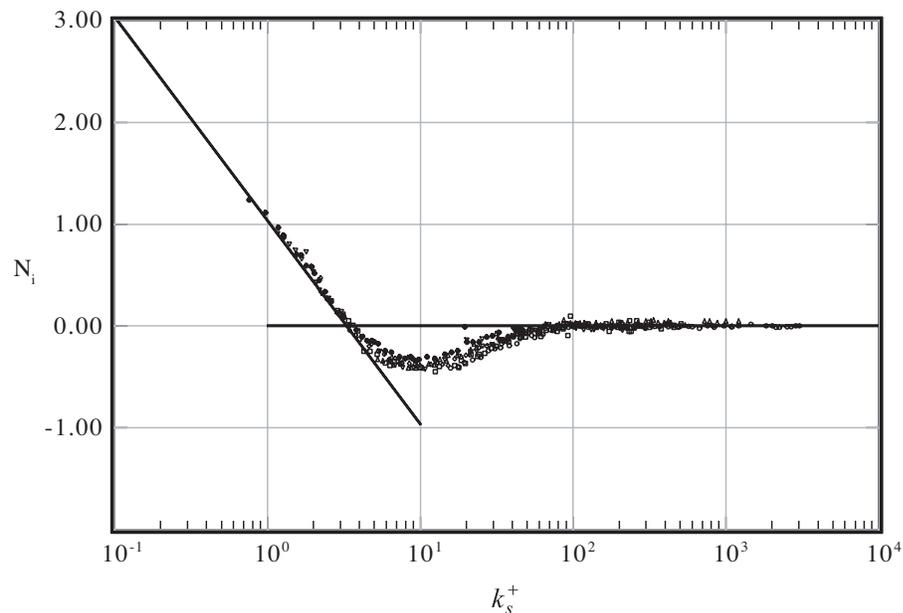


Fig. 2.1 Correlation between friction coefficient and roughness Reynolds number by Nikuradse [59].

With this information on the effects of roughness in mind, the case for first developing a model capable of modeling roughness at high roughness Reynolds numbers and approaching a model that can handle a smooth wall may be more intuitive. A more robust turbulence model may be developed by focusing on the model behavior for high roughness Reynolds numbers where at least the friction coefficient (if not other flow properties) is independent of Reynolds number. Once the model is robust in this region, transition to lower roughness Reynolds numbers can be considered, with the “smooth” or “hydraulically smooth” wall being an asymptote or a special case of the rough wall. To the author’s knowledge, this approach has not been taken before in the development of a turbulence model.

From a physical point of view, a rough wall is the more general case and what is termed a “smooth” or “hydraulically smooth” wall is simply the asymptotic behavior of the wall as the roughness height approaches zero. In reality, a perfectly smooth wall has never been manufactured. Therefore, all experimental data that has been taken near walls is inherently data taken for a rough wall. This may provide some justification for starting with the general case of a rough wall and working towards a model capable of correctly predicting flow near a hydraulically smooth wall.

VIII. Summary and Conclusions

Several significant concerns with traditional RANS-based turbulence modeling have been identified and were listed at the beginning of this chapter along with researchers who have mentioned the concerns. These concerns include the fact that the traditional smooth-wall boundary conditions used for dissipation-based turbulence models are incorrect, the traditional turbulent-kinetic-energy equation derivation assigns definitions and makes assumptions that are incorrect or unnecessary, the length scale used for traditional RANS-based modeling is the length scale associated with the dissipation of turbulent kinetic energy rather than the transport of turbulent kinetic energy, the second transport equation used in traditional energy-dissipation models is not derived rigorously from the Navier-Stokes equations, and few turbulence models are capable of modeling rough-wall effects. Each of these concerns was discussed in some detail throughout the chapter. The fact that these concerns are rooted deeply in fundamental physical and mathematical laws suggests that sufficiently addressing these concerns may result in significant improvements in RANS-based turbulence modeling.

Many of these fundamental concerns of turbulence modeling have been acknowledged for some time and past work has been initiated to address them. Durbin [19] mentioned the inconsistency related to the smooth-wall boundary conditions used for traditional turbulence models roughly 20 years ago. Phillips [13] performed extensive work in developing an alternate transport equation for the turbulent kinetic energy, which alleviates the concerns related to the development and definitions used in the traditional equation. Additionally, he has suggested an alternate length scale that is related to the energy-bearing eddies rather than those eddies associated with the dissipation. These fundamental developments of Phillips can be combined to form the basis of an energy-vorticity model which can be closed in terms of a $k-\hat{\epsilon}$, $k-\tilde{\omega}$, $k-\zeta$, or $k-\lambda$ model. This previous work forms the foundation of the work of this dissertation.

CHAPTER 3

OUTLINE OF THE PRESENT WORK

Several significant concerns with current turbulence models were discussed in the previous chapter. Some of the most notable concerns are that 1) the smooth-wall boundary conditions for traditional turbulence models are incorrect, 2) most turbulence models are incapable of modeling turbulent flow near rough walls, 3) the dissipation per unit mass used in the traditional turbulence models is not the true dissipation per unit mass, and 4) the length scale used in traditional models is the length scale associated with dissipation, not the length scales of the larger eddies which cause most of the momentum and energy transport. The work of this dissertation builds on the work of Durbin and Phillips to more fully address the concerns mentioned here. More specifically, the work of this dissertation includes:

1. implementing the correct smooth-wall boundary conditions suggested by Durbin [19] for dissipation-based turbulence models,
2. evaluating the behavior of the Phillips energy-vorticity model in the presence of perfectly smooth walls,
3. implementing one possible k - λ closure of the Phillips energy-vorticity model for fully rough pipe flow,
4. determining appropriate values for the closure constants of the Phillips k - λ model, and
5. comparing numerical results of the Phillips k - λ model to numerical results of other models as well as experimental data.

Each of these items is addressed in detail in the following chapters. Chapter 4 gives an overview of the numerical methods used for the research which include finite-difference, finite-volume, and direct numerical integration methods as well as one-dimensional and two-dimensional implementations. Chapter 5 discusses the smooth-wall boundary conditions for dissipation-based turbulence models and includes a discussion on the mathematical implications of applying incorrect boundary conditions. It also

presents results from traditional turbulence models once the correct boundary conditions have been applied. Chapter 6 presents the characteristics of two possible closure methods for the Phillips energy-vorticity model in the presence of a perfectly smooth wall, and discusses the physical implications of the models. Chapter 7 gives a detailed development of the Phillips k - λ model for fully rough pipe flow and preliminary results of the model. Chapter 8 presents the optimization techniques used to evaluate the closure coefficients for the Phillips k - λ model. The results of the model are shown in Chapter 9 and compared to experimental results and empirical relations. Chapter 9 also includes the conclusions of the work and suggests area for future work.

CHAPTER 4

COMPUTATIONAL FLUID DYNAMICS METHODS

I. Introduction

Turbulent fluid flow is commonly modeled using numerical algorithms called computational fluid dynamics (CFD) methods. In general, CFD codes use discretization methods to divide up the flow domain into what are termed “cells” or “elements” to which the flow equations can be applied. Many CFD codes have been developed by various researchers and organizations. For example, Fluent is a commercial CFD solver sold by Ansys, CFL3D is a CFD code developed using government funding which can be used by approved citizens of the United States, and OpenFOAM is an open-source CFD code that has recently gained popularity. The work of this dissertation could have been performed by altering public or open-source codes. However, the author chose to write CFD codes from the ground up rather than attempting to alter more traditional codes so that the fundamentals of the code would be understood and could readily be changed. Several CFD codes were written during the course of this work in order to test and implement various models and flows. These codes include a one-dimensional finite difference solver, a one-dimensional finite-volume solver, a two-dimensional finite-volume solver, and a one-dimensional Runge-Kutta integration solver. This chapter gives a brief overview of these codes and includes sample results.

II. One-Dimensional Finite-Difference Formulation

A one-dimensional finite-difference method was written to model fully developed flow in a pipe or channel. The original version of this method was second order and is detailed in Appendix D. Also included in the appendix are methods for developing higher-order differencing algorithms. The finite-difference methods used in this work can be used on non-uniform grids, and a second-order version has been used to obtain solutions to the $k-\omega$ model of Wilcox [45] for the case of fully developed pipe flow. Figure 4.1 shows results of a grid convergence study for this case.

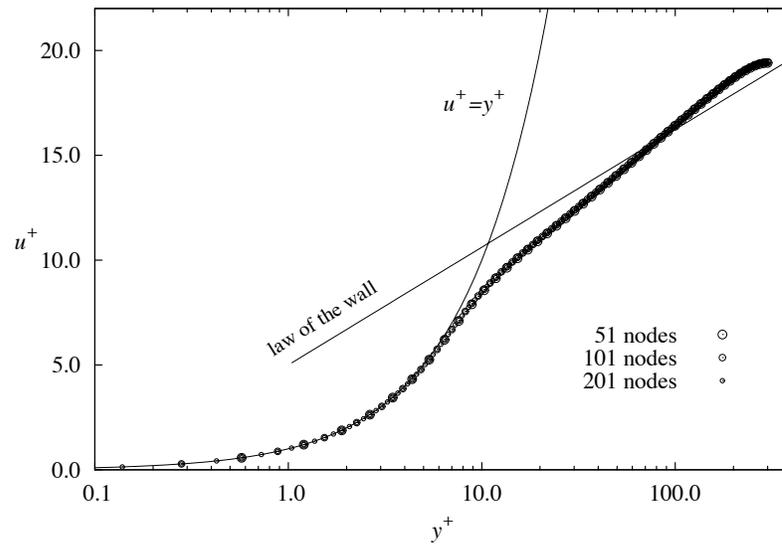


Fig. 4.1 Grid convergence results for Wilcox 1998 k - ω model for fully developed pipe flow.

Figure 4.2 shows the grid-converged results of the code for the case described above and includes the results of the same case using code written by Wilcox. Note that Wilcox's code required considerably more nodes to reach grid convergence. This is largely because his code is not second order.

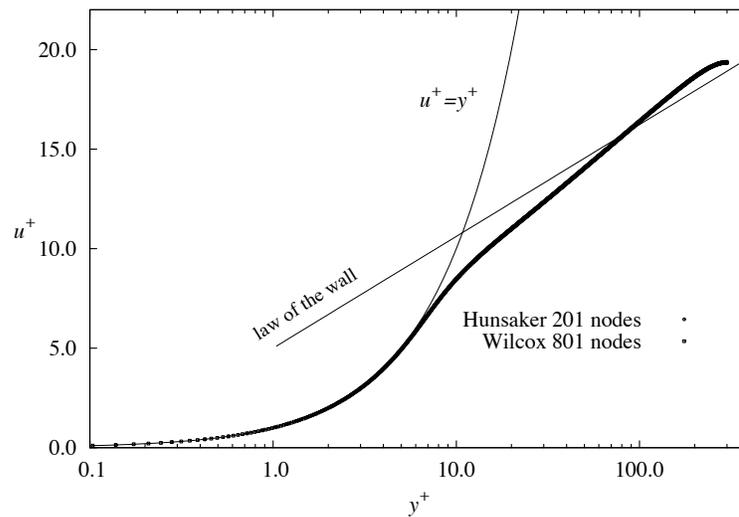


Fig. 4.2 Grid-converged results of codes written by Hunsaker and Wilcox for the Wilcox 1998 k - ω model for fully developed pipe flow.

A sixth-order code was required to model the near-smooth-wall behavior of Phillips's energy-vorticity model. The method for developing a sixth-order finite difference code is included in Appendix D along with sample code for the algorithm.

III. One-Dimensional Finite-Volume Formulation

Finite-volume codes are commonly employed in the CFD solution methods of two- or three-dimensional problems. In order to work out some inconsistencies in a two-dimensional finite-volume solver, a one-dimensional finite-volume code was developed for the case of fully developed channel flow. The details of this algorithm are included in Appendix E. Four two-equation turbulence models were incorporated into the one-dimensional finite-volume code including the $k-\varepsilon$ model by Lam and Bremhorst [68], the $k-\varepsilon$ model by Launder and Sharma [69], the traditional $k-\omega$ model, and the $k-\omega$ model of Wilcox [45]. A comparison of the grid-converged results of these models for the case of fully developed channel flow is shown in Figure 4.3.

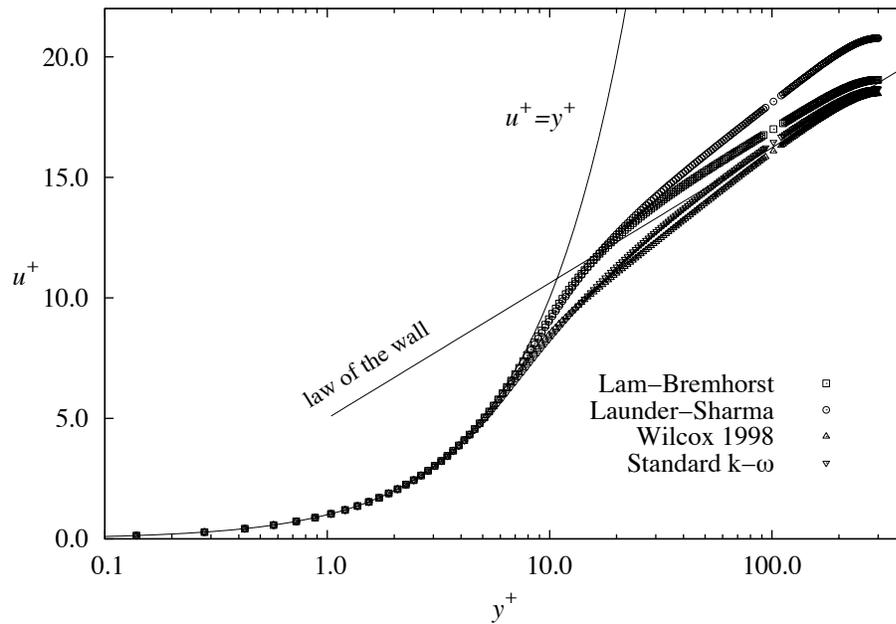


Fig. 4.3 Grid-converged results of four models for the case of fully developed pipe flow.

IV. Two-Dimensional Finite-Volume RANS Solver

The two-dimensional RANS solver is a more complex solver and was developed using the programming language C++. It will often be referred to here as an Incompressible Computational Enstrophy Structured Solver (ICESS). The code is capable of solving two-dimensional axisymmetric flows. The following sections give a brief overview of the methods employed in the two-dimensional code as well as initial results validating the code. The details of this code are included in Appendices F through L.

A. Coordinate System

Complex geometries are often modeled using a body-fitted coordinate system known as curvilinear coordinates. In two dimensions, this coordinate system is created by assuming that a computational domain (ξ, η) can be defined by a transformation of the physical domain (x, y) where $\xi = \xi(x, y)$ and $\eta = \eta(x, y)$. This type of transformation allows the governing equations to be discretized and solved in the computational domain. This method has been widely used for CFD. See for example the work of Rhie [74] or Shyy, Udaykumar, Rao, and Smith [75]. In this work, the grids are confined to orthogonal, rectilinear grids. Therefore,

$$\begin{aligned} x &= x(\xi) \\ y &= y(\eta) \end{aligned} \quad (4.1)$$

and $\xi_{,y} = \eta_{,x} = 0$. Partial derivatives of any continuously differentiable scalar, ϕ , in the physical domain can be written as partial derivatives in the computational domain according to

$$\begin{aligned} \frac{\partial}{\partial x} &= J y_{,\eta} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial y} &= J x_{,\xi} \frac{\partial}{\partial \eta} \end{aligned} \quad (4.2)$$

where J is the Jacobian scalar of the coordinate transformation and is defined as

$$J \equiv \frac{\partial(\xi, \eta)}{\partial(x, y)} = \begin{vmatrix} \xi_{,x} & 0 \\ 0 & \eta_{,y} \end{vmatrix} = \xi_{,x} \eta_{,y} \quad (4.3)$$

A scalar equation can be transformed from the physical domain to the computational domain by applying the properties of the transformation above. In this way, the two-dimensional steady-state general scalar transport equation can be written in curvilinear coordinates as

$$\frac{\partial(\bar{V}_\xi\phi)}{\partial\xi} + \frac{\partial(\bar{V}_\eta\phi)}{\partial\eta} = y_{,\eta}^2 \frac{\partial}{\partial\xi} \left(\Gamma J \frac{\partial\phi}{\partial\xi} \right) + x_{,\xi}^2 \frac{\partial}{\partial\eta} \left(\Gamma J \frac{\partial\phi}{\partial\eta} \right) + \frac{S}{J} \quad (4.4)$$

where Γ is the diffusion coefficient, S represents the source terms and

$$\begin{aligned} \bar{V}_\xi &\equiv y_{,\eta} \bar{V}_x \\ \bar{V}_\eta &\equiv x_{,\xi} \bar{V}_y \end{aligned} \quad (4.5)$$

are the contravariant velocity components in the curvilinear coordinate plane. More details on the coordinate transformation in Cartesian and cylindrical coordinates can be found in Appendices F and G.

B. Transport Equation Discretization

The transport equations for all scalar transport properties are solved using the finite-volume method. This method is applied by integrating Eq. (4.4) about a control volume referred to here as a cell. The resulting integral is then discretized using second-order discretization schemes. A collocated grid arrangement is used and all flow properties are defined at the cell centers. Therefore, for any given cell, P, the flow properties of neighboring cells are defined at the E, W, N, and S (east, west, north, and south respectively) cell centers. The convection terms are discretized to allow deferred correction between first-order upwinding and second-order upwinding. All other terms in the equations are discretized in the computational domain using second-order differencing methods. For work by others using similar solution methods, see Rhie [74], Versteeg and Malalasekera [76], Ferziger and Peric [77]. Details on the general discretization can be found in Appendix H. Details on the finite-volume discretization for the transport equations can be found in Appendix I for Cartesian coordinates and Appendix J for cylindrical coordinates.

C. Pressure Coupling

To complete the solution process, the continuity equation must be coupled with the momentum equations. This is done by implementing a well-known algorithm known as the Semi-Implicit Method for

Pressure-Linked Equations (SIMPLE) algorithm. The method was first suggested by Patankar and Spalding [78], and can be found in many CFD books. See for example Versteeg and Malalasekera [79]. The method solves for the pressure distribution using a combination of the continuity equation and momentum equations in a guess-and-correct manner. The method was originally suggested for staggered-grid arrangements and can induce extreme pressure oscillations in collocated grid arrangements. Rhie and Chow [80] suggest a correction to the way velocities are interpolated to cell faces in order to smooth out pressure fluctuations. This correction method is implemented in the current code. Details on this method in Cartesian coordinates is included in Appendix K. Details in cylindrical coordinates are included in Appendix L.

Other methods for coupling the continuity equation to the transport equations exist. For example, the fluid vorticity can also be evaluated and a vorticity transport equation can form the closing link between the continuity equation and the transport equations. A short overview of such a method is included in Appendix M which could prove useful in two dimensions. However, vorticity transport methods are difficult at best to implement in three dimensions, and seem to be waning in popularity in two dimensions.

D. Solution Procedure

The transport equations and SIMPLE algorithm are solved in an iterative manner. Inner loop iterations on each of the transport equations are performed within an outer loop. The outer loop sweeps through each of the transport equations in the following order: x -momentum, y -momentum, pressure (SIMPLE), turbulent kinetic energy, second differential equation (ε , ω , ζ , etc.). The pressure correction equation must be solved to a higher degree of accuracy than the transport equations during each outer iteration. Therefore, the default in the code employs 100 inner iterations on the pressure equation to every 10 inner iterations on each of the transport equations. Before repeating the outer iteration, the turbulent eddy viscosity is updated based on the new turbulence parameters. Iterations of the outer loop are repeated until the residuals of each equation stop changing. The number of outer and inner iterations as well as the values of relaxation factors can be specified by the user to speed convergence.

E. Code Validation

1. Laminar Lid-Driven Cavity

A case often used for testing the initial phases of a CFD code is the lid-driven cavity case. Figure 4.4 shows the results from ICESS for the x -velocity along the vertical centerline of the lid-driven cavity at a Reynolds number of 100. The Richardson extrapolation presented by Richardson [81] and Richardson and Gaunt [82] has been used to estimate the results for a fully grid-converged solution from the solutions of coarse, medium, and fine grids. The Richardson extrapolation method followed in this work is the method presented in the *Journal of Fluids Engineering* “Statement on the Control of Numerical Accuracy” (<http://journaltool.asme.org/Content/JFENumAccuracy.pdf>). In this case, uniform grids of sizes 2500, 10000, and 40000 cells were used for the coarse, medium, and fine grids respectively.

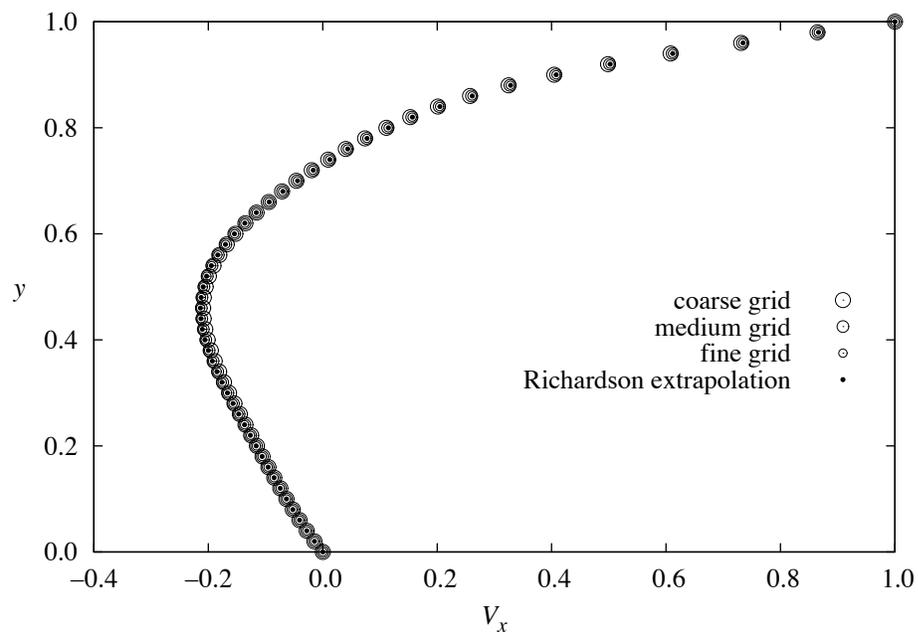


Fig. 4.4 ICESS grid refinement results of the x -velocity profile along the vertical centerline of the lid-driven cavity at a Reynolds number of 100.

Grid refinement studies for the lid-driven cavity case at a Reynolds number of 100 were also conducted in the Fluent and OpenFOAM software packages using the same grids as those used for the grid-refinement study in the author’s code. Figures 4.5 and 4.6 show the grid refinement results from Fluent and

OpenFOAM respectively for the x -velocity profile along the vertical centerline of the lid-driven cavity case. Notice that at $y = 0.64$, the Richardson extrapolation in Figure 4.5 seems inconsistent with the solutions from the coarse, medium, and fine grids. This is likely caused by the low precision at which the data from Fluent was reported. Although Fluent calculated the solution using double precision numbers, the results were reported to less than single precision accuracy. The Richardson extrapolation is very sensitive to differences between solutions. When the differences between solutions are small and the solutions are reported only to low precision, the differences calculated in the Richardson algorithm are very similar between the coarse and medium solutions and the medium and fine solutions. This similarity in differences causes the Richardson extrapolation to estimate that the grid is still far from converged, and the final grid-resolved solution to lie significantly outside of the reported results. This result is perhaps even more apparent when the Richardson extrapolation results of each of the software packages are compared.

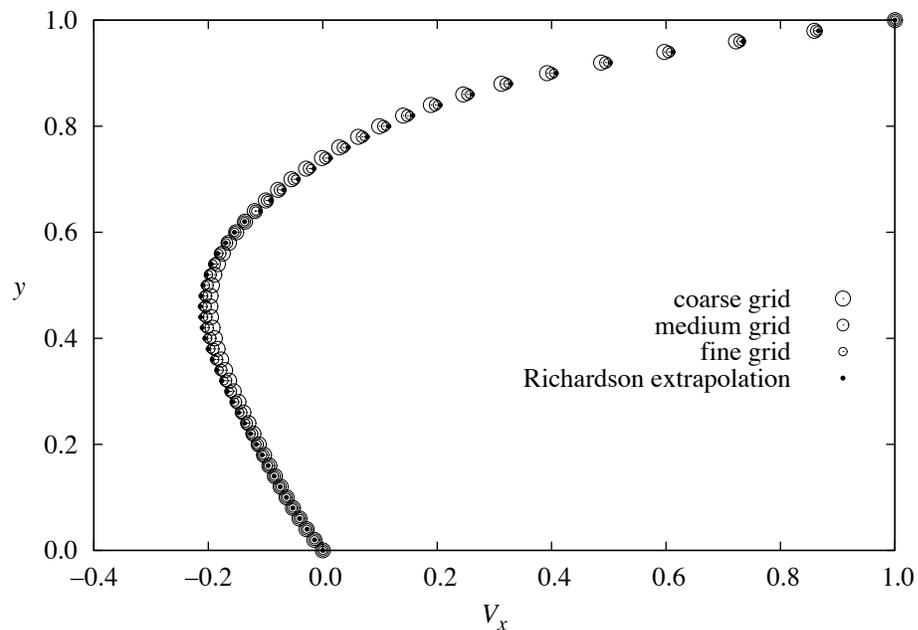


Fig. 4.5 Fluent grid refinement results of the x -velocity profile along the vertical centerline of the lid-driven cavity at a Reynolds number of 100.

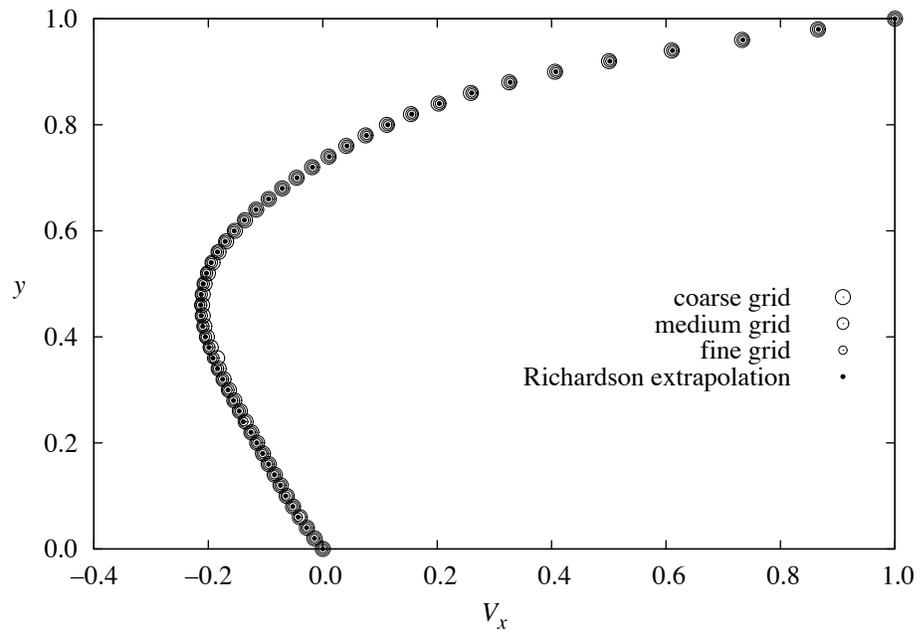


Fig. 4.6 OpenFOAM grid refinement results of the x -velocity profile along the vertical centerline of the lid-driven cavity at a Reynolds number of 100.

Figure 4.7 shows the extrapolated results from each of the software packages. The results suggest that the laminar portion of the author's code is working correctly.

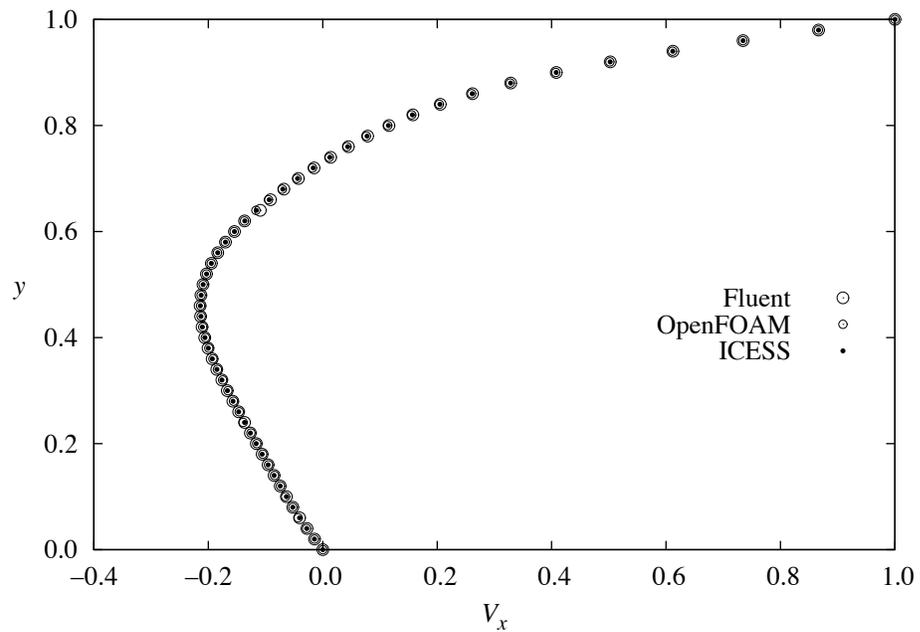


Fig. 4.7 Richardson extrapolation results for the x -velocity profile along the vertical centerline of the lid-driven cavity. Comparison of results from Fluent, OpenFOAM, and ICESS.

2. Order of Convergence

In order to ensure that the discretization schemes used in this work are second order, an order of convergence test was conducted. This is done by examining the error of the solution as the cell size approaches zero. In order to calculate the error of a solution, the exact solution of the case must be known.

The exact solution for laminar, fully developed channel flow is

$$\bar{V}_x = \frac{1}{\mu} \frac{d\hat{p}}{dx} \left(\frac{y^2}{2} - Ly \right) \quad (4.6)$$

where L is the channel half-width and y is measured outward normal to the wall. The fully developed channel flow case was run in ICESS where $\mu = .001$, $\rho = .1$, $L = 0.5$, and $d\hat{p}/dx = -7.2$. Variable grid sizes ranging from 25 to 800 cells in the y -direction were run with cells clustered near the wall using logarithmic clustering. Once the solution was fully converged, the RMS error of the solution was calculated. Figure 4.8 shows the RMS error with respect to the average cell size squared, Δy^2 . A line passing through the data points obtained from the two finest grids is superimposed on the plot. The fact that the error is nearly linear with Δy^2 shows that the solution method is nearly second order.

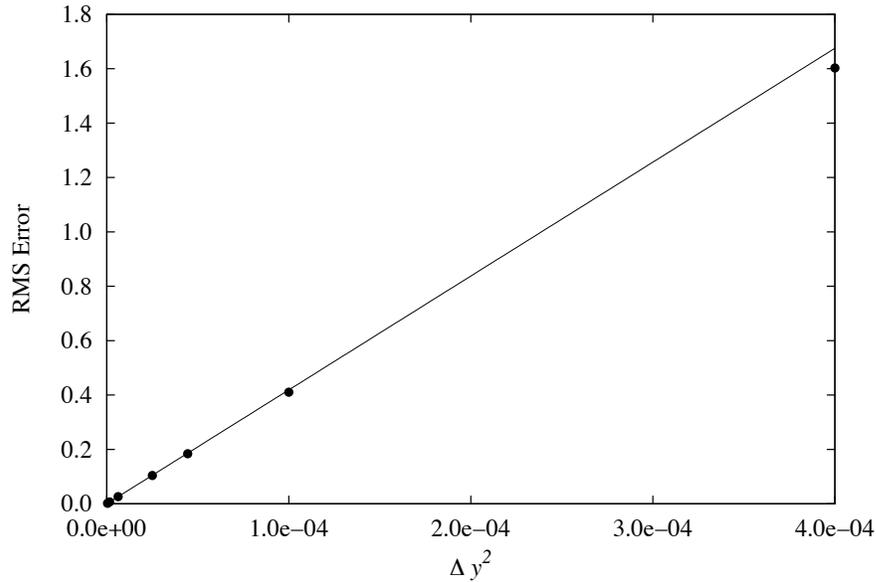


Fig. 4.8 Finite volume algorithm RMS error as a function of average cell size squared.

3. Turbulent Fully Developed Channel Flow

Turbulent fully developed flow in a channel is a common case for testing solution algorithms. The traditional $k-\omega$ model and the Wilcox [45] $k-\omega$ model were coded into ICESS. Figures 4.9 and 4.10 show the results of a grid resolution study of the predicted mean velocity from the Wilcox 1998 model using the one-dimensional and two-dimensional finite-volume algorithms where $\mu = .001$, $\rho = .1$, $L = 0.5$, and $d\hat{p}/dx = -7.2$. It was found that using 200 cells in the y -direction with logarithmic clustering near the wall produced grid-resolved results for each of the models.

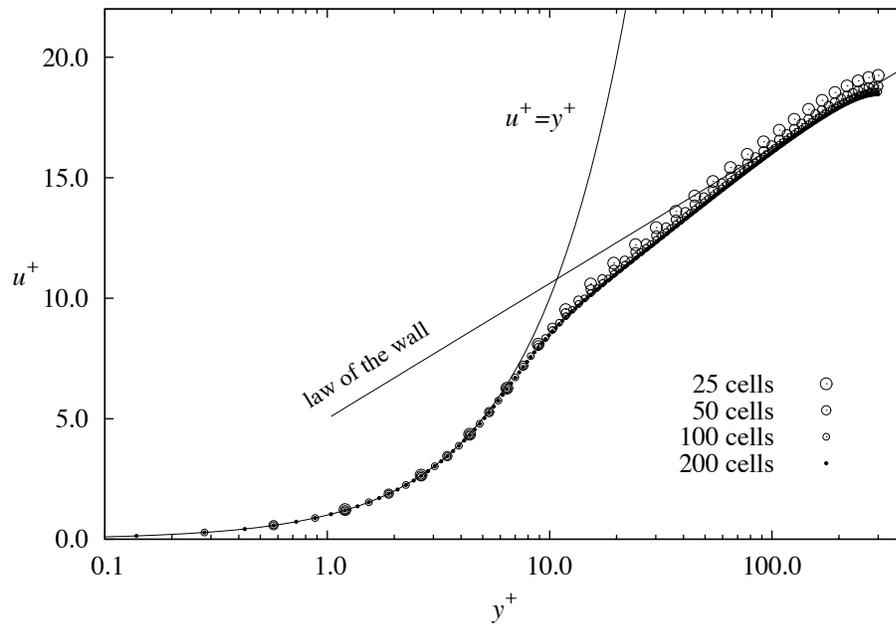


Fig. 4.9 Grid resolution for the mean velocity predicted by the Wilcox 1998 $k-\omega$ model using the one-dimensional finite-volume method.

The results of the one-dimensional and two-dimensional codes are compared here to results from a one-dimensional finite-difference code. Figures 4.11 and 4.12 show the predicted mean velocity from the traditional $k-\omega$ and Wilcox 1998 models using the one- and two-dimensional finite-volume codes compared with the finite-difference results. The results for these figures were obtained using 200 cells with logarithmic clustering near the wall in each of the algorithms.

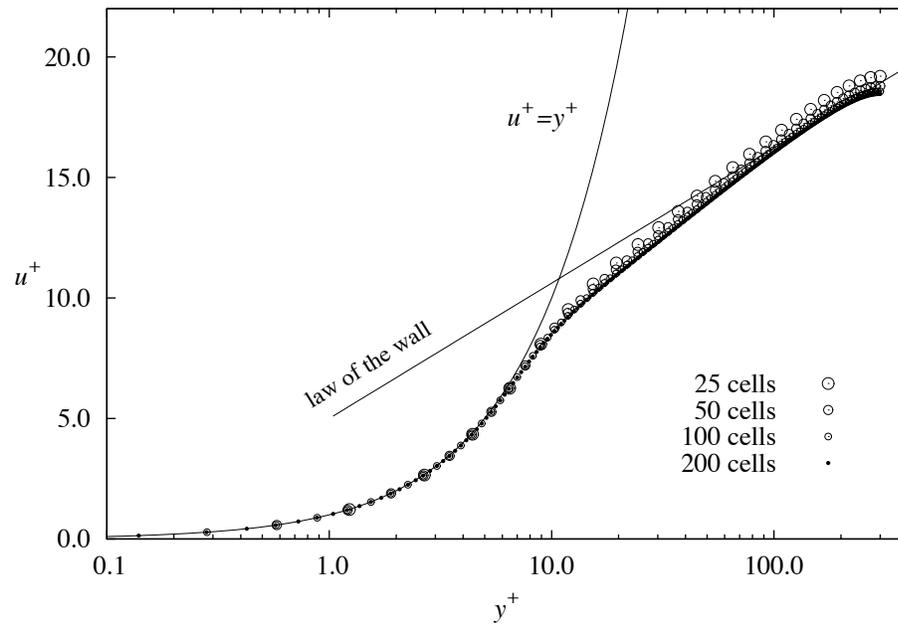


Fig. 4.10 Grid resolution for the mean velocity predicted by the Wilcox 1998 $k-\omega$ model using the two-dimensional finite-volume method.

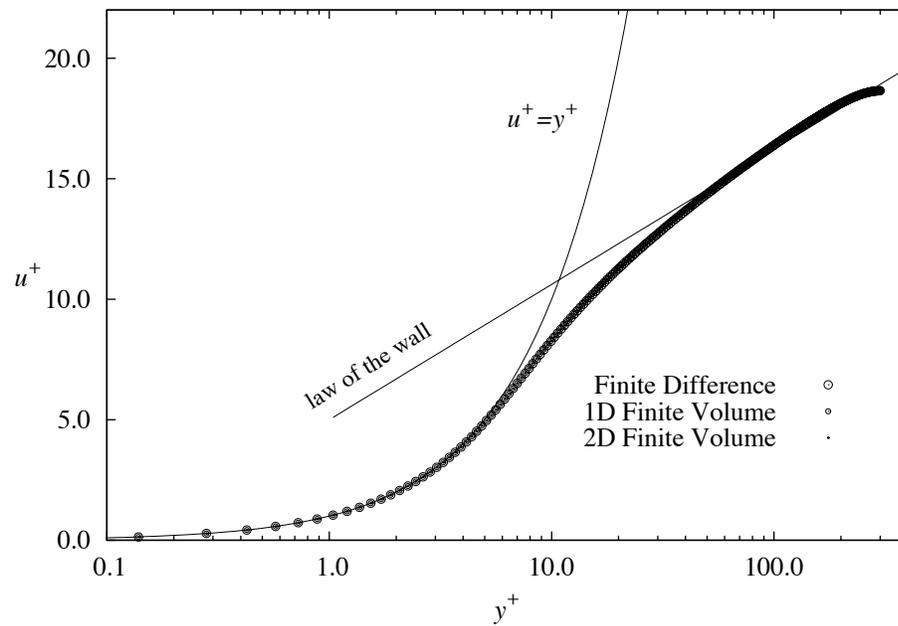


Fig. 4.11 Comparison of results from the one- and two-dimensional finite-volume methods and a finite-difference method for the mean velocity predicted by the traditional $k-\omega$ model.

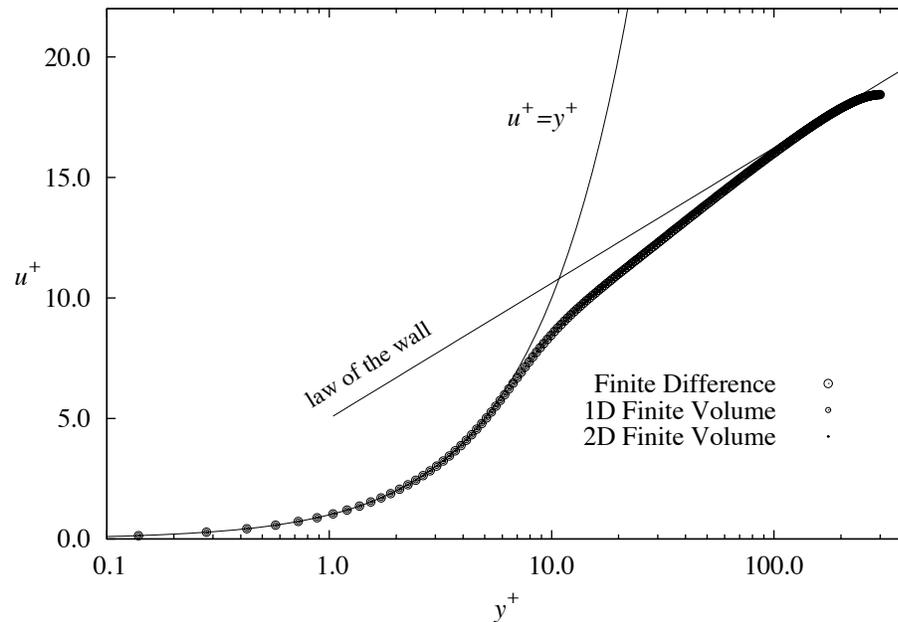


Fig. 4.12 Comparison of results from the one- and two-dimensional finite-volume methods and a finite-difference method for the mean velocity predicted by the Wilcox 1998 $k-\omega$ model.

The results presented in this section show that the two-dimensional finite-volume code, ICESS, written for this research is second-order accurate and able to correctly implement current two-equation turbulence models.

V. One-Dimensional Runge-Kutta Direct Integration

One-dimensional flows can also be solved using a direct numerical integration method. In general, the second-order transport equations can be written as a set of first-order equations and directly integrated using a high-order integration method. For example, in one dimension, the Boussinesq-RANS equation simplify to a first-order differential equation and two-equation turbulence models can be written as a system of two second-order equations. Each second-order equation can be rearranged to yield two first-order equations. In this way, a two-equation turbulence model can generally be written for a one-dimensional flow scenario as a series of five first-order equations. These can be directly integrated to yield the turbulence model solution. Details on how this is can be done for traditional $k-\varepsilon$ and $k-\omega$ models is included in Appendix N.

VI. Summary and Conclusions

Turbulence models are generally solved using computational methods known as CFD methods. In order to evaluate boundary conditions and turbulence models, a computational framework is necessary. Although the work of this dissertation could have been accomplished using previously written CFD algorithms, the author chose to write the CFD algorithms for this work from the ground up. This provided the author with the freedom to alter the code in any way desired and allowed the greatest flexibility for implementing and testing the model properties as needed. Several codes were written for this work including finite-difference, finite-volume, and direct integration codes. An overview of these codes has been included here, and details for the algorithms are included in the appendices.

CHAPTER 5

SMOOTH-WALL BOUNDARY CONDITIONS FOR DISSIPATION-BASED MODELS

I. Introduction

Many of the most-widely used turbulence models are based on the Boussinesq hypothesis and can be termed two-equation dissipation-based models. These models include the k - ε , k - ω , and k - ζ models and their development was discussed in Chapter 1. The correct smooth-wall boundary conditions for these models are commonly misunderstood and are therefore often implemented incorrectly. This chapter discusses the misconception and explains the correct implementation of smooth-wall boundary conditions giving examples from the k - ε and k - ω models. A meeting paper has been published on the subject [83].

In general, a turbulence model is a set of mathematical relations that are developed to model the physical phenomenon known as turbulence. Once these equations have been constructed, the laws of mathematics must be followed during the solution process to ensure a correct solution has been obtained. Most dissipation-based turbulence models are called two-equation models because they model two turbulent properties using second-order differential transport equations. The branch of mathematics known as differential equations has long been understood, and specific rules exist for the application of boundary conditions to such equations. A misconception in these boundary conditions is the fundamental flaw in the way many dissipation-based two-equation turbulence models are implemented today.

Durbin [19] was the first to point this out when he stated, “*These conditions must violate the energy balance...*” Although this statement was made about 20 years ago, it has apparently not been well-understood. The fundamental flaw in traditional smooth-wall boundary conditions can be seen by considering the physical characteristics of turbulence near a smooth wall.

II. Physics of Smooth-Wall Boundary Conditions

Most dissipation-based turbulence models are based on two second-order differential transport equations, one for the turbulent kinetic energy, and one for a second turbulence property. The turbulent kinetic energy per unit mass, k , is a measure of the kinetic energy of the fluid resulting from turbulent fluctuations in the flow and is defined as one-half the mean square magnitude of the velocity fluctuations

$$k \equiv \frac{1}{2} \overline{\tilde{V}^2} = \frac{1}{2} \left(\overline{\tilde{V}_x^2} + \overline{\tilde{V}_y^2} + \overline{\tilde{V}_z^2} \right) \quad (5.1)$$

The turbulent-kinetic-energy transport equation has traditionally been developed by taking the trace of the specific Reynolds-stress-transport equation and can be written for incompressible flow as

$$\frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)k = 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \varepsilon + \nabla \cdot [(v + \nu_t/\sigma_k)\nabla k] \quad (5.2)$$

where ε is the dissipation of turbulent kinetic energy. The dissipation of turbulent kinetic energy is defined as

$$\varepsilon \equiv \overline{\mathbf{J}(\tilde{\mathbf{V}}) \cdot \mathbf{J}(\tilde{\mathbf{V}})} \quad (5.3)$$

Most two-equation models apply a second-order differential equation to model the dissipation as a transport property and relate these two properties to the turbulent eddy viscosity algebraically. Examples can be seen in Chapter 1.

At a perfectly smooth, no-slip wall, the turbulent fluctuations must be exactly zero just as the mean velocity is exactly zero. This provides the widely acknowledged and implemented boundary condition

$$k = 0, \quad y = 0 \quad (5.4)$$

where y is the distance from the wall measured normal to the wall. Traditionally, a boundary condition is also imposed on the second transport equation by specifying the dissipation at a smooth surface. However, we see from Eq. (5.3) that the dissipation is a function only of the derivatives of the velocity fluctuations. Because physics imposes no constraints on these derivatives at a smooth wall, it is incorrect to apply a boundary condition on the dissipation at a smooth wall. We will see in subsequent sections that the laws of mathematics require an additional boundary condition to be applied in order to ensure a unique solution to

the turbulence model. The additional boundary condition is obtained by taking the gradient of the turbulent-kinetic-energy at a smooth wall. By definition,

$$\nabla k \equiv \nabla \left(\frac{1}{2} \overline{\tilde{V}^2} \right) = \overline{\tilde{V} \nabla \tilde{V}} \quad (5.5)$$

Because the velocity fluctuations at a smooth wall must be zero at a smooth wall ($\tilde{V} = 0$), we see that Eq. (5.5) requires the gradient of k to also be zero.

$$\nabla k = 0, \quad y = 0 \quad (5.6)$$

This is the correct second boundary condition at a smooth wall, and any boundary condition other than this, as Durbin states, “*must violate the energy balance...*” In a subsequent publication, Durbin [67] also makes the statement, “*These two conditions on k suffice to determine the solution for the coupled system of equations; there is no need to impose conditions of ε at the wall – indeed, it would be incorrect to do so.*” Hence, we see that a no-slip wall imposes two boundary conditions on k and none on ε . There is no need to impose a wall boundary condition on ε , and it is incorrect to do so. The value of ε at a smooth wall is that required to satisfy both Eqs. (5.4) and (5.6), as was originally pointed out by Durbin roughly 20 years ago. A better understanding of the mathematics of the problem can be seen by considering examples of the k - ε and k - ω models for the simple case of fully developed flow in a channel.

III. Smooth-Wall Boundary Conditions for the k - ε Model

The case of fully developed channel flow can be constructed in Cartesian coordinates where x is the ordinate in the direction of flow, and y is the ordinate normal to the channel wall as outlined in Appendix C. For this case, there are no gradients in the mean flow transport properties in the x -direction. This formulation is sometimes called the parallel-flow approximation because it assumes that the gradients of the flow properties with respect to x are negligible compared to the gradients with respect to y . This is nearly true very close to a wall even for flow that is not in a fully developed state. The formulation including the continuity, RANS, turbulent-kinetic-energy, and dissipation-transport equations can be written in nondimensional form as

$$\begin{aligned}
\frac{du^+}{dy^+} &= \frac{1+p^+y^+}{1+\nu^+}, \quad \frac{dp^+}{dy^+} = 0 \\
\frac{d}{dy^+} \left[(1+\nu^+/\sigma_k) \frac{dk^+}{dy^+} \right] &= \varepsilon^+ + \varepsilon_o^+ - \nu^+ \left(\frac{du^+}{dy^+} \right)^2 \\
\frac{d}{dy^+} \left[(1+\nu^+/\sigma_\varepsilon) \frac{d\varepsilon^+}{dy^+} \right] &= C_{\varepsilon 2} f_2 \frac{\varepsilon^{+2}}{k^+} - C_{\varepsilon 1} f_1 \nu^+ \frac{\varepsilon^+}{k^+} \left(\frac{du^+}{dy^+} \right)^2 - E^+ \\
\nu^+ &= C_\mu f_\mu k^{+2} / \varepsilon^+
\end{aligned} \tag{5.7}$$

where σ_k , σ_ε , C_μ , $C_{\varepsilon 1}$, and $C_{\varepsilon 2}$ are model-dependent constants, and f_μ , f_1 , f_2 , E^+ , and ε_o^+ are the model-dependent wall damping functions. For a detailed development of these equations, see Appendices B and C.

In order to close this sixth-order formulation, the damping functions and six boundary conditions must be specified. Three obvious conditions come from the channel centerline. At the centerline of the channel, the flow property gradients must be zero. These conditions can be written as

$$\frac{du^+}{dy^+}(R_\tau) = \frac{dk^+}{dy^+}(R_\tau) = \frac{d\varepsilon^+}{dy^+}(R_\tau) = 0 \tag{5.8}$$

Applying the first of these boundary conditions to Eq. (5.7) gives

$$p^+ = -1/R_\tau, \quad \frac{du^+}{dy^+} = \frac{1-y^+/R_\tau}{1+\nu^+} \tag{5.9}$$

The traditional no-slip boundary conditions are

$$u^+(0) = k^+(0) = 0 \tag{5.10}$$

The final boundary condition has been a topic of some confusion and has typically been chosen as a relationship for the dissipation at the wall. Various researchers have taken different approaches to this final condition, and therefore, the condition has traditionally differed with the model. However, this final condition should be model independent, and be a property of the physical aspects of the flow.

A. The Launder-Sharma k - ε Model

The Launder-Sharma [69] model is a special case of Eq. (5.7) where the wall-damping functions are

$$\begin{aligned}
f_\mu &= \exp[-3.4/(1+R_t/50)^2], \quad f_1 = 1.0, \quad f_2 = 1 - 0.3 \exp(-R_t^2), \\
R_t &= \frac{k^{+2}}{\varepsilon^+}, \quad \varepsilon_o^+ = \frac{1}{2k^+} \left(\frac{\partial k^+}{\partial y^+} \right)^2, \quad E^+ = 2\nu^+ \left(\frac{\partial^2 u^+}{\partial y^{+2}} \right)^2, \\
C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
\end{aligned} \tag{5.11}$$

Using Eqs. (5.9) and (5.11) in Eq. (5.7) gives

$$\begin{aligned}
p^+ &= -1/R_t, \quad \frac{du^+}{dy^+} = \frac{1-y^+/R_t}{1+\nu^+} \\
\frac{d}{dy^+} \left[(1+\nu^+/\sigma_k) \frac{dk^+}{dy^+} \right] &= \varepsilon^+ + \frac{1}{2k^+} \left(\frac{\partial k^+}{\partial y^+} \right)^2 - \nu^+ \left(\frac{du^+}{dy^+} \right)^2 \\
\frac{d}{dy^+} \left[(1+\nu^+/\sigma_\varepsilon) \frac{d\varepsilon^+}{dy^+} \right] &= C_{\varepsilon 2} f_2 \frac{\varepsilon^{+2}}{k^+} - C_{\varepsilon 1} C_\mu f_\mu k^+ \left(\frac{du^+}{dy^+} \right)^2 - 2\nu^+ \left(\frac{\partial^2 u^+}{\partial y^{+2}} \right)^2 \\
\nu^+ &= C_\mu f_\mu k^{+2} / \varepsilon^+, \quad R_t \equiv \frac{k^{+2}}{\varepsilon^+} \\
f_\mu &= \exp[-3.4/(1+R_t/50)^2], \quad f_2 = 1 - 0.3 \exp(-R_t^2), \\
C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
\end{aligned} \tag{5.12}$$

The near-wall behavior of this model can be observed by considering the Taylor-Series expansions

$$\begin{aligned}
k^+(y^+) &= k^+(0) + k^{+'}(0)y^+ + \frac{k^{+''}(0)}{2}y^{+2} + \dots \\
\varepsilon^+(y^+) &= \varepsilon^+(0) + \varepsilon^{+'}(0)y^+ + \frac{\varepsilon^{+''}(0)}{2}y^{+2} + \dots
\end{aligned} \tag{5.13}$$

Launder and Sharma suggest a final boundary condition at the wall of

$$\varepsilon^+(0) = 0 \tag{5.14}$$

Applying this boundary condition along with Eq. (5.10) to Eq. (5.13) gives the near-wall approximations

$$\begin{aligned}
k^+(y^+) &= k^{+'}(0)y^+ + \frac{k^{+''}(0)}{2}y^{+2} + \dots \\
\varepsilon^+(y^+) &= \varepsilon^{+'}(0)y^+ + \frac{\varepsilon^{+''}(0)}{2}y^{+2} + \dots \\
R_t \equiv \frac{k^{+2}}{\varepsilon^+} &= \frac{k^{+'2}(0)}{\varepsilon^{+'}(0)}y^+ + \left[\frac{k^{+'}(0)k^{+''}(0)}{\varepsilon^{+'}(0)} - \frac{k^{+'2}(0)\varepsilon^{+''}(0)}{2\varepsilon^{+'2}(0)} \right]y^{+2} + \dots \\
\nu^+ \equiv C_\mu f_\mu k^{+2}/\varepsilon^+ &= C_\mu \exp(-3.4) \frac{k^{+'2}(0)}{\varepsilon^{+'}(0)}y^+ + \dots \\
f_\mu &= \exp(-3.4) + \dots, \quad f_2 = 0.7 + \dots
\end{aligned} \tag{5.15}$$

Using these near-wall expansions in the k -transport equation in Eq. (5.12) gives

$$\frac{d^2 k^+}{dy^{+2}} = \frac{k^{+'}(0)}{y^+} + \dots \tag{5.16}$$

which is obviously indeterminate at the wall. On the other hand, if we apply the physically correct boundary condition $k^{+'}(0) = 0$ and treat $\varepsilon^+(0)$ as an unknown, the near-wall expansion for the k -transport equation becomes

$$\frac{d^2 k^+}{dy^{+2}} = \varepsilon^+(0) + k^{+''}(0) + \dots \tag{5.17}$$

which naturally requires $\varepsilon^+(0) = 0$. Thus we see that referring to $\varepsilon^+(0) = 0$ as a boundary condition is incorrect. Enforcing $\varepsilon^+(0) = 0$ does not require the true boundary condition $k^{+'}(0) = 0$ to hold, while forcing $k^{+'}(0) = 0$ requires $\varepsilon^+(0) = 0$ by virtue of the differential equations. Therefore, using $\varepsilon^+(0) = 0$ as a near-wall approximation for ε^+ is appropriate if and only if $k^{+'}(0) = 0$ is also enforced. With the full no-slip boundary condition enforced, the near-wall expansions for the Launder-Sharma damping function and eddy viscosity are

$$\begin{aligned}
k^+(y^+) &= \frac{k^{+''}(0)}{2} y^{+2} + \frac{k^{+'''}(0)}{6} y^{+3} + \dots \\
\varepsilon^+(y^+) &= \varepsilon^+(0) + \varepsilon^{+'}(0) y^+ + \frac{\varepsilon^{+''}(0)}{2} y^{+2} + \dots \\
R_t \equiv \frac{k^{+2}}{\varepsilon^+} &= \frac{k^{+''2}(0)}{4\varepsilon^{+'}(0)} y^{+3} + \left[\frac{k^{+''}(0)k^{+'''}(0)}{6\varepsilon^{+'}(0)} - \frac{k^{+''2}(0)\varepsilon^{+''}(0)}{8\varepsilon^{+'2}(0)} \right] y^{+4} + \dots \\
v^+ &\equiv C_\mu f_\mu k^{+2} / \varepsilon^+ = C_\mu \exp(-3.4) \frac{k^{+''2}(0)}{2\varepsilon^{+'}(0)} y^{+3} + \dots \\
f_\mu &= \exp(-3.4) + \dots, \quad f_2 = 0.7 + \dots
\end{aligned} \tag{5.18}$$

Thus we see the complete model given in Eq. (5.12) requires the correct no-slip wall boundary conditions $u^+(0) = 0$, $k^+(0) = 0$, and $k^{+'}(0) = 0$ along with the centerline boundary conditions $k^{+'}(R_\tau) = 0$ and $\varepsilon^{+'}(R_\tau) = 0$ to yield a solution that is mathematically determinate.

B. The Lam-Bremhorst k - ε Model

The Lam-Bremhorst [68] model is a special case of Eq. (5.7) where the wall-damping functions are

$$\begin{aligned}
f_\mu &= [1 - \exp(-0.0165R_y)]^2 (1 + 20.5/R_t), \quad f_1 = 1 + (0.05/f_\mu)^3, \quad f_2 = 1 - \exp(-R_t^2), \\
R_t &= \frac{k^{+2}}{\varepsilon^+}, \quad R_y = y^+ \sqrt{k^+}, \quad \varepsilon_o^+ = 0, \quad E^+ = 0, \\
C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
\end{aligned} \tag{5.19}$$

Using Eqs. (5.9) and (5.19) in Eq. (5.7) gives the model and boundary conditions

$$\begin{aligned}
p^+ &= -1/R_\tau, \quad \frac{du^+}{dy^+} = \frac{1 - y^+/R_\tau}{1 + v^+} \\
\frac{d}{dy^+} \left[(1 + v^+/\sigma_k) \frac{dk^+}{dy^+} \right] &= \varepsilon^+ - v^+ \left(\frac{du^+}{dy^+} \right)^2 \\
\frac{d}{dy^+} \left[(1 + v^+/\sigma_\varepsilon) \frac{d\varepsilon^+}{dy^+} \right] &= C_{\varepsilon 2} f_2 \frac{\varepsilon^{+2}}{k^+} - C_{\varepsilon 1} f_1 v^+ \frac{\varepsilon^+}{k^+} \left(\frac{du^+}{dy^+} \right)^2 \\
v^+ &= C_\mu f_\mu R_t, \quad R_t = \frac{k^{+2}}{\varepsilon^+}, \quad R_y = y^+ \sqrt{k^+} \\
f_\mu &= [1 - \exp(-0.0165R_y)]^2 (1 + 20.5/R_t), \quad f_1 = 1 + (0.05/f_\mu)^3, \quad f_2 = 1 - \exp(-R_t^2), \\
C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3 \\
u^+(0) &= 0, \quad k^+(0) = 0, \quad k^{+'}(R_\tau) = 0, \quad \varepsilon^{+'}(R_\tau) = 0
\end{aligned} \tag{5.20}$$

As the final boundary condition in addition to Eqs. (5.8) and (5.10), Lam and Bremhorst suggest the relation

$$\varepsilon^+(0) = k^{+''}(0) \quad (5.21)$$

Although this boundary conditions is commonly accepted in the literature as being the appropriate smooth-wall boundary condition for this model, it is in fact mathematically incorrect. There are an infinite number of solutions to Eq. (5.20) that also satisfy Eq. (5.21). This relation can be derived directly from the differential equations and therefore cannot be termed a boundary condition. This relation will be satisfied independent of the boundary condition and thus yields an infinite number of solutions to the system of equations.

To see why Eq. (5.21) is not a viable boundary condition for Eq. (5.20), consider Eq. (5.20) in the limit as y^+ approaches zero. In this limit, both R_t and R_y go to zero and the wall damping functions and eddy viscosity near a smooth wall reduce to

$$\begin{aligned} y^+ \rightarrow 0, \quad f_\mu &= \frac{20.5(0.0165)^2 \varepsilon^+ y^{+2}}{k^+}, \quad \nu^+ = 20.5(0.0165)^2 C_\mu k^+ y^{+2}, \\ f_1 &= 1 + \frac{(0.05)^3 k^{+3}}{(20.5)^3 (0.0165)^6 \varepsilon^{+3} y^{+6}}, \quad f_2 = \frac{k^{+4}}{\varepsilon^{+2}} \end{aligned} \quad (5.22)$$

Using these limiting relations in the differential equation from Eq. (5.20) produces the near-wall system of equations, which applies in the limit as y^+ approaches zero,

$$y^+ \rightarrow 0, \quad \frac{du^+}{dy^+} = 1, \quad \frac{d^2 k^+}{dy^{+2}} = \varepsilon^+, \quad \frac{d^2 \varepsilon^+}{dy^{+2}} = \frac{-(0.05)^3 C_\mu C_{\varepsilon 1} k^{+3}}{(20.5)^2 (0.0165)^4 \varepsilon^{+2} y^{+4}} \quad (5.23)$$

Thus we see that Eq. (5.21) is satisfied independent of the final boundary condition. Therefore, Eq. (5.21) does not provide the additional information required to obtain a unique solution to the indeterminate system of equations.

As a further demonstration of why Eq. (5.21) is not a viable boundary condition for completing the indeterminate system in Eq. (5.20), consider the similar system and boundary conditions

$$\begin{aligned} \frac{d\hat{u}}{d\hat{y}} &= 1 - \hat{y}, & \frac{d^2\hat{k}}{d\hat{y}^2} &= \hat{\varepsilon} - \hat{y}^4 \left(\frac{d\hat{u}}{d\hat{y}} \right)^2, & \frac{d^2\hat{\varepsilon}}{d\hat{y}^2} &= \hat{y}^6 - \hat{y}^2 \left(\frac{d\hat{u}}{d\hat{y}} \right)^2 \\ \hat{u}(0) &= \hat{k}(0) = \hat{k}'(1) = \hat{\varepsilon}'(1) = 0 \end{aligned} \quad (5.24)$$

Equation (5.24) is simple enough to obtain a closed-form solution. The general solution is

$$\begin{aligned} \hat{u} &= C_1 + \hat{y} - \frac{\hat{y}^2}{2}, & \hat{k} &= C_2 + C_3\hat{y} + \frac{C_4\hat{y}^2}{2} + \frac{C_5\hat{y}^3}{6} - \frac{13\hat{y}^6}{360} + \frac{21\hat{y}^7}{420} - \frac{31\hat{y}^8}{1680} + \frac{\hat{y}^{10}}{5040} \\ \hat{\varepsilon} &= C_4 + C_5\hat{y} - \frac{\hat{y}^4}{12} + \frac{\hat{y}^5}{10} - \frac{\hat{y}^6}{30} + \frac{\hat{y}^8}{56} \end{aligned} \quad (5.25)$$

The boundary conditions in Eq. (5.24) can be used to eliminate four of the five arbitrary constants

$$\begin{aligned} \hat{u} &= \hat{y} - \frac{\hat{y}^2}{2}, & \hat{k} &= C_4 \frac{\hat{y}^2 - 2\hat{y}}{2} + \frac{338\hat{y} - 92\hat{y}^3 - 182\hat{y}^6 + 252\hat{y}^7 - 93\hat{y}^8 + \hat{y}^{10}}{5040} \\ \hat{\varepsilon} &= C_4 + \frac{-92\hat{y}^2 - 70\hat{y}^4 + 84\hat{y}^5 - 28\hat{y}^6 + 15\hat{y}^8}{840} \end{aligned} \quad (5.26)$$

As should be expected, there are an infinite number of solutions to any indeterminate fifth-order system of differential equations with only four boundary conditions. However, if the mathematical logic presented by Lam and Bremhorst is correct, then we should be able to reduce Eq. (5.26) to a single unique solution by simply applying a boundary condition obtained from the second differential equation in Eq. (5.24) evaluated at $\hat{y} = 0$. If we accept this logic, then our final boundary condition for Eq. (5.24) is

$$\hat{\varepsilon}(0) = \hat{k}''(0) \quad (5.27)$$

However, applying this “boundary condition” yields the result $C_4 = C_4$. Clearly, applying this boundary condition yields a system of equations with an infinite number of solutions. It is a simple law of mathematics that a boundary condition cannot be derived directly from the differential equations themselves. It is apparent here that such a process can result in a system of equations with an infinite number of solutions.

Patel, Rodi, and Scheuerer [84] suggest a “more convenient boundary condition” for the Lam-Bremhorst model

$$\frac{d\varepsilon^+}{dy^+}(0) = 0 \quad (5.28)$$

However, this boundary condition is also incorrect because physics dictates no boundary condition on the dissipation of turbulent kinetic energy as outlined at the beginning of this chapter. The boundary condition for Eq. (5.26) that is analogous to the physically based condition given in Eq. (5.6) is $\hat{k}'(0) = 0$. Applying this to Eq. (5.26) gives the unique solution

$$\hat{u} = \hat{y} - \frac{\hat{y}^2}{2}, \quad \hat{k} = \frac{169\hat{y}^2 - 92\hat{y}^3 - 182\hat{y}^6 + 252\hat{y}^7 - 93\hat{y}^8 + \hat{y}^{10}}{5040} \quad (5.29)$$

$$\hat{\varepsilon} = \frac{169 - 276\hat{y} - 210\hat{y}^4 + 252\hat{y}^5 - 84\hat{y}^6 + 45\hat{y}^8}{2520}$$

and results in a nonzero result for both $\hat{\varepsilon}$ and $\hat{\varepsilon}'$. The boundary condition analogous to Eq. (5.28) is $\hat{\varepsilon}'(0) = 0$. Applying this boundary condition to Eq. (5.26) gives $-23/210 = 0$ which may cause some concern with regard to using Eq. (5.28) as a boundary condition for Eq. (5.20).

Examination of the incomplete fifth-order system given by Eq. (5.24) has revealed that using Eq. (5.27) as the fifth boundary condition results in an infinite number of solutions. On the other hand, Eq. (5.24) has no solution if Eq. (5.28) is used as the fifth boundary condition. It can be shown that the incomplete fifth-order system in Eq. (5.20) exhibits very similar behavior. However, solutions to Eq. (5.20) must be obtained numerically.

Because fully developed flow is one dimensional, a solution to Eq. (5.20) combined with Eq. (5.6) can be obtained by direct numerical integration. This permits the use of efficient high-order numerical methods such as the fourth-order Runge-Kutta algorithm. Because such solutions can be obtained quickly on very fine grids, fully developed channel flow provides an excellent benchmark for testing more computationally intensive CFD algorithms.

To facilitate direct numerical integration, the two second-order equations in Eq. (5.20) can be converted to four first-order equations by using the change of variables

$$q^+ \equiv -(1 + \nu^+ / \sigma_k) \frac{dk^+}{dy^+} \quad (5.30)$$

$$\theta^+ \equiv -(1 + \nu^+ / \sigma_\varepsilon) \frac{d\varepsilon^+}{dy^+}$$

Combining Eq. (5.20) with Eq. (5.6), applying the change of variables given in Eq. (5.30), and eliminating ν^+ by direct substitution provides the complete one-dimensional fifth-order formulation

$$\begin{aligned}
\frac{du^+}{dy^+} &= \frac{\varepsilon^+(1-y^+/R_\tau)}{\varepsilon^+ + C_\mu f_\mu k^{+2}} \equiv u^{+'} \\
\frac{dk^+}{dy^+} &\equiv -\frac{\sigma_k q^+ \varepsilon^+}{\sigma_k \varepsilon^+ + C_\mu f_\mu k^{+2}} \\
\frac{dq^+}{dy^+} &= \frac{C_\mu f_\mu k^{+2} \varepsilon^+ (1-y^+/R_\tau)^2}{(\varepsilon^+ + C_\mu f_\mu k^{+2})^2} - \varepsilon^+ \\
\frac{d\varepsilon^+}{dy^+} &\equiv -\frac{\sigma_\varepsilon \theta^+ \varepsilon^+}{\sigma_\varepsilon \varepsilon^+ + C_\mu f_\mu k^{+2}} \\
\frac{d\theta^+}{dy^+} &= C_{\varepsilon 1} f_1 C_\mu f_\mu k^{+u^{+'2}} - C_{\varepsilon 2} f_2 \frac{\varepsilon^{+2}}{k^+} \\
R_t &\equiv k^{+2}/\varepsilon^+, \quad R_y \equiv k^{+1/2} y^+, \\
f_{\mu 1} &= [1 - \exp(-0.0165 R_y)]^2, \quad f_{\mu 2} = 1 + 20.5/R_t, \\
f_\mu &= f_{\mu 1} f_{\mu 2}, \quad f_1 = 1 + (0.05/f_\mu)^3, \quad f_2 = 1 - \exp(-R_t^2), \\
u^+(0) &= 0, \quad k^+(0) = 0, \quad q^+(0) = 0, \\
q^+(R_\tau) &= 0, \quad \theta^+(R_\tau) = 0
\end{aligned} \tag{5.31}$$

It should be noted that the new variable q^+ is a dimensionless form of the total diffusive flux of turbulent kinetic energy k , which includes both molecular and turbulent diffusion. Similarly, θ^+ is a dimensionless diffusive flux for ε . This brings to light another physical interpretation of the boundary condition given in Eq. (5.6), which led directly to the equivalent boundary condition in Eq. (5.31), i.e., $q^+(0) = 0$. With this interpretation, Eq. (5.6) can be viewed as a mathematical statement of the simple fact that turbulent kinetic energy cannot be diffused through a solid wall. The formulation for fully developed flow that is given by Eq. (5.31) requires that q^+ vanish at the wall and at the centerline. Thus, all of the turbulent kinetic energy that is generated within this steady flow must also be dissipated within the flow. If a boundary condition obtained from either Eq. (5.21) or Eq. (5.28) is used in place of that obtained from Eq. (5.6), this energy balance is not enforced. This is the origin of Durbin's statement that, "*These conditions must violate the energy balance,*" [19].

A numerical solution to the five first-order differential equations given in Eq. (5.31) can be obtained using fourth-order Runge-Kutta integration combined with an appropriate numerical root-finding method.

Because only three of the five boundary conditions are given at $y^+ = 0$, the solution for $\varepsilon^+(0)$ and $\theta^+(0)$ must be obtained from the differential equations. The process is started with initial estimates for $\varepsilon^+(0)$ and $\theta^+(0)$. From these initial estimates, fourth-order Runge-Kutta integration can be used to obtain $q^+(R_\tau)$ and $\theta^+(R_\tau)$. The initial estimates are then refined using an appropriate numerical method until the solution is found, which corresponds to the correct centerline values $q^+(R_\tau) = 0$ and $\theta^+(R_\tau) = 0$.

A few words of caution may be in order here. Some of the terms in Eq. (5.31) are numerically indeterminate if $k^+ = 0$ and/or $\varepsilon^+ = 0$. Notice that a division by zero occurs in the definition of R_t for $\varepsilon^+ = 0$. Thus, depending on the compiler, conditional relations may be required to enforce $f_{\mu 2} = 1$ and $f_2 = 1$ for $\varepsilon^+ \rightarrow 0$. For most compilers, Eq. (5.31) is numerically indeterminate for $k^+ = 0$. In this limit, both R_t and R_y go to zero and the eddy viscosity and wall damping functions reduce to

$$\begin{aligned} k^+ \rightarrow 0, \quad \nu^+ &= 20.5(0.0165)^2 C_\mu k^+ y^{+2}, \quad f_2 \frac{\varepsilon^{+2}}{k^+} = k^{+3}, \\ f_1 f_\mu k^+ &= 20.5(0.0165)^2 \varepsilon^+ y^{+2} + \frac{(0.05)^3 \varepsilon^+(0) y^{+2}}{8(20.5)^2 (0.0165)^4} \end{aligned} \quad (5.32)$$

Hence, for the limit $k^+ \rightarrow 0$, the formulation given in Eq. (5.31) should be conditionally replaced with its near-wall asymptote

$$\begin{aligned} k^+ \rightarrow 0, \quad \frac{du^+}{dy^+} &= \frac{1 - y^+/R_\varepsilon}{1 + \nu^+} \equiv u^{+'}, \\ \frac{dk^+}{dy^+} &\equiv -\frac{\sigma_k q^+}{\sigma_k + \nu^+}, \\ \frac{dq^+}{dy^+} &= \nu^+ u^{+'2} - \varepsilon^+ \\ \frac{d\varepsilon^+}{dy^+} &\equiv -\frac{\sigma_\varepsilon \theta^+}{\sigma_\varepsilon + \nu^+}, \\ \frac{d\theta^+}{dy^+} &= C_{\varepsilon 1} C_\mu F_\varepsilon u^{+'2} - C_{\varepsilon 2} k^{+3} \\ \nu^+ &= 20.5(0.0165)^2 C_\mu k^+ y^{+2}, \\ F_\varepsilon &\equiv 20.5(0.0165)^2 \varepsilon^+ y^{+2} + \frac{(0.05)^3 \varepsilon^+(0) y^{+2}}{8(20.5)^2 (0.0165)^4} \end{aligned} \quad (5.33)$$

To demonstrate that Eq. (5.21) is not a valid boundary condition for completing the formulation given in Eq. (5.20), the results shown in Fig. 5.1 were obtained from Eq. (5.31) using randomly selected wall

boundary conditions. The no-slip wall boundary conditions were used for both u^+ and k^+ , but the wall boundary conditions for q^+ , ε^+ , and θ^+ , as well as the wall-scaled dimensionless half width R_τ were generated as listed in Fig. 5.1 using the “rand()” function, which generates a random number between 0.0 and 1.0. From the results presented in Fig. 5.1, it can be concluded that Eq. (5.21) is enforced directly by Eq. (5.20), completely independent of the boundary conditions.

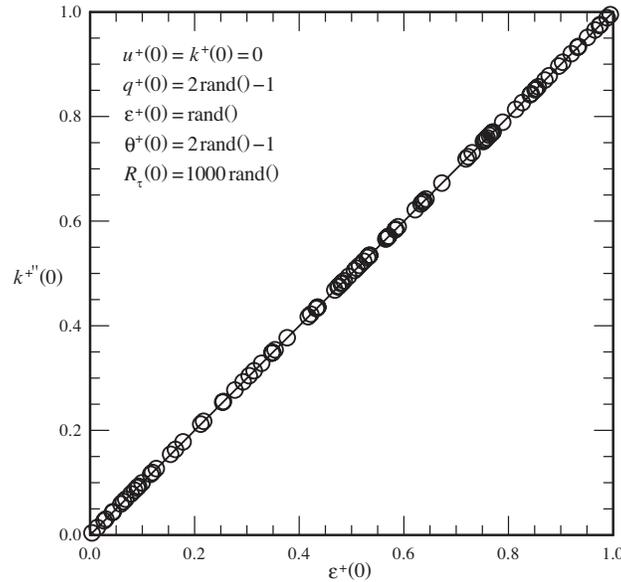


Fig. 5.1 Solutions to Eq. (5.31) with randomly selected wall boundary conditions.

To demonstrate that Eq. (5.28) is not a valid boundary condition for completing the formulation given in Eq. (5.20), the results shown in Fig. 5.2 were obtained from Eq. (5.31) using the no-slip wall boundary conditions for u^+ , k^+ , and q^+ , with the wall boundary conditions for θ^+ obtained from Eq. (5.28). For several values of R_τ , the computed value for q^+ at the centerline is plotted as a function of the remaining wall boundary condition $\varepsilon^+(0)$. Valid solutions to Eq. (5.20) could only correspond to those points where these curves intersect the axis $q^+(R_\tau) = 0$. From the results presented in Fig. 5.2, it can be seen that there is only one solution to Eq. (5.20) that satisfies Eq. (5.28) and the no-slip wall boundary conditions. That is the trivial laminar solution

$$u^+ = y^+ - y^{+2}/(2R_\tau), \quad k^+ = q^+ = \varepsilon^+ = \theta^+ = 0 \quad (5.34)$$

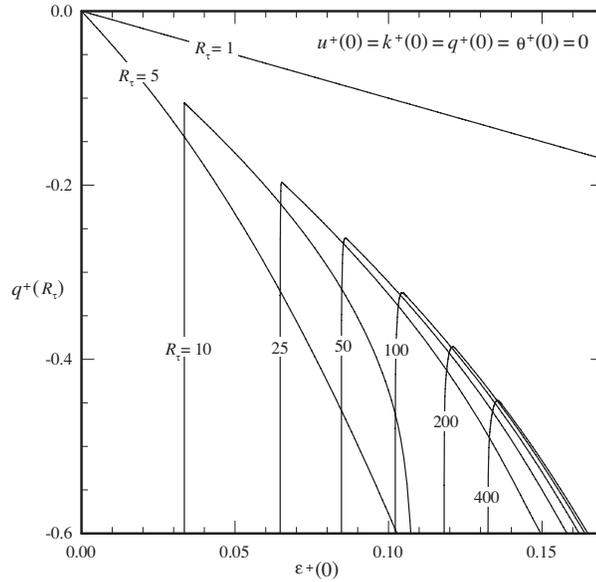


Fig. 5.2 Solutions to Eq. (5.31) with no slip and no dissipation gradient at the wall.

There is no turbulent flow solution to Eq. (5.20) that satisfies Eq. (5.28) and the no-slip wall boundary conditions.

Examination of the numerical results shown in Figs. 5.1 and 5.2 reveals that Eq. (5.20) exhibits behavior very similar to that demonstrated analytically for the hypothetical fifth-order system given by Eq. (5.24). Using $\varepsilon^+(0) = k^{+''}(0)$ as the fifth boundary condition for Eq. (5.20) results in an infinite number of solutions. On the other hand, Eq. (5.20) has no valid turbulent flow solution if $\varepsilon^{+'}(0) = 0$ is used as the fifth boundary condition. This underscores the critical importance of always using the correct no-slip boundary conditions $u^+(0) = k^+(0) = k^{+'}(0) = 0$.

IV. Numerical Results from CFD Algorithms

Because the zero-gradient boundary condition for k in Eq. (5.6) is not explicitly enforced in many commonly implemented k - ε turbulence models, solutions obtained from these models are not unique. To demonstrate this fact, the RANS formulations for fully developed channel flow for the Launder-Sharma and Lam-Bremhorst models were solved numerically using a second-order finite difference algorithm with successive underrelaxation. Solutions were obtained on the domain extending from the wall to the channel centerline, and grid points were clustered near the wall using logarithmic clustering. To ensure that all

results were fully converged, the successive underrelaxation was allowed to continue until the observed changes were reduced to within the double-precision machine accuracy.

To ensure that all results were grid resolved, the grids were uniformly refined until no significant changes were observed with additional grid refinement. For a given axial pressure gradient, the Launder-Sharma model required a somewhat finer grid than was required for the Lam-Bremhorst model. Results of an example grid-resolution study for the Launder-Sharma model are shown in Figs. 5.3–5.5. All results shown in these figures were obtained using the fixed axial pressure gradient, which yields a value of y^+ at the centerline equal to 300. For the grid refinements shown in Figs. 5.3–5.5, the four grids produced channel Reynolds numbers (based on the channel width and mean velocity) that were equal to 10,009, 10,653, 10,832, and 10,878, respectively. An additional refinement of the grid to 401 nodes, which is not shown in Fig. 5.3, produced a channel Reynolds number of 10,889. From these and other similar results, it was concluded that for Reynolds numbers on the order of 10,000, the 201-node grid used for Figs. 5.3–5.5 produced adequate grid resolution with both the Lam-Bremhorst and Launder-Sharma turbulence models.

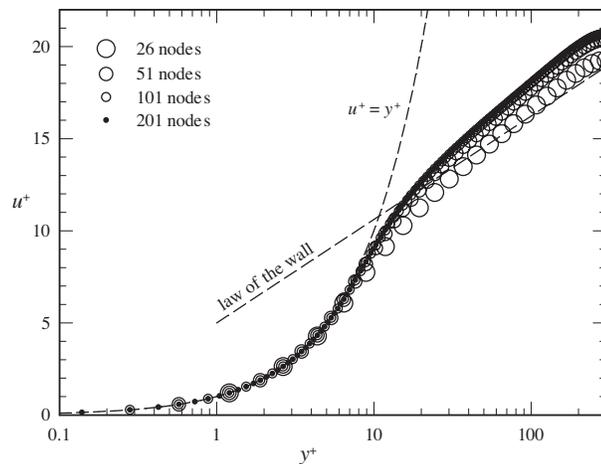


Fig. 5.3 Grid resolution for the mean velocity predicted from the Launder-Sharma k - ε model.

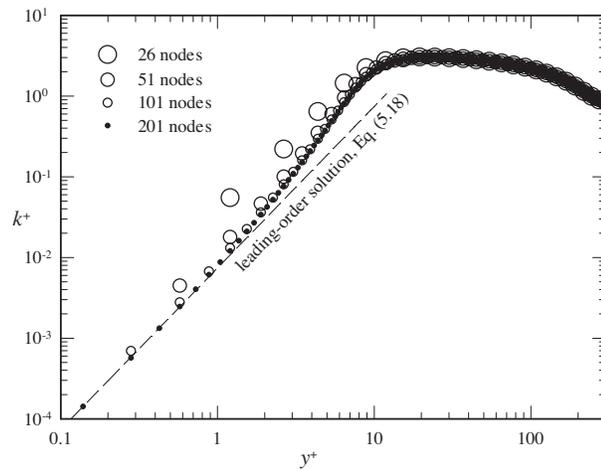


Fig. 5.4 Grid resolution for the turbulent energy predicted from the Launder-Sharma k - ε model.

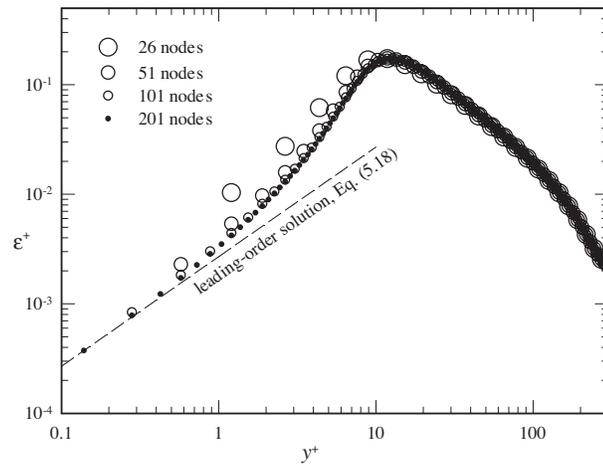


Fig. 5.5 Grid resolution for the turbulent dissipation predicted from the Launder-Sharma k - ε model.

To demonstrate that solutions obtained from commonly implemented k - ε turbulence models are not unique, the second-order successive underrelaxation algorithm was implemented using a slight variant of the wall boundary conditions specified in Eq. (5.6), which allows the user to specify an arbitrary value for the gradient of k at the wall. Figures 5.6 and 5.7 show converged and grid-resolved results obtained from this algorithm using three different gradient boundary conditions for k at the wall: $k^+(0) = 0.0$, $k^+(0) = 0.1$, and $k^+(0) = 1.0$.

To demonstrate that traditional implementations of these turbulence models do not necessarily converge to the solution that yields $k^+(0) = 0$, Figs. 5.6 and 5.7 also include results obtained from the same

algorithm, turbulence models, and grid, but with a traditional implementation of the wall boundary conditions, which uses only $u^+(0) = 0$ and $k^+(0) = 0$ together with an asymptotic relation obtained from the differential equations, i.e., $\varepsilon^+(0) = k^{+''}(0)$ for the Lam-Bremhorst model and $\varepsilon^+(0) = 0$ for the Launder-Sharma model. For additional comparison, Figs. 5.6 and 5.7 also show results from the general-purpose finite-volume CFD solver FLUENT [85], which were obtained using the same turbulence models and grid with only the traditional wall boundary conditions implemented.

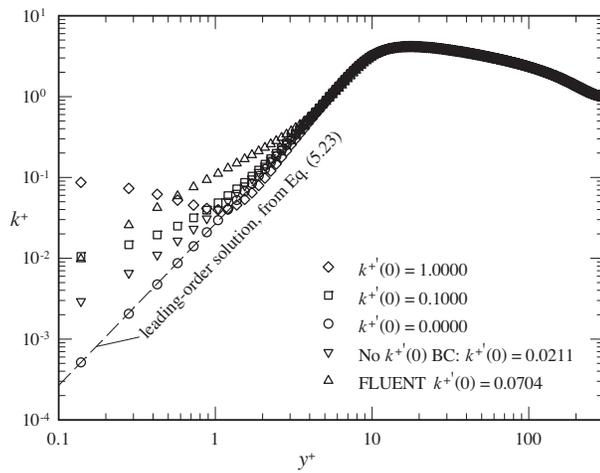


Fig. 5.6 Effects of wall boundary conditions on turbulent energy predicted from the Lam-Bremhorst model.

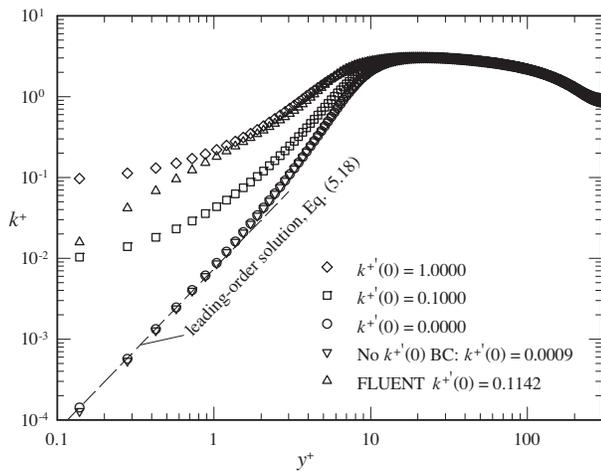


Fig. 5.7 Effects of wall boundary conditions on turbulent energy predicted from the Launder-Sharma model.

The results shown in Figs. 5.6 and 5.7 clearly demonstrate that when the natural boundary condition $k^+(0) = 0$ is not enforced, solutions obtained from commonly implemented k - ε turbulence models are not unique. When the boundary condition $k^+(0) = 0$ is omitted, solutions obtained from the resulting indeterminate formulation are implementation dependent. Notice from Fig. 5.6 that for the Lam-Bremhorst model with traditional implementation of the wall boundary conditions, the finite-difference algorithm converged to a different solution from that obtained using the finite-volume algorithm with the same indeterminate boundary conditions. Neither of these solutions agrees with that obtained from the finite-difference algorithm with the boundary condition $k^+(0) = 0$ enforced. Similarly, we see from Fig. 5.7 that these finite-difference and finite-volume implementations of the Launder-Sharma model converge to different solutions with traditional implementations of the wall boundary conditions. However, for the particular implementation used to obtain the results shown in Fig. 5.7, the indeterminate finite-difference algorithm converged to a solution that is very close to that obtained when the complete set of smooth-wall boundary conditions was enforced. This should not be viewed as an endorsement for implementing the Launder-Sharma turbulence model with mathematically incomplete boundary conditions.

From the near-wall expansions of the Launder-Sharma model given in Eqs. (5.16) and (5.17), it was shown that enforcing $k^+(0) = 0$ requires $\varepsilon^+(0) = 0$, whereas enforcing $\varepsilon^+(0) = 0$ does not require $k^+(0) = 0$. This can also be demonstrated numerically by examining the near-wall behavior of ε obtained from numerical solutions using different gradient boundary conditions for k at the wall. Such results are shown in Fig. 5.8, which were obtained from converged and grid-resolved solutions for the same channel flow that was used to obtain the results shown in Fig. 5.7. Notice that, although $k^+(0)$ does affect the near-wall behavior of ε^+ , all of these solutions satisfy $\varepsilon^+(0) = 0$. Only the solution corresponding to $k^+(0) = 0$ also satisfies the physically correct no-slip condition at the smooth wall.

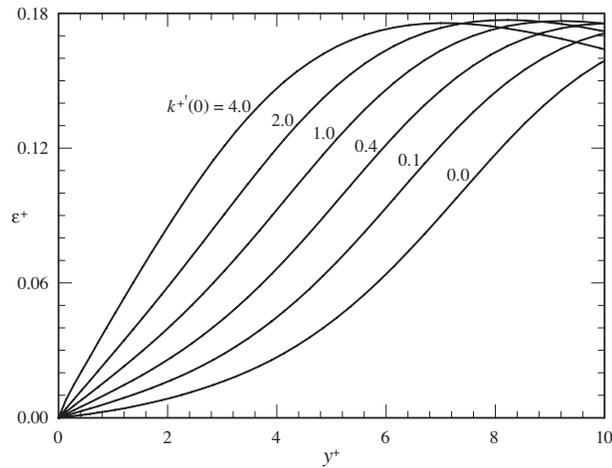


Fig. 5.8 Effects of wall boundary conditions on near-wall dissipation for Launder-Sharma model.

Because the wall damping functions for the Lam-Bremhorst model decay rapidly with increasing y^+ , the wall boundary condition $k^+(0)=0$ has little impact on the velocity profiles predicted from this turbulence model. On the other hand, the wall damping functions for the Launder-Sharma model decay slower and have a more significant effect on the predicted mean velocity farther from the wall. This can be seen in Fig. 5.9, which displays the velocity profiles for the same solutions that were used to obtain the turbulent energy profiles displayed in Fig. 5.7. It may be worth reiterating that all of the solutions shown in Fig. 5.9 satisfy the traditional wall boundary conditions $u^+(0)=0$ and $k^+(0)=0$ together with the asymptotic relation obtained from the differential equations, $\varepsilon^+(0)=0$.

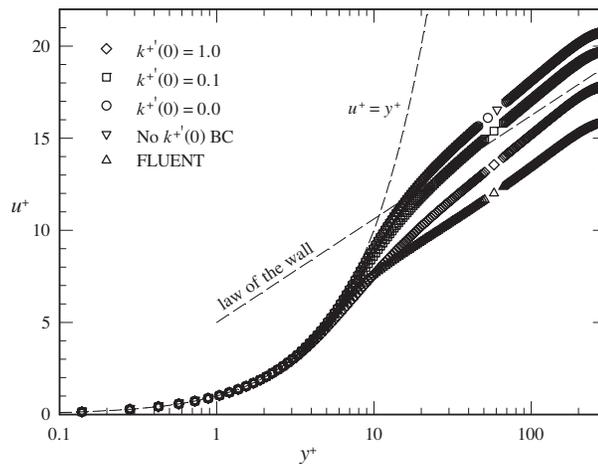


Fig. 5.9 Effects of wall boundary conditions on the mean velocity predicted from the Launder-Sharma model.

Anyone who has taken time to compare results obtained from different CFD algorithms and different k - ε turbulence models will likely have noticed that there is often a greater difference between the results obtained from two different implementations of the same turbulence model than there is between the results obtained from the same implementation of two different turbulence models. The results shown in Fig. 5.9 may shed some light on the reason for this phenomenon. We should not be too surprised to learn that results obtained from commonly used k - ε turbulence models are implementation dependent, if we recognize that these models are all short one boundary condition, and thus are mathematically indeterminate.

Because the CFD community has not traditionally implemented two wall boundary conditions on k and none on ε , implementation of the correct smooth-wall boundary conditions first proposed by Durbin [19] has been less than enthusiastically embraced. The actual implementation of these boundary conditions is dependent on the numerical method being used to solve the system of differential equations. However, this implementation should not be difficult using well-known methods in either finite-difference or finite-volume algorithms. For example, results presented in this section were obtained from a finite-difference algorithm. To implement the no-slip wall boundary conditions, the k -transport equation at any wall node was replaced with the boundary condition $k(0) = 0$. At the first node off the wall, the k -transport equation was replaced with a second-order finite-difference approximation for the boundary condition $k^+(0) = 0$. Because there is no wall boundary condition on ε , the ε -transport equation at any wall node was replaced with the asymptotic relation obtained from the differential equations, i.e., $\varepsilon^+(0) = k^{+''}(0)$ for the Lam-Bremhorst model and $\varepsilon^+(0) = 0$ for the Launder-Sharma model. The implementation of the ε -transport equation is identical to that of the traditional formulation. The error in the traditional formulation is not that the transport equation for ε is incorrectly implemented. Rather, the error is in assuming that the near-wall asymptote obtained from the differential equations can be used to replace the final boundary condition required at the wall.

As scientists and engineers, we do not have the luxury of choosing boundary conditions for ease of numerical implementation. Boundary conditions are dictated by physics. It is our obligation to understand

and implement them correctly if we hope to achieve mathematical formulations that correctly model physics. The fundamental mathematical error of deriving a so called boundary condition directly from the differential equations is not unique to the classical k - ε turbulence models that have been considered here. It is also an important concern for many other turbulence models developed more recently.

V. Smooth-Wall Boundary Conditions for the k - ω Model

The k - ω model formulation for fully developed channel flow is presented in Appendix B. The formulation including the continuity, RANS, turbulent-kinetic-energy, and dissipation-transport equations can be written in nondimensional form as

$$\begin{aligned} \frac{du^+}{dy^+} &= \frac{1+p^+y^+}{1+\nu^+}, & \frac{dp^+}{dy^+} &= 0 \\ \frac{d}{dy^+} \left[(1+\nu^+/\sigma_k) \frac{dk^+}{dy^+} \right] &= C_\mu f_k k^+ \omega^+ - \nu^+ \left(\frac{du^+}{dy^+} \right)^2 \\ \frac{\partial}{\partial y^+} \left[(1+\nu^+/\sigma_\omega) \frac{\partial \omega^+}{\partial y} \right] &= C_{\omega 2} f_2 \omega^{+2} - C_{\omega 1} f_1 \nu^+ \frac{\omega^+}{k^+} \left(\frac{du^+}{dy^+} \right)^2 \\ \nu^+ &= f_\mu \frac{k^+}{\omega^+} \end{aligned} \tag{5.35}$$

where σ_k , σ_ω , C_μ , $C_{\omega 1}$, and $C_{\omega 2}$ are model-dependent constants, and f_μ , f_1 , f_2 , and f_k are the model-dependent wall damping functions. Just as with the k - ε model, in order to close this sixth-order formulation, the damping functions and six boundary conditions must be specified. Again, three conditions come from the channel centerline

$$\frac{du^+}{dy^+}(R_\tau) = \frac{dk^+}{dy^+}(R_\tau) = \frac{d\omega^+}{dy^+}(R_\tau) = 0 \tag{5.36}$$

Applying the first of these conditions to Eq. (5.30) gives

$$p^+ = -1/R_\tau, \quad \frac{du^+}{dy^+} = \frac{1-y^+/R_\tau}{1+\nu^+} \tag{5.37}$$

To close the formulation, three additional boundary conditions are needed. Two of these are the no-slip boundary conditions traditionally used for dissipation-based models

$$u^+(0) = k^+(0) = 0 \quad (5.38)$$

This leaves one boundary condition to be specified.

A version of the k - ω model that is commonly implemented is the Wilcox 1998 model [45]. The closure coefficients and wall-damping functions for this model are

$$\begin{aligned} f_\mu &= \frac{0.024 + R_t/6}{1 + R_t/6}, \quad f_k = \frac{4/15 + (R_t/8)^4}{1 + (R_t/8)^4} \begin{cases} 1, & \psi_k \leq 0 \\ \frac{1 + 680\psi_k^2}{1 + 400\psi_k^2}, & \psi_k > 0 \end{cases}, \\ f_1 &= \frac{1/9 + R_t/2.95}{f_\mu(1 + R_t/2.95)}, \quad f_2 = 1.0, \quad \psi_k = \frac{\nabla k \cdot \nabla \omega}{\omega^3}, \quad R_t = \frac{k}{\nu\omega} \\ C_\mu &= 0.09, \quad C_{\omega 1} = 0.52, \quad C_{\omega 2} = 0.072, \quad \sigma_k = 2.0, \quad \sigma_\omega = 2.0 \end{aligned} \quad (5.39)$$

Using Eqs. (5.37) and the nondimensional form of Eq. (5.39) in Eq. (5.35) gives the model for fully developed flow in a channel

$$\begin{aligned} p^+ &= -1/R_\tau, \quad \frac{du^+}{dy^+} = \frac{1 - y^+/R_\tau}{1 + \nu^+} \\ \frac{d}{dy^+} \left[(1 + \nu^+/\sigma_k) \frac{dk^+}{dy^+} \right] &= C_\mu f_k k^+ \omega^+ - \nu^+ \left(\frac{du^+}{dy^+} \right)^2 \\ \frac{\partial}{\partial y^+} \left[(1 + \nu^+/\sigma_\omega) \frac{\partial \omega^+}{\partial y^+} \right] &= C_{\omega 2} f_2 \omega^{+2} - C_{\omega 1} f_1 \nu^+ \frac{\omega^+}{k^+} \left(\frac{du^+}{dy^+} \right)^2 \\ \nu^+ &= f_\mu R_t \\ f_\mu &= \frac{0.024 + R_t/6}{1 + R_t/6}, \quad f_k = \frac{4/15 + (R_t/8)^4}{1 + (R_t/8)^4} \begin{cases} 1, & \psi_k \leq 0 \\ \frac{1 + 680\psi_k^2}{1 + 400\psi_k^2}, & \psi_k > 0 \end{cases}, \\ f_1 &= \frac{1/9 + R_t/2.95}{f_\mu(1 + R_t/2.95)}, \quad f_2 = 1.0, \quad \psi_k = \frac{1}{\omega^{+3}} \frac{dk^+}{dy^+} \frac{d\omega^+}{dy^+}, \quad R_t = \frac{k^+}{\omega^+} \\ C_\mu &= 0.09, \quad C_{\omega 1} = 0.52, \quad C_{\omega 2} = 0.072, \quad \sigma_k = 2.0, \quad \sigma_\omega = 2.0 \\ u^+(0) &= k^+(0) = k^{+'}(R_\tau) = \omega^{+'}(R_\tau) = 0 \end{aligned} \quad (5.40)$$

As in the case of Eq. (5.20), one additional boundary condition is needed to complete the fifth-order formulation expressed in Eq. (5.40). In the presentation of his 1998 k - ω model Wilcox [45] states, “*The final condition follows from examination of the differential equations for k and ω approaching the surface.*” For a smooth wall in the limit $y^+ \rightarrow 0$, the boundary condition $k^+(0) = 0$ requires $R_t(0) = 0$ and $\nu^+(0) = 0$. Thus, the differential equation for u^+ and the ω -transport equation given in Eq. (5.40) reduce to

$$y^+ \rightarrow 0, \quad \frac{du^+}{dy^+} = 1, \quad \frac{d^2\omega^+}{dy^{+2}} = C_{\omega 2}\omega^{+2} - \frac{C_{\omega 1}}{9} \quad (5.41)$$

Let the leading-order term in the solution for ω^+ be written as

$$\omega^+(y^+) = Ay^{+a} + \dots \quad (5.42)$$

where A and a are as yet unknown constants. Using Eq. (5.42) in the near-wall approximation for the ω -transport equation given by Eq. (5.41) yields

$$a(a-1)Ay^{+a-2} \cong C_{\omega 2}A^2y^{+2a} - C_{\omega 1}/9 \quad (5.43)$$

Equating the exponents and coefficients of y^+ in the leading-order terms, this near-wall relation requires

$$a = -2, \quad A = \frac{a(a-1)}{C_{\omega 2}} = \frac{6}{C_{\omega 2}} \quad (5.44)$$

Hence, after using Eq. (5.44) in Eq. (5.42), the leading-order solution for ω^+ yields

$$\omega^+(y^+) \underset{y^+ \rightarrow 0}{=} \frac{6}{C_{\omega 2}y^{+2}} \quad (5.45)$$

To minimize numerical truncation error associated with the singularity, Wilcox [86] suggests that Eq. (5.45) should be used in place of the ω -transport equation “for the first 7 to 10 grid points above the surface.” Wilcox also points out that the grid must be fine enough so that “these grid points must lie below $y^+ = 2.5 \dots$ ” In practice, Eq. (5.45) is often used as the final boundary condition by applying this relation only at the first grid point off the wall.

Because the leading-order solution given by Eq. (5.45) follows exclusively from the ω -transport equation with application of only the single boundary condition $k^+(0) = 0$, all solutions to Eq. (5.40) will exhibit this asymptotic behavior, completely independent of the fifth boundary condition that is required to obtain a unique solution to this system of equations. Equation (5.45) is certainly a valid asymptote for the ω -transport equation in Eq. (5.40) near a smooth wall. Thus, Eq. (5.45) can be used as an alternative to the ω -transport equation for y^+ approaching zero, provided that it is combined with five appropriate boundary conditions. However, Eq. (5.45) cannot be used as a substitute for one of the five required boundary conditions.

To further exhibit this point, consider the similar system and boundary conditions

$$\begin{aligned} \frac{d\hat{u}}{d\hat{y}} &= 1 - \hat{y}, & \frac{d^2\hat{k}}{d\hat{y}^2} &= \hat{y}^2\hat{\omega} - \hat{y}^4\left(\frac{d\hat{u}}{d\hat{y}}\right)^2, & \frac{d^2\hat{\omega}}{d\hat{y}^2} &= \frac{1}{\hat{y}^4} - \left(\frac{d\hat{u}}{d\hat{y}}\right)^2 \\ \hat{u}(0) &= \hat{k}(0) = \hat{k}'(1) = \hat{\omega}'(1) = 0 \end{aligned} \quad (5.46)$$

The general solution to this system of equations is

$$\begin{aligned} \hat{u} &= C_1 + \hat{y} - \frac{\hat{y}^2}{2}, & \hat{k} &= C_2 + C_3\hat{y} + \frac{C_4\hat{y}^4}{12} + \frac{C_5\hat{y}^5}{20} + \frac{840\hat{y}^2 - 504\hat{y}^6 + 560\hat{y}^7 - 195\hat{y}^8}{10080} \\ \hat{\omega} &= \frac{1}{6\hat{y}^2} + C_4 + C_5\hat{y} + \frac{-6\hat{y}^2 + 4\hat{y}^3 - \hat{y}^4}{12} \end{aligned} \quad (5.47)$$

The boundary conditions in Eq. (5.38) can be used to eliminate four of the five arbitrary constants. This gives

$$\begin{aligned} \hat{u} &= \hat{y} - \frac{\hat{y}^2}{2}, & \hat{k} &= C_4 \frac{\hat{y}^4 - 4\hat{y}}{12} + \frac{-2696\hat{y} + 840\hat{y}^2 + 336\hat{y}^5 - 504\hat{y}^6 + 560\hat{y}^7 - 195\hat{y}^8}{10080} \\ \hat{\omega} &= \frac{1}{6\hat{y}^2} + C_4 + \frac{8\hat{y} - 6\hat{y}^2 + 4\hat{y}^3 - \hat{y}^4}{12} \end{aligned} \quad (5.48)$$

Again as should be expected, there are an infinite number of solutions to any indeterminate fifth-order system of differential equations with only four boundary conditions. The remaining constant of integration C_4 can be evaluated only by applying a mathematically appropriate boundary condition. No amount of analysis applied to the differential equations, no matter how sophisticated, will ever yield a result from which the remaining arbitrary constant in Eq. (5.48) can be determined.

Notice from Eq. (5.47) that, analogous to the result obtained from Eq. (5.40), the general solution for $\hat{\omega}$ approaches $y=0$ in proportion to y^{-2} . From examination of either Eq. (5.47) or Eq. (5.48), it should be clear that none of the integration constants could ever be obtained by using the asymptotic behavior of $\hat{\omega}$ for $y \rightarrow 0$ as the fifth boundary condition for Eq. (5.46). In fact, because the behavior of $\hat{\omega}$ for $y \rightarrow 0$ depends only on the differential equations in Eq. (5.46), no boundary condition for $\hat{\omega}$ can be applied to Eq. (5.46) at $y=0$. Likewise, because the near-wall behavior of ω^+ depends only on the differential equations in Eq. (5.40), no wall boundary condition for ω^+ can be applied to Eq. (5.40). The remaining boundary condition for Eq. (5.46) at $y=0$ must be applied to \hat{k} . Similarly, the remaining wall boundary condition for Eq. (5.40) must be applied to k^+ .

At a smooth wall the correct no-slip boundary condition for completing the fifth-order formulation presented in Eq. (5.40) is $k^+(0) = 0$. The analogous boundary condition for Eq. (5.46) is $\hat{k}'(0) = 0$. It is easily shown that applying this wall boundary condition to the solution of Eq. (5.46) that is given in Eq. (5.48) yields $C_4 = -337/420$, and the complete unique solution is

$$\begin{aligned} \hat{u} &= y - \frac{y^2}{2}, \quad \hat{k} = \frac{840y^2 - 674y^4 + 336y^5 - 504y^6 + 560y^7 - 195y^8}{10080}, \\ \hat{\omega} &= \frac{1}{6y^2} + \frac{-337 + 280y - 210y^2 + 140y^3 - 35y^4}{420} \end{aligned} \quad (5.49)$$

Hence, we see that imposing two wall boundary conditions on \hat{k} and none on $\hat{\omega}$ is sufficient to determine a unique solution to the coupled fifth-order system of differential equations given in Eq. (5.46). There is no need to impose a wall boundary condition on $\hat{\omega}$, and it is incorrect to do so.

It can be shown numerically that the Wilcox 1998 $k-\omega$ formulation given in Eq. (5.40) exhibits behavior similar to that shown analytically for Eq. (5.46). For example, Figs. 5.10 and 5.11 show k^+ and ω^+ for five solutions, which all satisfy both Eqs. (5.40) and (5.45). These converged and grid-resolved solutions were obtained using the same second-order successive underrelaxation algorithms that were used to obtain the $k-\varepsilon$ solutions shown in Figs. 5.6 and 5.7. These results demonstrate that it is mathematically incorrect to use Eq. (5.45) as the sole substitute for the remaining boundary condition, which is required to obtain a unique solution to Eq. (5.40). Neither Eq. (5.45) nor any other relation obtained solely from the differential equations can be used to obtain a unique solution from the indeterminate $k-\omega$ formulation given in Eq. (5.40).

Like the Lam-Bremhorst $k-\varepsilon$ model, the wall damping functions for the Wilcox 1998 $k-\omega$ model decay rapidly with increasing y^+ , so the wall boundary condition $k^+(0) = 0$ has little impact on the predicted velocity profiles.

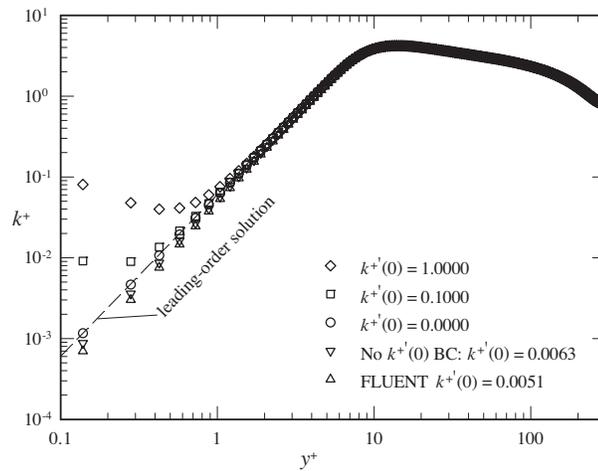


Fig. 5.10 Effects of wall boundary conditions on turbulent energy predicted from the Wilcox 1998 $k-\omega$ model.

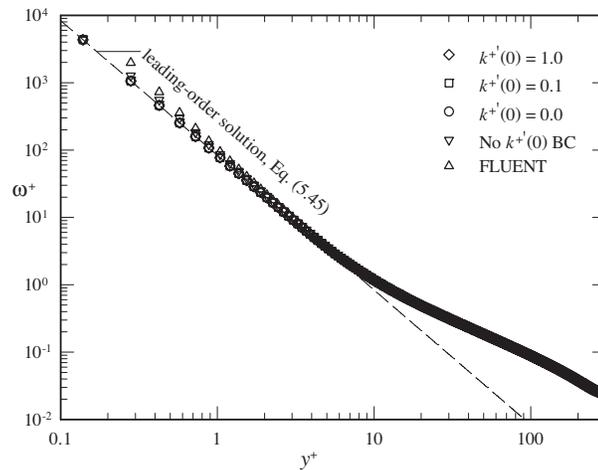


Fig. 5.11 Effects of wall boundary conditions on the turbulent dissipation frequency predicted from the Wilcox 1998 $k-\omega$ model.

VI. Comments on CFD Algorithms and Solutions

The reader may wonder how CFD algorithms have continued to obtain repeatable solutions for so many years if these traditional dissipation-based turbulence models have been implemented in such a way that many models are indeterminate. To this, the author wishes to make three comments.

First, CFD algorithms do not ensure that a unique solution is obtained. The CFD codes are usually written such that the computer is seeking one solution which satisfies the system of equations. Once a

solution is found, the algorithm exits. If multiple solutions exist for a given system of equations, the CFD algorithm will end once any one of the solutions is found, and will generally not report that other solutions exist. It is up to the user to ensure that the solution is unique. Therefore, solutions to these models have been reported for several years, although some have noted that there is often more difference between two different implementations of the same model than between two different models using the same implementation [87].

Second, most CFD algorithms seek to minimize the error in the system of equations rather than to find a solution that satisfies the equations perfectly. In fact, in most commercial codes, the user is able to specify a tolerance for the error at which the algorithm will consider the solution to be acceptable. This is likely the reason that solutions to the Lam-Bremhorst model have been obtained in the past. It has been shown that there is no solution to the Lam-Bremhorst model for turbulent flow that also satisfies the correct smooth-wall no-slip boundary conditions. However, solutions to this model have been obtained for many years by CFD algorithms. This is most probably due to the fact that most CFD algorithms seek only to minimize the error using finite-difference or finite-volume approaches.

Third, the wall-damping functions used by most models are designed in such a way as to force the turbulence properties near a wall to behave in a certain way. Therefore, these damping functions can curb the effects of indeterminate boundary conditions. For example, it has been shown that a deviation in the first derivative of k at the wall has a much more profound effect on the Launder-Sharma model than it does on the Lam-Bremhorst or Wilcox 1998 models. This is mainly due to the nature of the wall-damping functions used for each model. Therefore, the damping functions of dissipation-based models have likely reduced the effects of attempting to solve a system of equations that is short one boundary condition.

Even if a particular implementation or set of damping functions forces a CFD algorithm to converge to a solution that is close to the unenforced boundary condition, this is no excuse for implementing a model that is mathematically indeterminate. A full set of boundary conditions must always be enforced and it is incorrect to omit a boundary condition because it is difficult to enforce, or to choose an arbitrary condition

based on ease of implementation. Such neglectful behaviors have likely hindered progress that could have been made years ago in turbulence modeling.

VII. Summary and Conclusions

It has been shown that the traditional smooth-wall boundary conditions for dissipation-based turbulence models are incorrect. Traditional models generally enforce a boundary condition on the second turbulence variable, ε or ω , that is either physically incorrect or that can be directly derived from the differential equations. Because physics imposes two boundary conditions on the turbulent kinetic energy at a smooth wall and no boundary conditions on the dissipation at a smooth wall, it is incorrect to enforce a boundary condition on the second turbulence variable. In order for the solution to be unique, both boundary conditions required by physics at a smooth wall must be enforced on the turbulent kinetic energy. The near-wall behaviors developed from the differential equations are perfectly valid equations for ε and ω near the wall. However, it is incorrect to refer to these expressions as boundary conditions because they are derived directly from the differential equations. It is emphasized yet again that a boundary condition can never be developed from the differential equations themselves. A boundary condition is a condition that is imposed on the differential equations.

The algorithms commonly used to solve turbulence models are based on finite-difference or finite-volume CFD methods which attempt to minimize the error in a solution. These algorithms do not ensure that a unique solution has been found, nor do they ensure that a solution exists. This is likely the reason that solutions to traditional models have been obtained for years even when the correct boundary conditions have not been fully implemented. Even if a CFD algorithm converges to a solution near the correct solution without enforcing the correct boundary conditions, this is no excuse for not implementing the correct physically based boundary conditions for smooth walls.

The discussion of smooth-wall boundary conditions included here has focused on traditional models. Evaluating the performance of the Phillips energy-vorticity model in the presence of a smooth wall brings even more insight into the physics of turbulence near a perfectly smooth wall.

CHAPTER 6

PHILLIPS TWO-EQUATION MODEL CHARACTERISTICS

I. Introduction

Phillips has proposed several closure methods for his energy-vorticity model and a few of these are included in Chapter 2. Here we consider two closure methods and focus on their near-wall behavior for perfectly smooth walls. The near-wall behavior of any turbulence model is an important property of the model because this behavior determines the model prediction for the wall shear stress as well as the heat transfer at a wall. Traditionally, turbulence models have been developed for the case of perfectly smooth walls, and a select few have subsequently been modified to model rough walls. Because it is customary to evaluate the performance of a turbulence model for smooth walls before considering rough walls, the near-wall behavior of Phillips's energy-vorticity model was considered for perfectly smooth walls. This analysis presents interesting insight into the difference between smooth wall and rough wall turbulence characteristics.

II. RMS Turbulent Vorticity Closure: A k - $\tilde{\omega}$ Model

The Phillips energy-vorticity turbulence model could be closed as an RMS turbulent vorticity model, or k - $\tilde{\omega}$ model, including an expression for the kinematic eddy viscosity

$$\nu_t = C_\nu k / \tilde{\omega} \quad (6.1)$$

the turbulent-kinetic-energy equation

$$\begin{aligned} \frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla)k = & 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \nu \left(\tilde{\omega}^2 + \frac{4}{3} \nabla^2 k - 4 \nabla \cdot \{ \nabla \cdot [\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] \} \right) \\ & + \frac{5}{3} \nabla \cdot [(\nu + \nu_t / \sigma_k) \nabla k] - 2 \nabla \cdot \{ (\nu + \nu_t / \sigma_k) \nabla \cdot [\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] \} \end{aligned} \quad (6.2)$$

and a model version for the RMS-fluctuating-vorticity transport equation

$$\begin{aligned}
\frac{\partial \tilde{\omega}}{\partial t} + \bar{\mathbf{V}} \cdot \nabla \tilde{\omega} = & 2C_{\tilde{\omega}1} \nu_t \frac{\tilde{\omega}}{k} \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - C_{\tilde{\omega}2} \nu \frac{\tilde{\omega}}{k} \left(\tilde{\omega}^2 + \frac{4}{3} \nabla^2 k - 4 \nabla \cdot \{ \nabla \cdot [\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] \} \right) \\
& + C_{\tilde{\omega}3} \nabla \cdot [(\nu + \nu_t / \sigma_{\tilde{\omega}}) \nabla \tilde{\omega}] - C_{\tilde{\omega}4} \frac{\tilde{\omega}}{k} \nabla \cdot \{ (\nu + \nu_t / \sigma_k) \nabla \cdot [\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] \}
\end{aligned} \tag{6.3}$$

When the gradients in the flow are normal to the mean flow velocity vector, these equations can be greatly simplified. This simplified formulation is a good approximation for boundary layer flows, and holds exactly for fully developed flow in a pipe or channel. Applying this simplification gives what will be called the Phillips boundary-layer flow k - $\tilde{\omega}$ model

$$\begin{aligned}
\nu_t = C_\nu k / \tilde{\omega} \\
\frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) k = & 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \nu (\tilde{\omega}^2 + \frac{4}{3} \nabla^2 k) + \frac{5}{3} \nabla \cdot [(\nu + \nu_t / \sigma_k) \nabla k] \\
\frac{\partial \tilde{\omega}}{\partial t} + \bar{\mathbf{V}} \cdot \nabla \tilde{\omega} = & 2C_{\tilde{\omega}1} \nu_t \frac{\tilde{\omega}}{k} \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - C_{\tilde{\omega}2} \nu \frac{\tilde{\omega}}{k} (\tilde{\omega}^2 + \frac{4}{3} \nabla^2 k) + C_{\tilde{\omega}3} \nabla \cdot [(\nu + \nu_t / \sigma_{\tilde{\omega}}) \nabla \tilde{\omega}]
\end{aligned} \tag{6.4}$$

This model addresses several of the concerns with current turbulence modeling techniques discussed in Chapter 2. The correct smooth-wall boundary conditions can be used on the model, the turbulent kinetic energy equation alleviates a few of the concerns associated with the definition of dissipation, and the length scale is that associated with the energy-bearing eddies. In fact, this model alleviates all of the concerns mentioned at the beginning of Chapter 2 except for those numbered 7 and 8. The rough-wall characteristics of this model (Concern #8) is addressed in a later chapter, but will not be included in this section. The major concern with this formulation is that the RMS-fluctuating-vorticity transport equation given in Eq. (6.3) is no more rigorously derived than the second differential transport equations used in traditional models (Concern #7). Equation (6.3) was developed by analogy to Eq. (6.2) and was not derived rigorously from physical phenomenon. Such a method for developing transport equations for ε , ω , or ζ is commonly used in other two-equation turbulence models. However, this differential equation is likely an improvement over those equations used in other models because it can be expected that the transport of RMS fluctuating vorticity could be modeled using a differential transport equation. Dissipation-based models attempt to model the transport of dissipation with a differential transport equation, which is somewhat absurd because dissipation is not a transportable property of the flow. On the other hand, the RMS fluctuating vorticity is a

transportable property, and its transport could possibly be modeled by an equation similar to that given in Eq. (6.3).

A. Homogeneous, Isotropic Turbulence

Experiments show that the decay of turbulent kinetic energy can be expressed in the form

$$k = \frac{C}{(t+B)^a}, \quad \frac{\partial k}{\partial t} = -\frac{aC}{(t+B)^{a+1}} \quad (6.5)$$

where $a = 1.2 \pm 0.25$. For the case of decaying homogeneous turbulence, all gradients in the flow are zero, and the model transport equations reduce to the form

$$\frac{\partial k}{\partial t} = -\nu \tilde{\omega}^2 \quad (6.6)$$

$$\frac{\partial \tilde{\omega}}{\partial t} = -C_{\tilde{\omega}2} \nu \frac{\tilde{\omega}^3}{k} \quad (6.7)$$

Applying Eq. (6.5) to Eq. (6.6) gives

$$\tilde{\omega} = \frac{a^{1/2} C^{1/2}}{\nu^{1/2} (t+B)^{(a+1)/2}}, \quad \frac{d\tilde{\omega}}{dt} = -\frac{(a+1)a^{1/2} C^{1/2}}{2\nu^{1/2} (t+B)^{(a+3)/2}} \quad (6.8)$$

Using Eqs. (6.5) and (6.8) in Eq. (6.7) gives

$$C_{\tilde{\omega}2} = \frac{a+1}{2a} \quad (6.9)$$

Using $a = 1.2$ gives an initial estimate of $C_{\tilde{\omega}2} = 0.92$.

B. Smooth-Wall Behavior

The model can be written in Cartesian coordinates for steady, fully developed channel flow including the continuity, RANS, turbulent-kinetic-energy, and RMS-turbulent-vorticity equations as

$$\begin{aligned} \frac{d}{dy} \left[(\nu + \nu_t) \left(\frac{d\bar{V}_x}{dy} \right) \right] &= \frac{1}{\rho} \frac{d\hat{p}}{dx} \\ -\frac{d}{dy} \left[\left(\nu + \frac{\nu_t}{\sigma_k} \right) \left(\frac{5}{3} \frac{dk}{dy} \right) \right] &= \nu_t \left(\frac{d\bar{V}_x}{dy} \right)^2 - \nu \left[\tilde{\omega}^2 + \frac{4}{3} \frac{d}{dy} \left(\frac{dk}{dy} \right) \right] \\ -\frac{d}{dy} \left[\left(\nu + \frac{\nu_t}{\sigma_{\tilde{\omega}}} \right) \left(C_{\tilde{\omega}3} \frac{d\tilde{\omega}}{dy} \right) \right] &= C_{\tilde{\omega}1} \nu_t \frac{\tilde{\omega}}{k} \left(\frac{d\bar{V}_x}{dy} \right)^2 - C_{\tilde{\omega}2} \nu \frac{\tilde{\omega}}{k} \left[\tilde{\omega}^2 + \frac{4}{3} \frac{d}{dy} \left(\frac{dk}{dy} \right) \right] \end{aligned} \quad (6.10)$$

with boundary conditions

$$\bar{V}_x(0) = 0, \quad k(0) = 0, \quad \frac{dk}{dy}(0) = 0, \quad \frac{d\bar{V}_x}{dy}(L) = 0, \quad \frac{dk}{dy}(L) = 0, \quad \frac{d\tilde{\omega}}{dy}(L) = 0 \quad (6.11)$$

where L is the channel half-width. Applying the nondimensional definitions

$$y^+ \equiv \frac{u_\tau y}{\nu}, \quad u^+ \equiv \frac{\bar{V}_x}{u_\tau}, \quad \nu^+ \equiv \frac{\nu_t}{\nu}, \quad k^+ \equiv \frac{k}{u_\tau^2}, \quad \tilde{\omega}^+ \equiv \frac{\tilde{\omega} \nu}{u_\tau^2}, \quad R_\tau \equiv \frac{L u_\tau}{\nu}, \quad p^+ \equiv \frac{\nu}{\rho u_\tau^3} \frac{d\hat{p}}{dx} \quad (6.12)$$

the model equations can be written as

$$\begin{aligned} \nu^+ &= C_\nu \frac{k^+}{\tilde{\omega}^+}, \quad \frac{d}{dy^+} \left[(1 + \nu^+) \left(\frac{du^+}{dy^+} \right) \right] = p^+ \\ -\frac{d}{dy^+} \left[\left(1 + \frac{\nu^+}{\sigma_k} \right) \left(\frac{dk^+}{dy^+} \right) \right] &= \nu^+ \frac{3}{5} \left(\frac{du^+}{dy^+} \right)^2 - \left[\frac{3}{5} \tilde{\omega}^{+2} + \frac{4}{5} \frac{d}{dy^+} \left(\frac{dk^+}{dy^+} \right) \right] \\ -\frac{d}{dy^+} \left[\left(1 + \frac{\nu^+}{\sigma_\omega} \right) \left(\frac{d\tilde{\omega}^+}{dy^+} \right) \right] &= \frac{C_\nu C_{\tilde{\omega}1}}{C_{\tilde{\omega}3}} \left(\frac{du^+}{dy^+} \right)^2 - \frac{C_{\tilde{\omega}2}}{C_{\tilde{\omega}3}} \frac{\tilde{\omega}^{+3}}{k^+} - \frac{4C_{\tilde{\omega}2}}{3C_{\tilde{\omega}3}} \frac{\tilde{\omega}^+}{k^+} \frac{d}{dy^+} \left(\frac{dk^+}{dy^+} \right) \end{aligned} \quad (6.13)$$

with boundary conditions

$$u^+(0) = 0, \quad k^+(0) = 0, \quad \frac{dk^+}{dy^+}(0) = 0, \quad \frac{du^+}{dy^+}(R_\tau) = 0, \quad \frac{dk^+}{dy^+}(R_\tau) = 0, \quad \frac{d\tilde{\omega}^+}{dy^+}(R_\tau) = 0 \quad (6.14)$$

Near the channel wall, the Boussinesq-RANS equations reduce to the differential equation

$$(v + \nu_t) \frac{\partial \bar{V}_x}{\partial y} = u_\tau^2 \quad (6.15)$$

or in nondimensional form,

$$(1 + \nu^+) \frac{\partial u^+}{\partial y^+} = 1 \quad (6.16)$$

As $y^+ \rightarrow 0$, $\nu^+ \ll 1$, and $du^+/dy^+ \rightarrow 1$ and the k and $\tilde{\omega}$ equations can be simplified to

$$\begin{aligned} \frac{d^2 k^+}{dy^{+2}} &= 3\tilde{\omega}^{+2} - 3C_\nu \frac{k^+}{\omega^+} \\ \frac{d^2 \tilde{\omega}^+}{dy^{+2}} &= \frac{C_{\tilde{\omega}2}}{C_{\tilde{\omega}3}} \frac{\tilde{\omega}^{+3}}{k^+} + \frac{4C_{\tilde{\omega}2}}{3C_{\tilde{\omega}3}} \frac{\tilde{\omega}^+}{k^+} \frac{d^2 k^+}{dy^{+2}} - \frac{C_\nu C_{\tilde{\omega}1}}{C_{\tilde{\omega}3}} \end{aligned} \quad (6.17)$$

The leading-order term in the Taylor series expansion of the solution at the wall can be expressed for the two transport variables as

$$\begin{aligned}
k^+(y^+) &= Ay^{+a} + \dots, & k^{+'} &= Aay^{+a-1} + \dots, & k^{+''} &= Aa(a-1)y^{+a-2} + \dots \\
\tilde{\omega}^+(y^+) &= By^{+b} + \dots, & \tilde{\omega}^{+'} &= Bby^{+b-1} + \dots, & \tilde{\omega}^{+''} &= Bb(b-1)y^{+b-2} + \dots
\end{aligned}
\tag{6.18}$$

where the prime represents differentiation with respect to y^+ . Using Eq. (6.18) in Eq. (6.17) gives

$$\begin{aligned}
Aa(a-1)y^{+a-2} &= 3B^2y^{+2b} - 3C_v \frac{Ay^{+a}}{By^{+b}} \\
Bb(b-1)y^{+b-2} &= \frac{C_{\tilde{\omega}2}}{C_{\tilde{\omega}3}} \frac{B^3}{A} y^{+3b-a} + \frac{4C_{\tilde{\omega}2}}{3C_{\tilde{\omega}3}} Ba(a-1)y^{+b-2} - \frac{C_v C_{\tilde{\omega}1}}{C_{\tilde{\omega}3}}
\end{aligned}
\tag{6.19}$$

Equating the exponents gives

$$a - 2 = 2b = a - b \tag{6.20}$$

which can be solved to yield

$$a = 6, \quad b = 2 \tag{6.21}$$

Using this in Eq. (6.19) gives the relations

$$\begin{aligned}
A &= \frac{B^3}{10B + C_v} = \frac{C_v^2 (C_{\tilde{\omega}1} - C_{\tilde{\omega}2})^3}{8(25C_{\tilde{\omega}2} - C_{\tilde{\omega}3})^2 (5C_{\tilde{\omega}1} + 20C_{\tilde{\omega}2} - C_{\tilde{\omega}3})} \\
B &= \frac{C_v}{2} \frac{C_{\tilde{\omega}1} - C_{\tilde{\omega}2}}{(25C_{\tilde{\omega}2} - C_{\tilde{\omega}3})}
\end{aligned}
\tag{6.22}$$

Using Eqs. (6.21) and (6.22) in (6.18) gives the leading order terms

$$\begin{aligned}
k^+(y^+) &= \frac{C_v^2 (C_{\tilde{\omega}1} - C_{\tilde{\omega}2})^3}{8(25C_{\tilde{\omega}2} - C_{\tilde{\omega}3})^2 (5C_{\tilde{\omega}1} + 20C_{\tilde{\omega}2} - C_{\tilde{\omega}3})} y^{+6} + \dots \\
\tilde{\omega}^+(y^+) &= \frac{C_v}{2} \frac{C_{\tilde{\omega}1} - C_{\tilde{\omega}2}}{(25C_{\tilde{\omega}2} - C_{\tilde{\omega}3})} y^{+2} + \dots
\end{aligned}
\tag{6.23}$$

From this solution, the leading order relationship for the nondimensional turbulent eddy viscosity can be expressed as

$$\nu^+ = C_v \frac{k^+}{\tilde{\omega}^+} = \frac{C_v^2 (C_{\tilde{\omega}1} - C_{\tilde{\omega}2})^2}{4(25C_{\tilde{\omega}2} - C_{\tilde{\omega}3})(5C_{\tilde{\omega}1} + 20C_{\tilde{\omega}2} - C_{\tilde{\omega}3})} y^{+4} + \dots \tag{6.24}$$

Equation (6.24) shows that the turbulent eddy viscosity approaches the wall as y^{+4} .

Prandtl's mixing length theory suggests that the eddy viscosity can be written in terms of a mixing length, ℓ , which can be used in Eq. (6.15) to give

$$\left(\nu + \ell^2 \frac{\partial \bar{V}_x}{\partial y} \right) \frac{\partial \bar{V}_x}{\partial y} = u_\tau^2 \quad (6.25)$$

or in nondimensional form

$$\left(1 + \ell^{+2} \frac{\partial u^+}{\partial y^+} \right) \frac{\partial u^+}{\partial y^+} = 1 \quad (6.26)$$

where $\ell^+ \equiv \ell u_\tau / \nu$. Van Driest [88] proposed a near-wall empirical correlation for the mixing length which matches experimental data very well

$$\ell^+ = \kappa y^+ [1 - \exp(-y^+ / A^+)] \quad (6.27)$$

where κ is the von Kármán constant, generally accepted to be 0.41, and A^+ is a constant that depends on the wall roughness. For a smooth wall, A^+ is about 26. Using this along with Eq. (6.26) gives the Van Driest expression for the turbulent eddy viscosity

$$\nu^+ = \ell^{+2} \frac{\partial u^+}{\partial y^+} = \kappa^2 y^{+2} [1 - \exp(-y^+ / A^+)]^2 \frac{\partial u^+}{\partial y^+} \quad (6.28)$$

As $y^+ \rightarrow 0$, $\partial u^+ / \partial y^+ \rightarrow 1$, and the near-wall expansion of Eq. (6.28) simplifies to

$$\nu^+ = \frac{\kappa^2}{A^{+2}} y^{+4} \quad (6.29)$$

Therefore, the Van Driest equation also suggests that the eddy viscosity approaches the wall as y^{+4} .

Equating the constants in Eqs. (6.24) and (6.29) provides a relation for estimating closure coefficients

$$C_\nu = 2 \frac{\kappa}{A^+} \frac{[(25C_{\tilde{\omega}2} - C_{\tilde{\omega}3})(5C_{\tilde{\omega}1} + 20C_{\tilde{\omega}2} - C_{\tilde{\omega}3})]^{1/2}}{|C_{\tilde{\omega}1} - C_{\tilde{\omega}2}|} \quad (6.30)$$

Note that the product within the radical in the numerator must be positive. This can be accomplished in one of four ways:

- 1) $C_{\tilde{\omega}1} > C_{\tilde{\omega}2}$ and $C_{\tilde{\omega}3} < 25C_{\tilde{\omega}2}$
or
 - 2) $C_{\tilde{\omega}1} > C_{\tilde{\omega}2}$ and $C_{\tilde{\omega}3} > 5C_{\tilde{\omega}1} + 20C_{\tilde{\omega}2}$
or
 - 3) $C_{\tilde{\omega}1} < C_{\tilde{\omega}2}$ and $C_{\tilde{\omega}3} < 5C_{\tilde{\omega}1} + 20C_{\tilde{\omega}2}$
or
 - 4) $C_{\tilde{\omega}1} < C_{\tilde{\omega}2}$ and $C_{\tilde{\omega}3} > 25C_{\tilde{\omega}2}$
- (6.31)

These constraints are further confined by the constraint that A and B in Eq. (6.22) must both be positive. By inspection, if B is positive, A must also be positive. Therefore the constraints apply

$$\begin{aligned} 1) \quad & C_{\tilde{\omega}1} > C_{\tilde{\omega}2} \quad \text{and} \quad C_{\tilde{\omega}3} < 25C_{\tilde{\omega}2} \\ & \text{or} \\ 2) \quad & C_{\tilde{\omega}1} < C_{\tilde{\omega}2} \quad \text{and} \quad C_{\tilde{\omega}3} > 25C_{\tilde{\omega}2} \end{aligned} \tag{6.32}$$

These are the true constraints for the coefficients because these constraints are more confining than those in Eq. (6.31).

It is important to note that if the constraints given in Eq. (6.32) are adhered to, this k - $\tilde{\omega}$ model predicts that k approaches the wall as y^6 and $\tilde{\omega}$ approaches the wall as y^2 . This differs from the traditional dissipation-based turbulence models, which predict that k approaches the wall as y^2 . Note, however, that both dissipation-based models and this model predict that the turbulent eddy viscosity approaches the wall as y^4 . Because k is a turbulence quantity that can be measured experimentally, it may be insightful to consider how each of these predictions for k compares with the physics of the flow. It is commonly accepted that k approaches a smooth wall as y^2 for two main reasons. First, it can be shown analytically that the lowest order at which k could approach a smooth wall is y^2 . Second, experimental data seems to support this y^2 claim. Each of these seeming supporting evidences is discussed here.

It can be shown analytically that it is possible for k to approach a smooth wall as y^2 . This is commonly shown by first acknowledging that the fluctuating velocities must be zero at a smooth wall, just as the average velocities go to zero. In other words, at a no-slip wall $\tilde{V}_x = \tilde{V}_y = \tilde{V}_z = 0$ just as $V_x = V_y = V_z = 0$. The continuity equation for the fluctuating velocities can be written as

$$\frac{d\tilde{V}_x}{dx} + \frac{d\tilde{V}_y}{dy} + \frac{d\tilde{V}_z}{dz} = 0 \tag{6.33}$$

If y is the ordinate normal to the wall and x and z lie in the plane of the wall, gradients with respect to x and z will be much smaller than gradients with respect to y and the continuity equation can be simplified to

$$\frac{d\tilde{V}_y}{dy} \cong 0 \tag{6.34}$$

Therefore, the Taylor series expansion for each of the velocity fluctuation terms near the wall is assumed to take the form

$$\begin{aligned}\tilde{V}_x &= f_x(x, y, z)y + O(y^2) \\ \tilde{V}_y &= f_y(x, y, z)y^2 + O(y^3) \\ \tilde{V}_z &= f_z(x, y, z)y + O(y^2)\end{aligned}\tag{6.35}$$

where f_x , f_y , and f_z satisfy the governing equations of fluid motion and have an ensemble average of zero. Using these Taylor series expansions in the definition for the turbulent kinetic energy gives a near-wall approximation for k

$$k \equiv \frac{1}{2} \left(\overline{\tilde{V}_x^2} + \overline{\tilde{V}_y^2} + \overline{\tilde{V}_z^2} \right) = \frac{1}{2} \left(\overline{f_x^2} + \overline{f_y^2} \right) y^2 + O(y^3)\tag{6.36}$$

which shows that the leading-order term is proportional to y^2 . The oversight of this analysis is that it does not prove that k must approach the wall as y^2 . This analysis shows that it must approach the wall at a power of y of at least two but it does not show that it can't approach the wall at a higher power of y . In other words, this analysis puts a lower bound on the power of y but does not provide an upper bound. Therefore, k could approach a smooth wall as a higher power, although it could not approach a smooth wall as a lower power.

Some have cited experimental data that suggest that k approaches a wall as y^2 . The difficulty in justifying this asymptotic behavior of k on experimental data is twofold. First, it is difficult to get accurate measurements of the fluctuating components of velocity near a wall. Because of the nature of conducting experiments extremely close to a wall and the size of the instruments required for such measurements, very near-wall measurements for the fluctuating components of velocity are extremely difficult to obtain, and measurements are susceptible to considerable error. In order to claim near-wall asymptotic behavior, accurate measurements would be needed for $y^+ \ll 1$. However, such data is virtually non-existent in the literature. For example, Laufer [57,60] published results down to $y^+ \cong 1$, Metzger and Klewicki [89] published results down to $y^+ \cong 1$, and Marusic, McKeon, Molnkewitz, Nagib, Smits, and Sreenivasan [90] published results down to $y^+ \cong 7$. Second, although the experimental results that have been obtained are often reported as “smooth-wall” measurements, no measurements have ever been conducted on a perfectly

smooth wall. Those that report “smooth-wall measurements” generally mean “hydraulically smooth-wall measurements” for which it is assumed that roughness effects are negligible. It is true that roughness effects decay asymptotically as the wall roughness height approaches zero, and therefore, “hydraulically smooth” is a perfectly valid term for this region of wall roughness. However, although the fluctuating components of velocity become very small at a hydraulically smooth surface, they only go to exactly zero at a perfectly smooth surface. Therefore, the turbulent kinetic energy, k , cannot be exactly zero at a surface unless that surface is a perfectly smooth surface. Because no physical surface is perfectly smooth, although k may be very small at a hydraulically smooth surface, it must be a nonzero constant for every physical experiment ever conducted.

Based on the discussion presented here, it is the opinion of the author that there is insufficient evidence to claim that k approaches the wall as y^2 for a perfectly smooth surface. The behavior of k near a smooth wall as predicted by this k - $\tilde{\omega}$ model is just as plausible as the behavior of k predicted by dissipation-based models. More insight into the near-wall behavior of k can be gleaned from examining a system of equations that is analogous to this k - $\tilde{\omega}$ model.

Before continuing this discussion, it may be worth mentioning that if the near-wall solution of this model is to be numerically evaluated, a sixth-order algorithm must be used in order to capture the correct behavior of k . Most CFD algorithms in use today are second-order accurate and would therefore be incapable of correctly implementing this model. Traditional dissipation-based turbulence models show that k approaches the wall as y^2 , and therefore, second-order CFD implementations are capable of capturing the near-wall behavior of k for such models.

C. A Closed-Form Analog

Much can be learned about the behavior of the system of equations given in Eq. (6.13) by studying a system of equations that has similar characteristics in the limit as $y \rightarrow 0$. Consider the system of equations

$$\frac{d^2 u}{dy^2} = -1, \quad \frac{d^2 k}{dy^2} = -A_1 y^4 \left(\frac{du}{dy} \right)^2 + A_2 \tilde{\omega}^2, \quad \frac{d^2 \tilde{\omega}}{dy^2} = -B_1 \left(\frac{du}{dy} \right)^2 + B_2 \quad (6.37)$$

where A_1 , A_2 , B_1 , and B_2 are the model constants. The boundary conditions that are analogous to the fully developed channel or pipe flow boundary conditions are

$$u(0) = 0, \quad k(0) = 0, \quad \frac{dk}{dy}(0) = 0, \quad \frac{du}{dy}(1) = 0, \quad \frac{dk}{dy}(1) = 0, \quad \frac{d\tilde{\omega}}{dy}(1) = 0 \quad (6.38)$$

We wish to determine constants for this system of equations such that this system has similar near-wall behavior as the k - $\tilde{\omega}$ model given in Eq. (6.13). Integrating the u equation and applying the boundary conditions $u(0) = 0$ and $du/dy(1) = 0$ gives

$$u = y - \frac{y^2}{2}, \quad \frac{du}{dy} = 1 - y \quad (6.39)$$

Integrating the $\tilde{\omega}$ equation and using the solution for u gives

$$\frac{d\tilde{\omega}}{dy} = -\frac{B_1}{3}y^3 + B_1y^2 + (B_2 - B_1)y + C_1 \quad (6.40)$$

Applying the boundary condition $d\tilde{\omega}/dy(1) = 0$ gives

$$C_1 = \frac{B_1}{3} - B_2 \quad (6.41)$$

We wish the solution for $\tilde{\omega}$ to have a lowest order of y^2 . This requires

$$B_1 = 3B_2 \quad (6.42)$$

Using this relation and integrating again gives

$$\tilde{\omega} = -\frac{1}{4}B_2y^4 + B_2y^3 - y^2 + C_2 \quad (6.43)$$

Squaring $\tilde{\omega}$ and using it in the k equation gives

$$\begin{aligned} \frac{d^2k}{dy^2} = & \frac{1}{48}A_2B_2^2y^8 - \frac{1}{2}A_2B_2^2y^7 + \left(\frac{1}{2}A_2B_2^2 + A_2B_2^2 - A_1\right)y^6 + \left(-2A_2B_2^2 + 2A_1\right)y^5 \\ & + \left(-\frac{1}{2}A_2B_2C_2 + A_2B_2^2 - A_1\right)y^4 + 2A_2C_2B_2y^3 - 2A_2C_2B_2y^2 + A_2C_2^2 \end{aligned} \quad (6.44)$$

Integrating gives

$$\begin{aligned} \frac{dk}{dy} = & \frac{1}{432}A_2B_2^2y^9 - \frac{1}{16}A_2B_2^2y^8 + \frac{1}{7}\left(\frac{1}{2}A_2B_2^2 + A_2B_2^2 - A_1\right)y^7 + \frac{1}{6}\left(-2A_2B_2^2 + 2A_1\right)y^6 \\ & + \frac{1}{5}\left(-\frac{1}{2}A_2B_2C_2 + A_2B_2^2 - A_1\right)y^5 + \frac{1}{2}A_2C_2B_2y^4 - \frac{2}{3}A_2C_2B_2y^3 + A_2C_2^2y + C_3 \end{aligned} \quad (6.45)$$

Applying the boundary condition $dk/dy(0) = 0$ gives

$$C_3 = 0 \quad (6.46)$$

Applying the boundary condition $dk/dy(1) = 0$ gives

$$C_2 = \frac{12B_2 \pm \left[144B_2^2 - 180 \left(\frac{8}{7} B_2^2 - \frac{3}{7} \frac{A_1}{A_2} \right) \right]^{1/2}}{90} \quad (6.47)$$

Because we wish the solution for $\tilde{\omega}$ to have a lowest order of y^2 , the relation

$$C_2 = 0 = \frac{12B_2 \pm \left[144B_2^2 - 180 \left(\frac{8}{7} B_2^2 - \frac{3}{7} \frac{A_1}{A_2} \right) \right]^{1/2}}{90} \quad (6.48)$$

must hold. This is possible only if

$$B_2^2 = \frac{3}{8} \frac{A_1}{A_2} \quad (6.49)$$

However, this condition is not sufficient to force $C_2 = 0$. Even if the condition in Eq. (6.49) is met, there are two solutions for C_2

$$C_2 = 0, \quad \text{or} \quad C_2 = \frac{4}{15} B_2 \quad (6.50)$$

Using the solution $C_2 = 0$ in Eq. (6.45) and integrating gives

$$k = \frac{1}{4320} A_2 B_2^2 y^{10} - \frac{1}{144} A_2 B_2^2 y^9 + \frac{1}{56} \left(\frac{1}{2} A_2 B_2^2 + A_2 B_2^2 - A_1 \right) y^8 \\ + \frac{1}{42} \left(-2A_2 B_2^2 + 2A_1 \right) y^7 + \frac{1}{30} \left(-\frac{1}{2} A_2 B_2 C_2 + A_2 B_2^2 - A_1 \right) y^6 + C_4 \quad (6.51)$$

Applying the boundary condition $k(0) = 0$ gives

$$C_4 = 0 \quad (6.52)$$

Choosing

$$A_1 = -8, \quad A_2 = -3, \quad B_1 = -3, \quad B_2 = -1 \quad (6.53)$$

gives the system of equations

$$\frac{d^2 u}{dy^2} = -1, \quad \frac{d^2 k}{dy^2} = 8y^4 \left(\frac{du}{dy} \right)^2 - 3\tilde{\omega}^2, \quad \frac{d^2 \tilde{\omega}}{dy^2} = 3 \left(\frac{du}{dy} \right)^2 - 1 \quad (6.54)$$

with the solution

$$\begin{aligned}
 u &= y - \frac{y^2}{2} \\
 k &= \frac{1}{6}y^6 - \frac{5}{21}y^7 + \frac{1}{16}y^8 + \frac{1}{48}y^9 - \frac{1}{480}y^{10} \\
 \tilde{\omega} &= y^2 - y^3 + \frac{1}{4}y^4
 \end{aligned} \tag{6.55}$$

However, this is not a unique solution. Using the second solution from Eq. (6.50), $C_2 = 4B_2/15 = -4/15$ gives

$$\begin{aligned}
 u &= y - \frac{y^2}{2} \\
 k &= -\frac{8}{75}y^2 + \frac{2}{15}y^4 - \frac{2}{25}y^5 + \frac{9}{50}y^6 - \frac{5}{21}y^7 + \frac{1}{16}y^8 + \frac{1}{48}y^9 - \frac{1}{1440}y^{10} \\
 \tilde{\omega} &= -\frac{4}{15}y^2 - y^3 + \frac{1}{4}y^4
 \end{aligned} \tag{6.56}$$

Thus, the system of differential equations has two solutions, and a computer algorithm may tend to favor one solution over the other. Note that the solution in Eq. (6.56) results in negative values for k and $\tilde{\omega}$ near the wall. A similar anomaly may be exhibited by the system of equations given in Eq. (6.13). However, the correct solution to that system cannot have negative values for k or $\tilde{\omega}$. Therefore, only one viable solution should exist for that system of equations even if it also exhibits the characteristic of having a higher-order solution that is positive across the entire domain and a lower-order solution that is negative over some portion of the domain.

Note that the relationship between the closure coefficients given in Eqs. (6.42) and (6.49) are vital to creating a system of equations that exhibits the correct asymptotic behavior near a smooth wall. A similar characteristic here can also be observed in Eq. (6.23) for the k - $\tilde{\omega}$ model. For example, if $C_{\tilde{\omega}1} = C_{\tilde{\omega}2}$, the order at which k and $\tilde{\omega}$ approach the wall changes. Therefore, just like this closed-form example, the near-wall results of the k - $\tilde{\omega}$ turbulence model are sensitive to the closure coefficients chosen for the model.

Using an eighth-order finite-difference algorithm, the system of equations given in Eq. (6.54) with boundary conditions given in Eq. (6.38) was solved. The algorithm employs successive under-relaxation in an iterative manner until convergence was reached. In order to ensure that the solution given in Eq. (6.55)

was obtained rather than that in Eq. (6.56), the solutions for k and $\tilde{\omega}$ were limited to positive numbers following each iteration. Uniform grid spacing was used for grids with 51, 101, and 201 nodes. The results for k and $\tilde{\omega}$ are shown in Figs. 6.1 and 6.2.

The results are shown here on log-log plots so that the near-wall asymptotic behavior of k and $\tilde{\omega}$ can be observed. The exact solution given in Eq. (6.55) is included to demonstrate that the eighth-order algorithm used for the solution procedure is capable of capturing the high-order phenomenon near the wall. Note that even with a relatively coarse grid, the algorithm is capable of solving for the near-wall behavior of the solution. Similar near-wall behavior would be expected for the solution of the k - $\tilde{\omega}$ model, and therefore a high-order algorithm similar to that used for this solution would be needed to solve the k - $\tilde{\omega}$ model in the presence of a perfectly smooth wall. High-order characteristics of the Phillips energy-vorticity model near a perfectly smooth wall are exhibited even with other closure methods. Another promising closure method, the k - ζ model, is included here.

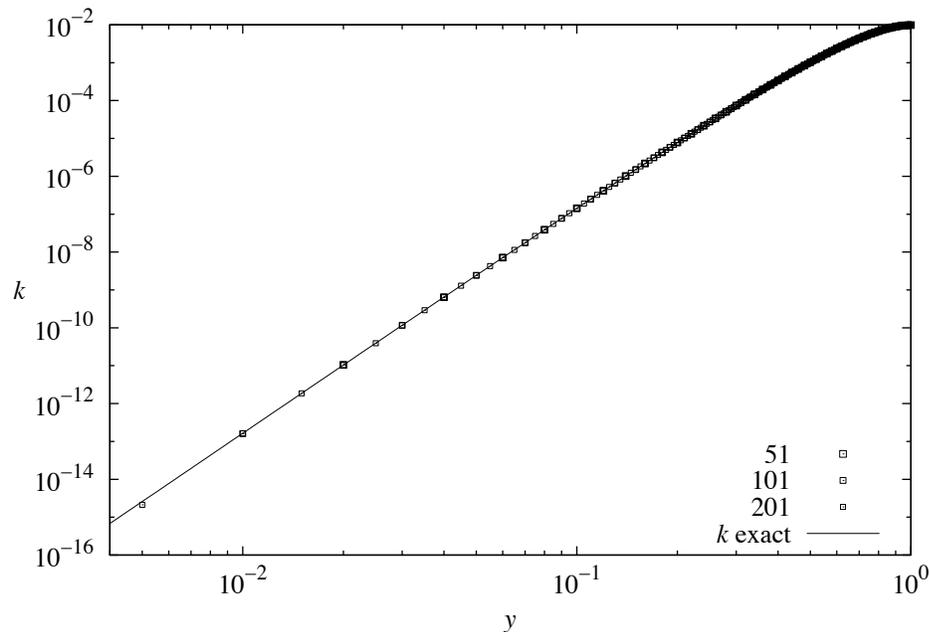


Fig. 6.1 Numerical solution for k in the closed-form analog to the k - $\tilde{\omega}$ model.

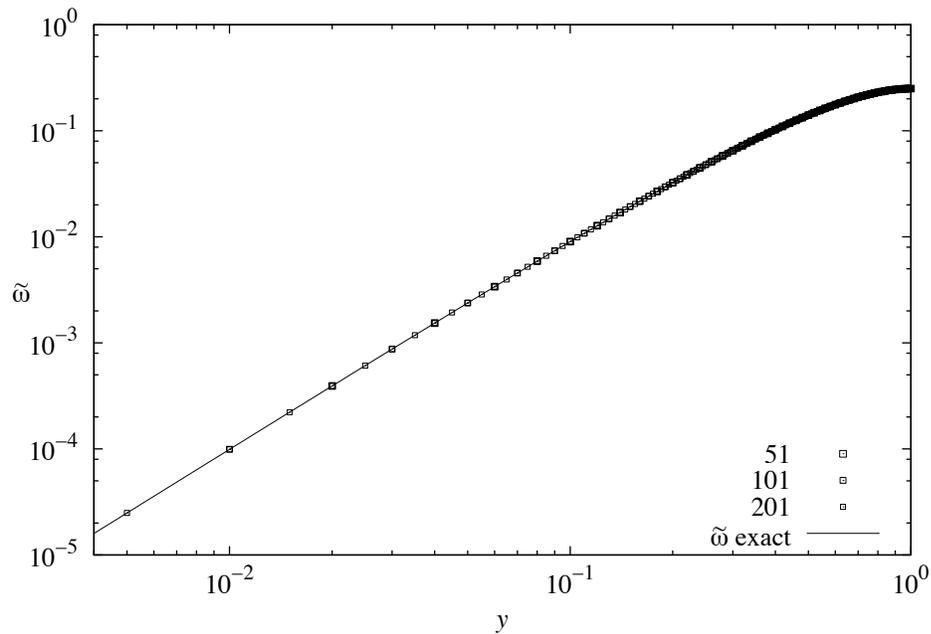


Fig. 6.2 Numerical solution for $\tilde{\omega}$ in the closed-form analog to the k - $\tilde{\omega}$ model.

III. Enstrophy Closure: A k - ζ Model

The Phillips energy-vorticity turbulence model could be closed as an energy-enstrophy model, or k - ζ model, including an expression for the kinematic eddy viscosity

$$v_t = C_v k / \sqrt{\zeta} \quad (6.57)$$

the turbulent-kinetic-energy equation

$$\begin{aligned} \frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) k = 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \nu \left(\zeta + \frac{4}{3} \nabla^2 k - 4\nabla \cdot \{ \nabla \cdot [v_t \bar{\mathbf{S}}(\bar{\mathbf{V}})] \} \right) \\ + \frac{5}{3} \nabla \cdot [(\nu + \nu_t / \sigma_k) \nabla k] - 2\nabla \cdot \{ (\nu + \nu_t / \sigma_k) \nabla \cdot [v_t \bar{\mathbf{S}}(\bar{\mathbf{V}})] \} \end{aligned} \quad (6.58)$$

and a model version for the enstrophy transport equation

$$\begin{aligned} \frac{\partial \zeta}{\partial t} + \bar{\mathbf{V}} \cdot \nabla \zeta = 2C_{\zeta 1} \nu_t \frac{\zeta}{k} \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - C_{\zeta 2} \nu \frac{\zeta}{k} \left(\zeta + \frac{4}{3} \nabla^2 k - 4\nabla \cdot \{ \nabla \cdot [v_t \bar{\mathbf{S}}(\bar{\mathbf{V}})] \} \right) \\ + C_{\zeta 3} \nabla \cdot [(\nu + \nu_t / \sigma_\zeta) \nabla \zeta] - C_{\zeta 4} \frac{\zeta}{k} \nabla \cdot \{ (\nu + \nu_t / \sigma_k) \nabla \cdot [v_t \bar{\mathbf{S}}(\bar{\mathbf{V}})] \} \end{aligned} \quad (6.59)$$

Applying the assumption that the gradients in the flow are normal to the mean flow velocity vector gives what will be called the Phillips boundary-layer flow k - ζ model

$$\begin{aligned}
\nu_t &= C_\nu k / \sqrt{\zeta} \\
\frac{\partial k}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) k &= 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \nu(\zeta + \frac{4}{3} \nabla^2 k) + \frac{5}{3} \nabla \cdot [(\nu + \nu_t / \sigma_k) \nabla k] \\
\frac{\partial \zeta}{\partial t} + \bar{\mathbf{V}} \cdot \nabla \zeta &= 2C_{\zeta 1} \nu_t \frac{\zeta}{k} \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - C_{\zeta 2} \nu \frac{\zeta}{k} (\zeta + \frac{4}{3} \nabla^2 k) + C_{\zeta 3} \nabla \cdot [(\nu + \nu_t / \sigma_\zeta) \nabla \zeta]
\end{aligned} \tag{6.60}$$

Again, this model alleviates all of the concerns mentioned in Chapter 2, except concerns number 7 and 8. The second transport equation used in this model is, again, not derived rigorously from physical phenomenon. However, because enstrophy is a transport property, it can be expected that its transport could be modeled in a form similar to that given in Eq. (6.59). This closure method will be considered here in terms of its behavior near a perfectly smooth wall but could be extended to rough walls at a later date.

A. Homogeneous, Isotropic Turbulence

For the case of decaying homogeneous turbulence, all gradients in the flow are zero, and the model transport equations reduce to the form

$$\frac{\partial k}{\partial t} = -\nu \zeta \tag{6.61}$$

$$\frac{\partial \zeta}{\partial t} = -C_{\zeta 2} \nu \frac{\zeta^2}{k} \tag{6.62}$$

Applying Eq. (6.5) to Eq. (6.61) gives

$$\zeta = \frac{aC}{\nu(t+B)^{a+1}}, \quad \frac{d\zeta}{dt} = -\frac{aC(a+1)}{\nu(t+B)^{a+2}} \tag{6.63}$$

Using Eqs. (6.5) and (6.63) in Eq. (6.62) gives

$$C_{\zeta 2} = a+1 \tag{6.64}$$

Using $a = 1.2$ gives the initial estimate $C_{\zeta 2} = 2.2$.

B. Smooth-Wall Behavior

The model can be written in Cartesian coordinates for steady, fully developed channel flow including the continuity, RANS, turbulent-kinetic-energy, and RMS-turbulent-vorticity equations as

$$\begin{aligned}
\frac{d}{dy} \left[(v + v_t) \left(\frac{d\bar{V}_x}{dy} \right) \right] &= \frac{1}{\rho} \frac{d\hat{p}}{dx} \\
-\frac{d}{dy} \left[\left(v + \frac{v_t}{\sigma_k} \right) \left(\frac{5}{3} \frac{dk}{dy} \right) \right] &= v_t \left(\frac{d\bar{V}_x}{dy} \right)^2 - v \left[\zeta + \frac{4}{3} \frac{d}{dy} \left(\frac{dk}{dy} \right) \right] \\
-\frac{d}{dy} \left[\left(v + \frac{v_t}{\sigma_\zeta} \right) \left(C_{\zeta 3} \frac{d\zeta}{dy} \right) \right] &= C_{\zeta 1} v_t \frac{\zeta}{k} \left(\frac{d\bar{V}_x}{dy} \right)^2 - C_{\zeta 2} v \frac{\zeta}{k} \left[\zeta + \frac{4}{3} \frac{d}{dy} \left(\frac{dk}{dy} \right) \right]
\end{aligned} \tag{6.65}$$

with boundary conditions

$$\bar{V}_x(0) = 0, \quad k(0) = 0, \quad \frac{dk}{dy}(0) = 0, \quad \frac{d\bar{V}_x}{dy}(L) = 0, \quad \frac{dk}{dy}(L) = 0, \quad \frac{d\zeta}{dy}(L) = 0 \tag{6.66}$$

Applying the nondimensional definitions given in Eq. (6.12) along with the definition

$$\zeta^+ \equiv \frac{\zeta v^2}{u_\tau^4} \tag{6.67}$$

the model equations for fully developed flow can be written as

$$\begin{aligned}
v^+ &= C_v k^+ / \sqrt{\zeta^+}, \quad \frac{d}{dy^+} \left[(1 + v^+) \left(\frac{du^+}{dy^+} \right) \right] = p^+ \\
-\frac{d}{dy^+} \left[\left(1 + \frac{v^+}{\sigma_k} \right) \left(\frac{dk^+}{dy^+} \right) \right] &= v^+ \frac{3}{5} \left(\frac{du^+}{dy^+} \right)^2 - \frac{3}{5} \left[\zeta^+ + \frac{4}{3} \frac{d}{dy^+} \left(\frac{dk^+}{dy^+} \right) \right] \\
-\frac{d}{dy^+} \left[\left(1 + \frac{v^+}{\sigma_\zeta} \right) \left(\frac{d\zeta^+}{dy^+} \right) \right] &= \frac{C_v C_{\zeta 1}}{C_{\zeta 3}} \sqrt{\zeta^+} \left(\frac{du^+}{dy^+} \right)^2 - \frac{C_{\zeta 2}}{C_{\zeta 3}} \frac{\zeta^+}{k^+} \left[\zeta^+ + \frac{4}{3} \frac{d}{dy^+} \left(\frac{dk^+}{dy^+} \right) \right]
\end{aligned} \tag{6.68}$$

with boundary conditions

$$u^+(0) = 0, \quad k^+(0) = 0, \quad \frac{dk^+}{dy^+}(0) = 0, \quad \frac{du^+}{dy^+}(R_\tau) = 0, \quad \frac{dk^+}{dy^+}(R_\tau) = 0, \quad \frac{d\zeta^+}{dy^+}(R_\tau) = 0 \tag{6.69}$$

The leading-order solutions for k^+ and ζ^+ can be helpful in evaluating closure coefficients. The leading-order term in the Taylor series expansion of the solution at the wall can be expressed for the two transport variables as

$$\begin{aligned}
k^+(y^+) &= Ay^{+a} + O(y^{+a+1}), \quad k^{+'} = Aay^{+a-1} + O(y^{+a}), \quad k^{+''} = Aa(a-1)y^{+a-2} + O(y^{+a-1}) \\
\zeta^+(y^+) &= By^{+b} + O(y^{+b+1}), \quad \zeta^{+'} = Bby^{+b-1} + O(y^{+b}), \quad \zeta^{+''} = Bb(b-1)y^{+b-2} + O(y^{+b-1})
\end{aligned} \tag{6.70}$$

where the prime represents differentiation with respect to y^+ . Near the channel wall, $y^+ \rightarrow 0$, $du^+/dy^+ \rightarrow 1$, and $\nu^+ \ll 1$, and the k and ζ equations can be simplified to

$$\begin{aligned} \frac{d^2 k^+}{dy^{+2}} &\cong 3\zeta^+ - 3C_\nu \frac{k^+}{\sqrt{\zeta^+}} \\ \frac{d^2 \zeta^+}{dy^{+2}} &\cong \frac{C_{\zeta 2}}{C_{\zeta 3}} \frac{\zeta^{+2}}{k^+} + \frac{4 C_{\zeta 2}}{3 C_{\zeta 3}} \frac{\zeta^+}{k^+} \frac{d^2 k^+}{dy^{+2}} - \frac{C_\nu C_{\zeta 1}}{C_{\zeta 3}} \sqrt{\zeta^+} \end{aligned} \quad (6.71)$$

Using the leading-order terms from Eq. (6.70) in Eq. (6.71) gives

$$\begin{aligned} Aa(a-1)y^{+a-2} &= 3By^{+b} - 3C_\nu \frac{Ay^{+a}}{B^{1/2}y^{+b/2}} \\ Bb(b-1)y^{+b-2} &= \frac{C_{\zeta 2}}{C_{\zeta 3}} \frac{B^2 y^{+2b}}{Ay^{+a}} + \frac{4 C_{\zeta 2}}{3 C_{\zeta 3}} \frac{By^{+b}}{Ay^{+a}} Aa(a-1)y^{+a-2} - \frac{C_\nu C_{\zeta 1}}{C_{\zeta 3}} B^{1/2} y^{+b/2} \end{aligned} \quad (6.72)$$

Equating the exponents gives

$$\begin{aligned} a-2 &= b = a-b/2 \\ b-2 &= 2b-a = b/2 \end{aligned} \quad (6.73)$$

which can be solved to give

$$a = 6, \quad b = 4 \quad (6.74)$$

Using this in Eq. (6.72) gives the relations

$$\begin{aligned} A &= \frac{C_\nu^2}{R_C(R_C - 10)^2} \\ B &= \frac{C_\nu^2}{(R_C - 10)^2} \end{aligned} \quad (6.75)$$

where

$$R_C \equiv \frac{(40C_{\zeta 2} - 12C_{\zeta 3} + 10C_{\zeta 1})}{(C_{\zeta 1} - C_{\zeta 2})} \quad (6.76)$$

Using Eqs. (6.75) and (6.76) in (6.70) gives the leading order terms

$$\begin{aligned} k^+(y^+) &= \frac{C_\nu^2}{R_C(R_C - 10)^2} y^{+6} + \mathcal{O}(y^{+7}) \\ \zeta^+(y^+) &= \frac{C_\nu^2}{(R_C - 10)^2} y^{+4} + \mathcal{O}(y^{+5}) \end{aligned} \quad (6.77)$$

Note that this form of the energy-vorticity model also predicts that k approaches a perfectly smooth wall as y^6 . From this solution, the leading order relationship for the nondimensional turbulent eddy viscosity can be expressed as

$$\nu^+ = C_\nu \frac{k^+}{\sqrt{\zeta^+}} = \frac{C_\nu^2}{R_C(R_C - 10)} y^{+4} + \dots \quad (6.78)$$

Equation (6.78) shows that the turbulent eddy viscosity approaches the wall as a constant multiplied by y^{+4} , just as the Van Driest equation suggests. Equating the constants in Eqs. (6.78) and (6.29) provides a relation for estimating closure coefficients

$$C_\nu = \frac{\kappa}{A^+} \sqrt{R_C(R_C - 10)} \quad (6.79)$$

The near-wall behavior of this model can perhaps be better understood by considering a closed-form system of equations that is analogous to this turbulence model.

C. A Closed-Form Analog

1. Perfectly Smooth Wall

Consider the three second-order differential equations that are analogous to Eq. (6.68)

$$\frac{d^2 u}{dy^2} = -1, \quad \frac{d^2 k}{dy^2} = -C_{k1} y^4 \left(\frac{du}{dy} \right)^2 + C_{k2} \zeta, \quad \frac{d^2 \zeta}{dy^2} = -C_{\zeta 1} y^2 \left(\frac{du}{dy} \right)^2 + C_{\zeta 2} y^2 \quad (6.80)$$

These equations are similar to the k - ζ turbulence model and require six boundary conditions to ensure a unique solution. In keeping with the similarity of the turbulence model, let us apply the following boundary conditions

$$u(0) = 0, \quad k(0) = 0, \quad \frac{dk}{dy}(0) = 0, \quad \frac{du}{dy}(1) = 0, \quad \frac{dk}{dy}(1) = 0, \quad \frac{d\zeta}{dy}(1) = 0 \quad (6.81)$$

This can be analytically integrated to yield

$$\begin{aligned}
u &= y - \frac{y^2}{2} \\
k &= -\frac{1}{56} \left(C_{k1} + \frac{C_{k2} C_{\zeta 1}}{30} \right) y^8 + \frac{1}{42} \left(\frac{C_{k2} C_{\zeta 1}}{10} + 2C_{k1} \right) y^7 + \frac{1}{30} \left(\frac{C_{k2} (C_{\zeta 2} - C_{\zeta 1})}{12} - C_{k1} \right) y^6 \\
&\quad + \frac{C_{k2}}{6} \left(\frac{C_{\zeta 1}}{30} - \frac{C_{\zeta 2}}{3} \right) y^3 + \frac{C_{k2}}{2} \left(\frac{1}{105} \frac{C_{k1}}{C_{k2}} - \frac{1}{84} C_{\zeta 1} + \frac{3}{20} C_{\zeta 2} \right) y^2 \\
\zeta &= -\frac{C_{\zeta 1}}{30} y^6 + \frac{C_{\zeta 1}}{10} y^5 + \frac{(C_{\zeta 2} - C_{\zeta 1})}{12} y^4 + \left(\frac{C_{\zeta 1}}{30} - \frac{C_{\zeta 2}}{3} \right) y + \frac{1}{105} \frac{C_{k1}}{C_{k2}} - \frac{1}{84} C_{\zeta 1} + \frac{3}{20} C_{\zeta 2}
\end{aligned} \tag{6.82}$$

If we desire a solution where ζ approaches the wall as y^4 and k approaches the wall as y^6 , the relations must hold

$$C_{\zeta 1} = 10C_{\zeta 2}, \quad C_{\zeta 2} = -\frac{4}{13} \frac{C_{k1}}{C_{k2}} \tag{6.83}$$

From this we see that the constants must be related not only to each other in the ζ equation, but the constants between the two transport equations must also be related if a solution is desired that matches the near-wall behavior of the k - ζ model. This can also be shown through a near-wall order analysis as follows.

Beginning with the equations

$$\frac{d^2 u}{dy^2} = -1, \quad \frac{d^2 k}{dy^2} = -C_{k1} y^4 \left(\frac{du}{dy} \right)^2 + C_{k2} \zeta, \quad \frac{d^2 \zeta}{dy^2} = -C_{\zeta 1} y^2 \left(\frac{du}{dy} \right)^2 + C_{\zeta 2} y^2 \tag{6.84}$$

Integrating the u -equation and applying the boundary condition $du/dy(1) = 0$ gives the set of equations

$$\begin{aligned}
\frac{du}{dy} &= 1 - y, & \frac{d^2 k}{dy^2} &= -C_{k1} y^4 + 2C_{k1} y^5 - C_{k1} y^6 + C_{k2} \zeta \\
\frac{d^2 \zeta}{dy^2} &= (C_{\zeta 2} - C_{\zeta 1}) y^2 + 2C_{\zeta 1} y^3 - C_{\zeta 1} y^4
\end{aligned} \tag{6.85}$$

Near the wall, these equations have the solution

$$\begin{aligned}
k(y) &= A_0 + A_1 y + A_2 y^2 + \dots \\
\zeta(y) &= B_0 + B_1 y + B_2 y^2 + \dots \\
u(y) &= C_0 + C_1 y + C_2 y^2 + \dots
\end{aligned} \tag{6.86}$$

Applying the wall boundary conditions

$$u(0) = 0, \quad k(0) = 0, \quad \frac{dk}{dy}(0) = 0 \tag{6.87}$$

gives the near-wall solution

$$\begin{aligned}
k(y) &= A_2 y^2 + A_3 y^3 + A_4 y^4 + \dots \\
\zeta(y) &= B_0 + B_1 y + B_2 y^2 + \dots \\
u(y) &= C_1 y + C_2 y^2 + C_3 y^3 + \dots
\end{aligned} \tag{6.88}$$

Using Eq. (6.87) in the u -equation and equating exponents gives

$$(C_1 + 2C_2 y + \dots) = 1 - y \rightarrow C_1 = 1, \quad C_2 = -\frac{1}{2} \tag{6.89}$$

This gives the new near-wall solution

$$\begin{aligned}
k(y) &= A_2 y^2 + A_3 y^3 + A_4 y^4 + \dots \\
\zeta(y) &= B_0 + B_1 y + B_2 y^2 + \dots \\
u(y) &= y - \frac{1}{2} y^2
\end{aligned} \tag{6.90}$$

with the differential equations

$$\begin{aligned}
\frac{d^2 k}{dy^2} &= -C_{k1} y^4 + 2C_{k1} y^5 - C_{k1} y^6 + C_{k2} \zeta \\
\frac{d^2 \zeta}{dy^2} &= (C_{\zeta 2} - C_{\zeta 1}) y^2 + 2C_{\zeta 1} y^3 - C_{\zeta 1} y^4
\end{aligned} \tag{6.91}$$

Using Eq. (6.90) in Eq. (6.91) gives

$$\begin{aligned}
(2A_2 + 6A_3 y + \dots) &= -C_{k1} y^4 + 2C_{k1} y^5 - C_{k1} y^6 + C_{k2} (B_0 + B_1 y + \dots) \\
(2B_2 + 6B_3 y + \dots) &= (C_{\zeta 2} - C_{\zeta 1}) y^2 + 2C_{\zeta 1} y^3 - C_{\zeta 1} y^4
\end{aligned} \tag{6.92}$$

This yields

$$A_2 = \frac{C_{k2}}{2} B_0, \quad A_3 = \frac{C_{k2}}{6} B_1, \quad B_2 = 0, \quad B_3 = 0 \tag{6.93}$$

and gives the near-wall solution

$$\begin{aligned}
k(y) &= \frac{C_{k2}}{2} B_0 y^2 + \frac{C_{k2}}{6} B_1 y^3 + A_4 y^4 + \dots \\
\zeta(y) &= B_0 + B_1 y + B_4 y^4 + \dots
\end{aligned} \tag{6.94}$$

Repeating this process will never tell us anything about B_0 and B_1 . However, the process can be repeated to yield

$$\begin{aligned}
(C_{k2} B_0 + C_{k2} B_1 y + 12A_4 y^2 + 20A_5 y^3 + 30A_6 y^4 + 42A_7 y^5 + 56A_8 y^6 + \dots) &= \\
-C_{k1} y^4 + 2C_{k1} y^5 - C_{k1} y^6 + C_{k2} (B_0 + B_1 y + B_4 y^4 + B_5 y^5 + B_6 y^6 + \dots) & \\
(12B_4 y^2 + 20B_5 y^3 + 30B_6 y^4 + \dots) &= (C_{\zeta 2} - C_{\zeta 1}) y^2 + 2C_{\zeta 1} y^3 - C_{\zeta 1} y^4
\end{aligned} \tag{6.95}$$

This gives

$$\begin{aligned}
 A_4 = 0, \quad A_5 = 0, \quad A_6 = \frac{1}{30} \left(\frac{C_{k2}(C_{\zeta2} - C_{\zeta1})}{12} - C_{k1} \right), \quad A_7 = \frac{1}{42} \left(2C_{k1} + \frac{C_{k2}C_{\zeta1}}{10} \right) \\
 A_8 = -\frac{1}{56} \left(C_{k1} + \frac{C_{k2}C_{\zeta1}}{30} \right), \quad A_{9-\infty} = 0 \\
 B_4 = \frac{(C_{\zeta2} - C_{\zeta1})}{12}, \quad B_5 = \frac{C_{\zeta1}}{10}, \quad B_6 = -\frac{C_{\zeta1}}{30}, \quad B_{7-\infty} = 0
 \end{aligned} \tag{6.96}$$

with the solution

$$\begin{aligned}
 k(y) &= \frac{C_{k2}}{2} B_0 y^2 + \frac{C_{k2}}{6} B_1 y^3 + \frac{1}{30} \left(\frac{C_{k2}(C_{\zeta2} - C_{\zeta1})}{12} - C_{k1} \right) y^6 \\
 &\quad + \frac{1}{42} \left(2C_{k1} + \frac{C_{k2}C_{\zeta1}}{10} \right) y^7 - \frac{1}{56} \left(C_{k1} + \frac{C_{k2}C_{\zeta1}}{30} \right) y^8 \\
 \zeta(y) &= B_0 + B_1 y + \frac{(C_{\zeta2} - C_{\zeta1})}{12} y^4 + \frac{C_{\zeta1}}{10} y^5 - \frac{C_{\zeta1}}{30} y^6
 \end{aligned} \tag{6.97}$$

The final two constants, B_0 and B_1 , are found by applying the two boundary conditions at the centerline

$$\frac{dk}{dy}(1) = 0, \quad \frac{d\zeta}{dy}(1) = 0 \tag{6.98}$$

Taking the first derivative of Eq. (6.97) and applying Eq. (6.98) gives

$$\begin{aligned}
 B_0 &= \frac{1}{105} \frac{C_{k1}}{C_{k2}} - \frac{1}{84} C_{\zeta1} + \frac{3}{20} C_{\zeta2} \\
 B_1 &= \frac{1}{3} \left(\frac{C_{\zeta1}}{10} - C_{\zeta2} \right)
 \end{aligned} \tag{6.99}$$

which gives the complete solution

$$\begin{aligned}
 u &= y - \frac{y^2}{2} \\
 k &= \frac{C_{k2}}{2} \left(\frac{1}{105} \frac{C_{k1}}{C_{k2}} - \frac{1}{84} C_{\zeta1} + \frac{3}{20} C_{\zeta2} \right) y^2 + \frac{C_{k2}}{18} \left(\frac{C_{\zeta1}}{10} - C_{\zeta2} \right) y^3 \\
 &\quad + \frac{1}{30} \left(\frac{C_{k2}(C_{\zeta2} - C_{\zeta1})}{12} - C_{k1} \right) y^6 + \frac{1}{42} \left(2C_{k1} + \frac{C_{k2}C_{\zeta1}}{10} \right) y^7 - \frac{1}{56} \left(C_{k1} + \frac{C_{k2}C_{\zeta1}}{30} \right) y^8 \\
 \zeta &= \frac{1}{105} \frac{C_{k1}}{C_{k2}} - \frac{1}{84} C_{\zeta1} + \frac{3}{20} C_{\zeta2} + \frac{1}{3} \left(\frac{C_{\zeta1}}{10} - C_{\zeta2} \right) y + \frac{(C_{\zeta2} - C_{\zeta1})}{12} y^4 + \frac{C_{\zeta1}}{10} y^5 - \frac{C_{\zeta1}}{30} y^6
 \end{aligned} \tag{6.100}$$

Again we find that if we desire a solution where ζ approaches the wall as y^4 and k approaches the wall as y^6 , the relations given in Eq. (6.83) must hold. It is interesting to note also that the same set of closure

coefficients that would cause k to approach the wall at a different power also cause ζ to approach the wall at a different power. As was the case with the previous closed-form system of equations given in Eq. (6.37), the behavior of this closed-form system can be very sensitive to the closure coefficients chosen for the model. Changing the closure coefficients can change the fundamental characteristics of the model near the wall.

2. Rough Wall

An alternate solution to this system of equations can be developed by applying a different set of boundary conditions that may be more appropriate for a rough-wall turbulence model. Such an exercise may provide insight into the behavior of the k - ζ model for rough walls. Consider the three second-order differential equations

$$\frac{d^2 u}{dy^2} = -1, \quad \frac{d^2 k}{dy^2} = -C_{k1} y^4 \left(\frac{du}{dy} \right)^2 + C_{k2} \zeta, \quad \frac{d^2 \zeta}{dy^2} = -C_{\zeta 1} y^2 \left(\frac{du}{dy} \right)^2 + C_{\zeta 2} y^2 \quad (6.101)$$

Again, these equations require six boundary conditions to ensure a unique solution. In keeping with the similarity of a rough-wall turbulence model, let us apply the following boundary conditions

$$u(0) = 0, \quad k(0) = k_0, \quad \zeta(0) = \zeta_0, \quad \frac{du}{dy}(1) = 0, \quad \frac{dk}{dy}(1) = 0, \quad \frac{d\zeta}{dy}(1) = 0 \quad (6.102)$$

This system can be solved to yield

$$\begin{aligned} u &= y - \frac{y^2}{2} \\ k &= -\frac{1}{56} \left(C_{k1} + \frac{C_{k2} C_{\zeta 1}}{30} \right) y^8 + \frac{1}{42} \left(\frac{C_{k2} C_{\zeta 1}}{10} + 2C_{k1} \right) y^7 + \frac{1}{30} \left(\frac{C_{k2} (C_{\zeta 2} - C_{\zeta 1})}{12} - C_{k1} \right) y^6 \\ &\quad + \frac{C_{k2}}{18} \left(\frac{C_{\zeta 1}}{10} - C_{\zeta 2} \right) y^3 + \frac{C_{k2} \zeta_0}{2} y^2 + C_{k2} \left(\frac{1}{105} \frac{C_{k1}}{C_{k2}} - \frac{1}{84} C_{\zeta 1} + \frac{3}{20} C_{\zeta 2} - \zeta_0 \right) y + k_0 \\ \zeta &= -\frac{C_{\zeta 1}}{30} y^6 + \frac{C_{\zeta 1}}{10} y^5 + \frac{(C_{\zeta 2} - C_{\zeta 1})}{12} y^4 + \frac{1}{3} \left(\frac{C_{\zeta 1}}{10} - C_{\zeta 2} \right) y + \zeta_0 \end{aligned} \quad (6.103)$$

This solution differs from that given in Eq. (6.100) only by the boundary conditions that were applied. For the smooth-wall solution given in Eq. (6.100), two boundary conditions were applied to the k equation

at the wall, and none were applied to the ζ equation at the wall. If the second boundary condition is removed on k in favor of a wall boundary condition for ζ , the solution shown in Eq. (6.103) results.

Expressing the solution in the form shown in Eq. (6.103) allows some interesting characteristics of the solution to be observed. For example, it can be seen that even if $k_0 = 0$ and $\zeta_0 = 0$, certain combinations of closure coefficients can cause the solutions for k and ζ to approach the wall at various powers of y . If a solution is desired that is analogous to the system of equations given in Eq. (6.68) near a perfectly smooth wall, the relations must hold exactly

$$k_0 = 0, \quad \zeta_0 = 0, \quad C_{\zeta 1} = 10C_{\zeta 2}, \quad C_{\zeta 2} = -\frac{4}{13} \frac{C_{k1}}{C_{k2}} \quad (6.104)$$

Additionally if the closure coefficients are chosen to satisfy Eq. (6.83), any nonzero value for ζ_0 results in a solution for k that has components proportional to y^1 and y^2 . Thus we see that the solution of these equations near a perfectly smooth wall is sensitive to the closure coefficients chosen for the model. On the other hand, for rough walls, $k_0 \neq 0$ and $\zeta_0 \neq 0$, and the solution very near a wall becomes less sensitive to the closure coefficients. If this behavior is indicative of the behavior of the k - ζ model for rough walls, it may be beneficial to study the k - ζ model in the presence of rough walls before attempting to develop the model coefficients for perfectly smooth walls.

IV. Summary and Conclusions

Two formulations of Phillips energy-vorticity model have been discussed in this chapter. These include the k - $\tilde{\omega}$ and k - ζ formulations which each address all the concerns presented at the beginning of Chapter 2 except for concerns numbered 7 and 8. Both of these formulations predict that k approaches the wall as y^6 , which goes against mainstream acceptance that k should approach a smooth wall as y^2 . It has been shown that the analysis and experimental data often cited in support of this mainstream assertion is insufficient to prove the y^2 behavior, and therefore, these models could give insight into the true behavior of k near a perfectly smooth wall.

Two closed-form systems of equations that are analogous to the turbulence model formulations were also presented in this chapter. These closed-form solutions exhibit behaviors that suggest why solutions to the $k-\tilde{\omega}$ and $k-\zeta$ models for perfectly smooth walls may be difficult to obtain. It has been shown that multiple solutions exist for the closed-form analog to the $k-\tilde{\omega}$ model and that solutions to both closed-form systems can be highly sensitive to the closure coefficients chosen. These closure coefficients can determine the order at which the models approach a perfectly smooth wall. Additionally, application of boundary conditions similar to those that would be used at a rough wall may alleviate some of the difficulties associated with the turbulence models near a wall.

Because this near-wall region for the turbulence models presented in this chapter seem to be highly sensitive to closure coefficients and can even be sensitive to solution algorithms, the work from this point on focuses on the behavior of turbulence models in the presence of rough walls rather than perfectly smooth walls. Such an approach will address Concern #8 mentioned in Chapter 2. The analysis in this chapter suggests that developing a turbulence model for rough walls and extending it to hydraulically smooth walls by looking at its asymptotic behavior as $k_0 \rightarrow 0$ may be more straight forward than developing a turbulence model for perfectly smooth walls and extending the model to handle surface roughness. Extending this concept one step further, it may make even more sense to develop a turbulence model for high roughness Reynolds numbers where certain flow properties become independent of roughness Reynolds number (as shown in Fig. 2.1) and extend the model to lower roughness Reynolds numbers as the model matures. This is the philosophy pursued in the following chapter where a turbulence model for fully rough pipe flow is presented that is based on the Phillips energy-vorticity model.

CHAPTER 7

DEVELOPMENT OF THE PHILLIPS k - λ MODEL FOR FULLY ROUGH PIPE FLOW**I. Introduction**

It has been shown in previous chapters that modeling turbulent behavior near perfectly smooth walls can present significant numerical difficulties. Additionally, because there is no such thing as a perfectly smooth wall, all experimental data available has been taken near walls with some degree of roughness. Furthermore, because certain flow properties become independent of roughness Reynolds number at high roughness Reynolds numbers, the argument has been made that developing a model for fully rough flow may be more straight forward than developing a model that exhibits the correct behavior near a perfectly smooth wall. Such a model could then be extended to lower roughness Reynolds numbers, perhaps even to the hydraulically smooth wall asymptote. The first step in this process is to develop a turbulence model that is consistent for high roughness Reynolds numbers.

Many wall-bounded flows including channel, boundary layer, and pipe flows have been used to evaluate the effects of surface roughness. However, fully developed flow in a pipe has historically been the foundational case and has been studied in great detail. This flow scenario makes an ideal case for evaluating the effects of surface roughness because the bulk flow properties such as the mass flow rate and wall shear stress are easily measured in an experimental setting. A wealth of experimental data exists for a range of wall roughness values from which empirical relations have been developed. Because flow in a pipe is so well understood and established, any viable turbulence model should be capable of predicting the bulk flow properties for this flow scenario. This scenario was chosen as the beginning point in the development of the Phillips k - λ model. The following sections present an overview of some of the fundamental work on rough pipe flow, and conclude with the development of one possible closure for the Phillips k - λ model.

II. Rough Pipe Flow

A. Fundamental Relations and Definitions

Fully developed pipe flow is one of the best-understood turbulent flow cases. As flow enters a pipe, a boundary layer develops along the pipe wall. As the flow moves downstream, the boundary layer eventually fills the entire pipe, and the flow reaches the fully developed state. In this state, the gradients of the mean turbulent properties are zero in both the azimuthal and axial directions of the pipe, and the case can be simplified to a one-dimensional flow as a function of pipe radius. The governing equations for fully developed flow in a pipe are developed in detail in Appendix C. A few of the most important relations are included here.

The Boussinesq-RANS equations for fully developed pipe flow can be simplified to a single equation and the no-slip wall boundary condition

$$\frac{d\bar{V}_z}{dr} = -\frac{u_\tau^2}{(\nu + \nu_t)} \frac{r}{R}, \quad \bar{V}_z(R) = 0 \quad (7.1)$$

It can be shown that the wall shear stress is related to the pressure drop in the pipe according to

$$\tau_w \equiv -\mu \left. \frac{d\bar{V}_z}{dr} \right|_{r=R} = -\frac{R}{2} \frac{d\hat{p}}{dz} \quad (7.2)$$

The pressure drop can easily be measured along the length of a pipe and the wall shear stress directly computed. The wall shear stress is generally expressed in terms of the Darcy friction factor

$$\text{Darcy friction factor} \equiv 4C_f \equiv \frac{8\tau_w}{\rho \bar{V}_{\text{bulk}}^2} \quad (7.3)$$

where C_f is the Fanning friction factor also called the skin-friction coefficient and the bulk velocity is defined as

$$\bar{V}_{\text{bulk}} \equiv \frac{2}{R^2} \int_{r=0}^R \bar{V}_z r dr \quad (7.4)$$

The bulk Reynolds number is defined as

$$\text{Re} \equiv \bar{V}_{\text{bulk}} D_h / \nu \quad (7.5)$$

where

$$D_h = 2R \quad (7.6)$$

is the hydraulic diameter based on the pipe radius, R . Experimentally, the bulk velocity can be measured from the mass flow rate or volume flow rate through the pipe. The friction velocity is defined as

$$u_\tau \equiv \sqrt{\tau_w/\rho} \quad (7.7)$$

and is commonly used to nondimensionalize many flow properties. The following nondimensional definitions are useful for pipe flow

$$\begin{aligned} \hat{r} &\equiv \frac{r}{R}, \quad R_\tau \equiv \frac{u_\tau R}{\nu}, \quad u^+ \equiv \frac{\bar{V}_z}{u_\tau}, \\ y^+ &\equiv \frac{u_\tau(R-r)}{\nu} = \frac{u_\tau R(1-\hat{r}/R)}{\nu} = R_\tau(1-\hat{r}), \\ p^+ &\equiv \frac{d\hat{p}}{dz} \frac{\nu}{\rho u_\tau^3} = -\frac{2\nu}{u_\tau^3} \frac{u_\tau^2}{R} = -\frac{2\nu}{u_\tau R} = -\frac{2}{R_\tau} \end{aligned} \quad (7.8)$$

Using these definitions, the Boussinesq-RANS equation including the no-slip wall boundary condition can be written in nondimensional forms as

$$\begin{aligned} \frac{du^+}{d\hat{r}} &= -\frac{R_\tau \hat{r}}{(1+\nu^+)}, \quad u^+ \Big|_{\hat{r}=1} = 0 \\ \frac{du^+}{dy^+} &= \frac{(1-y^+/R_\tau)}{(1+\nu^+)}, \quad u^+ \Big|_{y^+=0} = 0 \end{aligned} \quad (7.9)$$

and the bulk velocity, bulk Reynolds number, and Darcy friction factor can be expressed as

$$u_{\text{bulk}}^+ \equiv \frac{\bar{V}_{\text{bulk}}}{u_\tau} \equiv 2 \int_{\hat{r}=0}^1 u^+ \hat{r} d\hat{r}, \quad \text{Re}_e \equiv 2u_{\text{bulk}}^+ R_\tau, \quad 4C_f \equiv \frac{8}{(u_{\text{bulk}}^+)^2} \quad (7.10)$$

One of the most well-established properties of fully developed turbulent pipe flow is the relation between the wall shear stress, the bulk Reynolds number, and the wall roughness. The wall roughness is commonly characterized in terms of the ratio of the equivalent sand grain roughness to the pipe diameter. Here we define this ratio as

$$k_r \equiv \frac{k_s}{2R} \quad (7.11)$$

This is commonly termed the relative roughness. Modern notation has elected to express the relative roughness as $k_r = \varepsilon/D$ where ε denotes the surface roughness and D is the pipe diameter. Although this notation has gained popularity and is used in several modern fluid mechanics books, the classical notation shown in Eq. (7.11) will be used throughout this work to avoid confusion with the notation for dissipation which is nearly universally given the symbol ε . The roughness Reynolds number is defined as

$$k_s^+ \equiv \frac{u_\tau k_s}{\nu} = 2k_r R_\tau \quad (7.12)$$

Equation (7.12) can be rearranged to yield a convenient expression for evaluating the shear Reynolds number, R_τ , as a function of the relative roughness and roughness Reynolds number.

$$R_\tau = \frac{k_s^+}{2k_r} \quad (7.13)$$

It is easily shown that combining Eqs. (7.10) and (7.12) gives an expression for the roughness Reynolds number as a function of the relative roughness, Reynolds number, and friction factor

$$k_s^+ = k_r R_e \sqrt{4C_f/8} \quad (7.14)$$

These terms are used throughout this work, although modern convention deviates somewhat from these original definitions.

B. Mixing-Length Theory

The foundational theory from which turbulent flow in pipes was originally studied is the mixing-length theory of Prandtl [22]. This theory is built on a semi-empirical relation that the turbulent eddy viscosity can be expressed as

$$\nu_t = \ell^2 \left| \frac{d\bar{V}_z}{dy} \right| \quad (7.15)$$

where ℓ is called the mixing length. Applying the definitions given in Eq. (7.8) along with $\nu^+ \equiv \nu_t/\nu$ and $\ell^+ \equiv u_\tau \ell/\nu$, Eq. (7.15) can be written

$$\nu^+ = \ell^{+2} \left| \frac{du^+}{dy^+} \right| = \frac{\ell^{+2}}{R_\tau} \left| \frac{du^+}{d\hat{r}} \right| \quad (7.16)$$

The mixing length cannot be measured directly, and therefore is estimated from experimental results for the velocity profile. Using Eq. (7.16) in Eq. (7.9) gives

$$\left(1 + \ell^{+2} \left| \frac{du^+}{dy^+} \right| \right) \frac{du^+}{dy^+} = 1 - \frac{y^+}{R_\tau}, \quad u^+ \Big|_{y^+=0} = 0 \quad (7.17)$$

Mixing-length theory formed the foundation for much of the historical work on flows in smooth and rough pipes. This theory is still respected as a central part of turbulence modeling and will be referred to in the following pages.

C. The Nikuradse Number

Much of what is understood today about rough pipe flow is based on the foundational research of one of Ludwig Prandtl's students, Johann Nikuradse [59], who performed an immense amount of experimental work on pipes with roughened walls. Nikuradse roughened the walls of several pipes with sand grains of specific sizes and was careful to minimize the variation in the sand grain size within a given pipe. These pipes were tested over a range of Reynolds numbers to produce a substantial set of experimental data for study and correlation. From this set of data, Nikuradse found relations for the eddy viscosity and mixing length. Near the wall, where measurements for the mixing length are most accurate, Nikuradse found that the mixing length was a function of both the distance from the wall and the roughness height

$$\ell = \kappa(y + \gamma k_s) \quad (7.18)$$

Given values for the two constants κ and γ , Eq. (7.18) can be used in Eq. (7.17) and integrated to yield a velocity distribution. However, an analytical solution to this integral does not exist and numerical methods were not practical at the time of Nikuradse. Therefore, two assumptions were made by Nikuradse to simplify Eq. (7.17). First, it was assumed that the turbulent eddy viscosity is much greater than the molecular viscosity over the entire pipe. This will be referred to here as the fully rough flow approximation. This assumption is generally true in the bulk-flow region of turbulent pipe flow, but is also true near the wall for fully rough flows. In the near-wall region, y^+/R_τ is small compared to unity. Applying this

assumption over the entire flow drastically simplifies the problem, and forms the second assumption applied by Nikuradse. Applying these two assumptions to Eq. (7.17) yields

$$\ell^+ \frac{du^+}{dy^+} = 1, \quad u^+ \Big|_{y^+=0} = 0 \quad (7.19)$$

which is strictly true only in the fully rough near-wall region of the flow. However, applying Eq. (7.18) to Eq. (7.19) gave an expression that could be analytically integrated to yield a velocity profile. This expression was then matched to experimental results for the velocity profile to yield the historical values for the constants $\kappa = 0.400$ and $\gamma = 0.0334$.

Once the velocity profile was obtained, it could be analytically integrated to yield the bulk velocity. Using this in Eq. (7.10) gives a relation for the friction factor

$$4C_f = [2.04 \log_{10}(R/k_s) + 1.68]^{-2} \quad (7.20)$$

However, after comparing these results to his experimental results for the friction factor, Nikuradse suggested altering the coefficients slightly to

$$4C_f = [2.00 \log_{10}(R/k_s) + 1.74]^{-2} \quad (7.21)$$

Equation (7.21) was later rearranged slightly and written in a form with coefficients accurate to only two significant digits

$$4C_f = \left[2.0 \log_{10} \left(\frac{3.7}{k_s/(2R)} \right) \right]^{-2} \quad (7.22)$$

Equation (7.22) forms the foundation of what is understood today about fully rough flows. In fact, it is commonly used to define a surface roughness. Following the work of Schlichting [91], the equivalent sand-grain roughness for a set of data is usually defined to be the roughness that satisfies Eq. (7.22) at high roughness Reynolds numbers.

Using a similar method to the development of Eq. (7.22), Nikuradse developed an expression that matches the smooth-wall asymptote of the experimental data

$$\frac{1}{\sqrt{4C_f}} = 2.0 \log_{10} \left(R_e \sqrt{4C_f} \right) - 0.8 \quad (7.23)$$

Nikuradse plotted his experimental results over a range of roughness Reynolds numbers as a deviation from Eq. (7.22). Although he left this expression unnamed, it will be called the Nikuradse number in this work

$$\text{Nikuradse number} \equiv N_i \equiv 2.0 \log_{10} \left(\frac{3.7}{k_r} \right) - \frac{1}{\sqrt{4C_f}} \quad (7.24)$$

In the fully rough limit, the Nikuradse number is zero. However, in the limit as the roughness approaches zero, the Nikuradse number approaches a result obtained from Eq. (7.23) which can be rearranged to yield the smooth wall limit in terms of the Nikuradse number

$$N_i = 2.0 \log_{10} \left(\frac{9.294}{\sqrt{8k_s^+}} \right) \quad (7.25)$$

Figure 7.1 shows results for the Nikuradse number as a function of roughness Reynolds number for several relative roughness data sets taken by Nikuradse. The plot includes lines representing the asymptotes for the fully rough and smooth-wall limits. At low roughness Reynolds numbers, which can occur at low relative roughness values or low Reynolds numbers, the Nikuradse number approaches a linear function of the roughness Reynolds number. At high roughness Reynolds numbers, which can occur at high relative roughness values or high Reynolds numbers, the Nikuradse number is zero and is independent of the roughness Reynolds number.

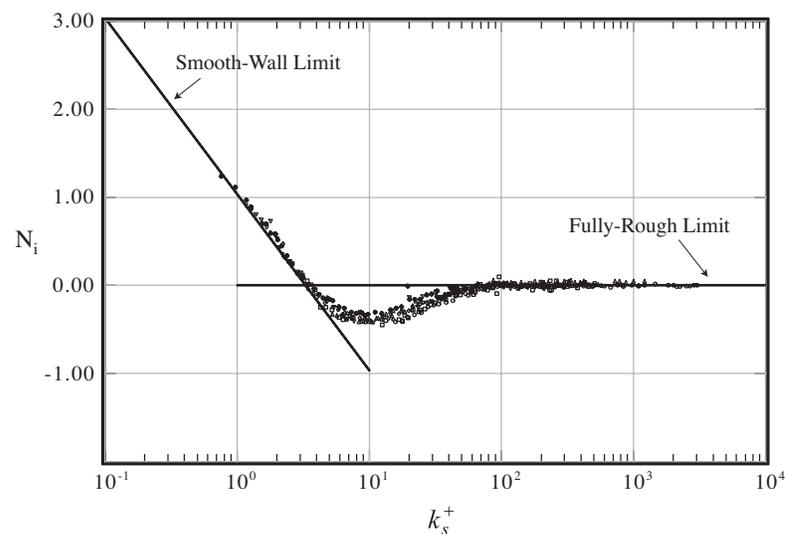


Fig. 7.1 Nikuradse number as a function of roughness Reynolds number for experimental data of Nikuradse [59].

The data of Nikuradse in conjunction with the smooth-wall and fully rough limits included in the plot suggest three significant flow regions. It may have been Schlichting [92,93] who was the first to suggest that the data of Nikuradse deviate from the smooth-wall asymptote at about $k_s^+ \approx 5.0$ and from the fully rough asymptote at about $k_s^+ \approx 70.0$. Thus, flow with roughness Reynolds numbers $k_s^+ < 5.0$ have traditionally been termed “hydraulically smooth” because roughness effects appear to be negligible in this region, and flows with roughness Reynolds numbers $k_s^+ > 70.0$ are traditionally termed “fully rough” flows because the friction factor appears to be independent of the roughness Reynolds number in this region. The region for $5.0 \leq k_s^+ \leq 70.0$ is traditionally called the transition region where the properties of the flow transition from the smooth-wall asymptote to the fully rough asymptote.

The work of Nikuradse has long been viewed as the definitive work on pipe roughness. As stated earlier, today the relative roughness values of experiments are often determined by forcing the data to match the data of Nikuradse for high roughness Reynolds numbers. The roughness of a surface is commonly reported as an equivalent sand grain roughness in the literature to suggest that it would match the work of Nikuradse for an equivalent sand grain size. For example, see the work of Shockling, Allen, and Smits [61].

D. The Colebrook Equation

The work of Colebrook [73] has also had a profound influence on how pipe flow results are now tabulated. Colebrook correlated data for many sets of experiments on commercial pipe. Because a measurement of the roughness on commercial pipe was difficult to obtain, the results were matched to Nikuradse’s results in the fully rough region, and the equivalent sand grain roughness for each experimental data set was estimated. Following Nikuradse, Colebrook plotted experimental data as a deviation from Eq. (7.22) in the same form as shown in Fig. 7.1. Figure 7.2 includes the data sets of both Nikuradse and Colebrook. Although there is considerable scatter in the commercial pipe data, Colebrook developed the empirical relation

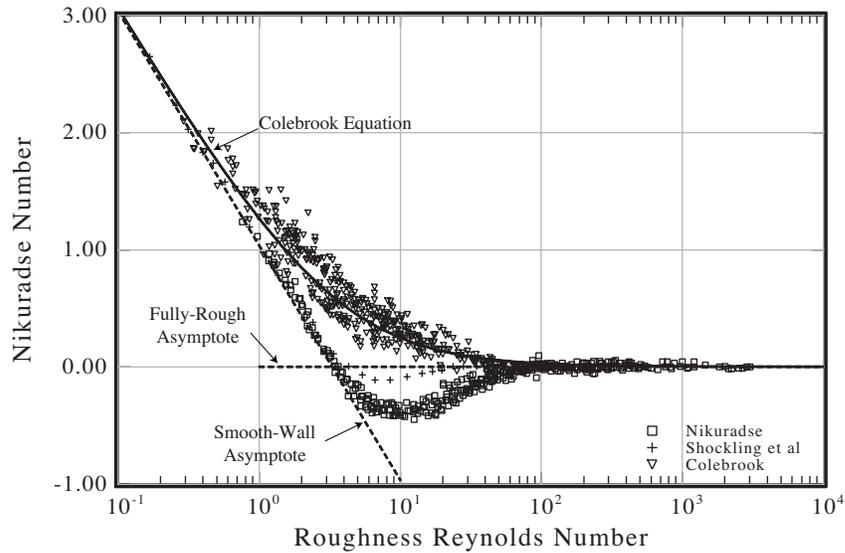


Fig. 7.2 Nikuradse number as a function of roughness Reynolds number.

$$N_i \equiv 2.0 \log_{10} \left(\frac{3.7}{k_r} \right) - \frac{1}{\sqrt{4C_f}} = 2.0 \log_{10} \left(1 + \frac{9.287}{\sqrt{8k_s^+}} \right) \quad (7.26)$$

which runs roughly through the mean of these data. Equation (7.26) will be referred to here as the Colebrook equation. Results obtained from this equation are also included in Fig. 7.2. Using Eq. (7.14), Colebrook rearranged Eq. (7.26) to express the friction factor in terms of the bulk Reynolds number

$$4C_f = \left[-2.0 \log_{10} \left(\frac{k_r}{3.7} + \frac{2.51}{Re \sqrt{4C_f}} \right) \right]^{-2} \quad (7.27)$$

which spans the smooth-wall and fully rough regions. This relationship is implicit because an iterative process is required to solve for the friction factor given the Reynolds number and relative roughness.

The fully rough limit for the Colebrook equation can be obtained by taking the limit of Eq. (7.26) as k_s^+ approaches infinity. This yields

$$N_i \lim_{k_s^+ \rightarrow \infty} = 0 \quad (7.28)$$

Likewise, the smooth-wall limit can be obtained by taking the limit of Eq. (7.26) as k_s^+ approaches zero.

This yields

$$N_i \lim_{k_s^+ \rightarrow 0} = 2.0 \log_{10} \left(\frac{9.287}{\sqrt{8k_s^+}} \right) \quad (7.29)$$

which matches the smooth-wall limit of Nikuradse given in Eq. (7.25) to three significant digits. Figure 7.2 shows the Colebrook equation in conjunction with the experimental results used to obtain the Colebrook equation. The experimental results of Nikuradse [59] and Shockling, Allen, and Smits [61] are also included as well as the fully rough and smooth-wall limits. Note the considerable scatter in the data used to generate the Colebrook equation.

Several significant characteristics of this plot should be mentioned. First, note that the data used by Colebrook does not match the data taken by Nikuradse nor that taken by Shockling et al. in the transition region between the fully rough and smooth-wall asymptotes. This reveals a fundamental difference in the characteristics of pipe roughness of the various experimental setups. In order to understand this difference, it must be understood that it is impossible to create a surface with perfectly uniform roughness, just as it is impossible to manufacture a perfectly smooth wall. Therefore, a surface roughness could more accurately be reported as having a mean relative roughness and a standard deviation in relative roughness. Those surfaces with very uniform roughness elements would be characterized as having a low standard deviation in relative roughness, and those surfaces with a wide variety of roughness element shapes or sizes would be characterized as having a high standard deviation in relative roughness. It has been argued that because the standard deviation in surface roughness for any given commercial pipe is greater than the standard deviation in sand-grain sizes used in Nikuradse's pipes, the transition from smooth-wall to fully rough flow for commercial pipes is much more gradual than the transition shown by the experimental data of Nikuradse. The Colebrook equation does in fact match the experimental data from which it was derived to within the scatter of the experimental datasets themselves, and therefore is a viable equation for flow in commercial pipes. This variation in the transition region noted between different experimental setups suggests that if the standard deviation of relative roughness is small, the transition between the smooth-wall region and fully rough regions is more abrupt than if the standard deviation of relative roughness is large. Note that the smooth-wall and fully rough asymptotes are independent of the standard deviation of relative roughness while the transition region is a function of both the mean relative roughness and the standard deviation of relative roughness.

The dependence of the transition region on the standard deviation of relative roughness is further demonstrated by the data of Shockling, Allen, and Smits [61] also included in Fig. 7.2. Shockling et al. used drawn pipe that had been honed in their experimental setup, and therefore, it can be expected that the standard deviation in relative roughness along the pipe wall was somewhere between that of commercial pipe and that of Nikuradse's pipes. This explains why the data of Shockling et al. falls between the Colebrook equation and the data taken by Nikuradse in the transition region.

Second, note that the transition region suggested by the Colebrook equation is significantly larger than the traditional region of $5.0 \leq k_s^+ \leq 70.0$ suggested by the data from Nikuradse. The data from which the Colebrook equation was obtained suggests that the transition region for commercial pipes may be more accurately defined as $0.2 \leq k_s^+ \leq 100.0$. The variation from the findings of Nikuradse in the definition of the lower limit of the transition region can also be justified on the grounds of a difference in relative roughness standard deviation for the two experimental data sets. For a given mean relative roughness, roughness effects would be more pronounced for a surface with a large standard deviation of relative roughness than for a surface with a small standard deviation of relative roughness. Therefore, roughness effects would be more pronounced at low roughness Reynolds numbers for pipes with high relative roughness standard deviations than those with low relative roughness standard deviations. Likewise, at high roughness Reynolds numbers, pipes with low relative roughness standard deviations would approach the fully rough region before those with high relative roughness standard deviations.

The results of Fig. 7.2 also show that the smooth-wall and fully rough asymptotes of the Colebrook equation match all of the experimental data sets very well. Therefore, the smooth-wall and fully rough limits of the Colebrook equation given in Eqs. (7.28) and (7.29) can be used for pipes with any given variation in roughness provided that the roughness Reynolds number is within the smooth-wall or fully rough regions. However, the Nikuradse number in the transition region predicted by the Colebrook equation should be used with caution because this region is sensitive to the standard deviation in relative roughness of the pipe.

The fact that the Colebrook equation is so widely used today is partly due to the work of Moody [94] who plotted the Colebrook equation in a form that could easily be used by engineers. In his abstract, Moody states, “*The object of this paper is to furnish the engineer with a simple means of estimating the friction factors to be used in computing the loss of head in clean new pipes and in closed conduits running full with steady flow.*” Moody used the Colebrook equation to plot the friction factor as a function of the bulk Reynolds number over a range of relative roughness values. The famous Moody chart is shown in Fig. 7.3 along with the data from Nikuradse and Shockling, et al.

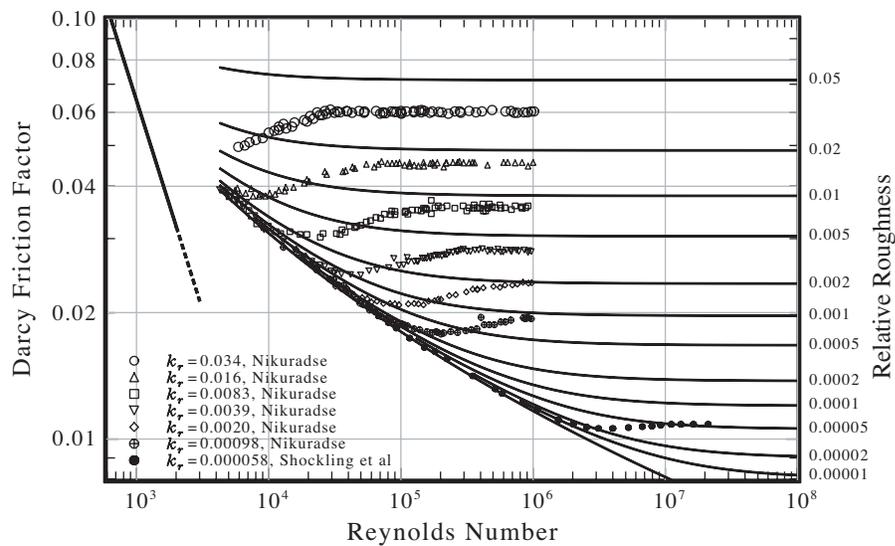


Fig. 7.3 The Moody chart with experimental data from Nikuradse and Shockling, et al.

This figure without the experimental data can be found in many fluid mechanics text books and is perhaps the most widely used engineering chart in fluid mechanics. The chart allows the engineer to estimate the head losses of a pipe if an estimate for the surface roughness and bulk flow rate are known. Note that at very low Reynolds numbers, the flow becomes laminar and follows the well-known and easily verified relation

$$4C_f = \frac{64}{R_e} \quad (7.30)$$

However, the transition between laminar and turbulent flow is difficult to predict, and remains a vague region of the Moody chart. This transition region is denoted by a dotted line in the figure.

It has been shown that the most widely accepted correlations for pipe flow are all fundamentally based on the work of Nikuradse. Further, it has been shown that Nikuradse's work on the fully rough limit forms the basis for determining the roughness of a given surface. Because the fully rough relation developed by Nikuradse and included here as Eq. (7.22) forms the foundation for rough-wall flow measurements, any turbulence model claiming to be capable of modeling surface roughness must match this relation between the friction factor and surface roughness in the limit as the flow becomes fully rough.

E. Velocity Profile

A wealth of experimental data has shown that the velocity profile follows the law of the wall in the near-wall region. An empirical correlation from Nikuradse's data that includes roughness effects is

$$u^+ = 2.5 \ln \left(\frac{y^+}{k_s^+} \right) + 8.5 \quad (7.31)$$

In the near-wall region of fully rough flows, an analytical approximation for the velocity profile can be developed using mixing-length theory. Using Eq. (7.18) in Eq. (7.19) and integrating subject to the no-slip boundary condition gives the near-wall fully rough relation

$$u^+ = \frac{1}{\kappa} \ln \left(\frac{y^+}{\gamma k_s^+} + 1 \right) \quad (7.32)$$

Figure 7.4 shows the relations given in Eqs. (7.31) and (7.32) along with the experimental velocity profile data of Nikuradse. Note that the analytical approximation given in Eq. (7.32) only deviates from the empirical relation given in Eq. (7.31) in the near wall region where the wall coordinate is on the same order as the roughness. Because this region of the flow is so minute compared to the bulk flow, the integral of Eq. (7.32) yields a value very close to the bulk velocity. This is the approach taken by Nikuradse in the development of his relation between bulk Reynolds number, friction factor, and roughness. Also note from the figure that the experimental data matches the relation given in Eq. (7.32) very well over the entire region of the flow. Thus, although Eq. (7.32) was derived using the near-wall approximations from mixing-length theory, it can be used across the entire flow, and lacks only in its ability to satisfy the centerline symmetry boundary condition.

The existence of the law of the wall for turbulent flows has been substantiated by many sets of data. Therefore, any viable turbulence model should be capable of predicting this flow profile for rough flows.

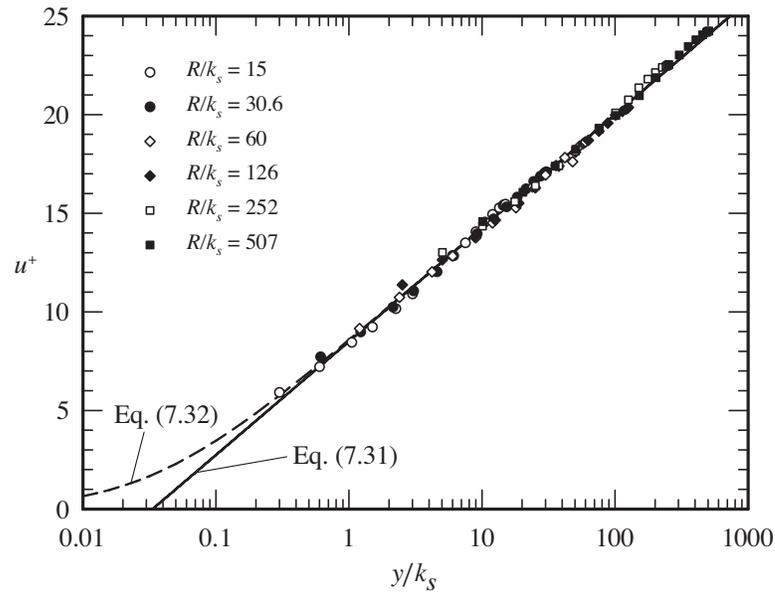


Fig. 7.4 Velocity profiles in rough pipes at high Reynolds numbers.

F. Turbulent Eddy Viscosity

Equation (7.9) is the Boussinesq-based RANS equation for fully developed flow in a pipe, and holds independent of any turbulence model. It can be rearranged to yield

$$\frac{du^+}{d\hat{r}} = -\frac{\hat{r}}{(\hat{\nu} + \hat{\nu}_t)}, \quad u^+(1) = 0 \quad (7.33)$$

where $\hat{\nu} \equiv \nu/(u_\tau R)$, $\hat{\nu}_t \equiv \nu_t/(u_\tau R)$, and $\hat{r} \equiv r/R$. The goal of any RANS-based turbulence model is to correctly model the turbulent eddy viscosity such that Eq. (7.33) can be integrated to yield a velocity profile that matches experimental data. Therefore, Eq. (7.33) can also be seen as a definition for the turbulent eddy viscosity if a velocity profile is known. Given a velocity profile, the eddy viscosity as a function of radius can be calculated from

$$\hat{\nu}_t = -\frac{\hat{r}}{du^+/d\hat{r}} - \hat{\nu} \quad (7.34)$$

In the core region of the pipe where the velocity gradient is small, this method for estimating the turbulent eddy viscosity becomes very sensitive to small errors in the velocity gradient measurements. As the axis of the pipe is approached, both \hat{r} and $du^+/d\hat{r}$ approach zero, and Eq. (7.34) is indeterminate. Applying L'Hospital's rule, it can be shown that at the pipe centerline, the eddy viscosity is related to the second derivative of the velocity profile according to

$$\hat{\nu}_t \Big|_{\lim \hat{r} \rightarrow 0} = -\frac{1}{d^2 u^+ / d\hat{r}^2}$$

Therefore, accurate estimates for the eddy viscosity at the center of a pipe are dependent on accurate estimates for the second derivative of the velocity profile in that region. Because these measurements are difficult to obtain near the pipe centerline, estimates for the eddy viscosity are more accurate near the pipe wall than near the pipe centerline.

It can be shown that mixing-length theory predicts a turbulent eddy viscosity for fully rough pipe flow of

$$\hat{\nu}_t = \kappa [C_{\ell 0} + 2\gamma k_r - (2C_{\ell 0} - 0.5)\hat{r}^2 - (0.5 - C_{\ell 0})\hat{r}^4] \hat{r}^{1/2} \quad (7.35)$$

where $C_{\ell 0}$ is about 0.345. This matches the experimental data of Nikuradse [59] very well over the entire pipe cross section. Figure 7.5 shows the experimental results of Nikuradse as well as the analytical relation from mixing-length theory. Note that mixing-length theory predicts a turbulent eddy viscosity of identically zero at the pipe centerline. This comes as a direct result of the model assumption that the eddy viscosity is directly proportional to the velocity gradient as can be seen in Eq. (7.16). The symmetry boundary condition requires that the change in eddy viscosity with respect to radius be zero at the center of the pipe. In contrast, Eq. (7.35) has an infinite gradient at the pipe centerline. Therefore, this relation cannot be used in the center of the pipe.

Reichardt [95] correlated sets of experimental data and suggested a simple relation that could be used to model the eddy-viscosity distribution for a pipe with smooth walls

$$\hat{\nu}_t = \frac{\kappa}{6} (1 + \hat{r}^2 - 2\hat{r}^4)$$

Applying Nikuradse's near-wall results, this relation can be modified to account for roughness effects

$$\hat{\nu}_t = \frac{\kappa}{6}(1 - \hat{r} + 2\gamma k_r)(1 + \hat{r})(1 + 2\hat{r}^2) \quad (7.36)$$

This relation along with Reichardt's eddy-viscosity data is also included in Fig. 7.5.

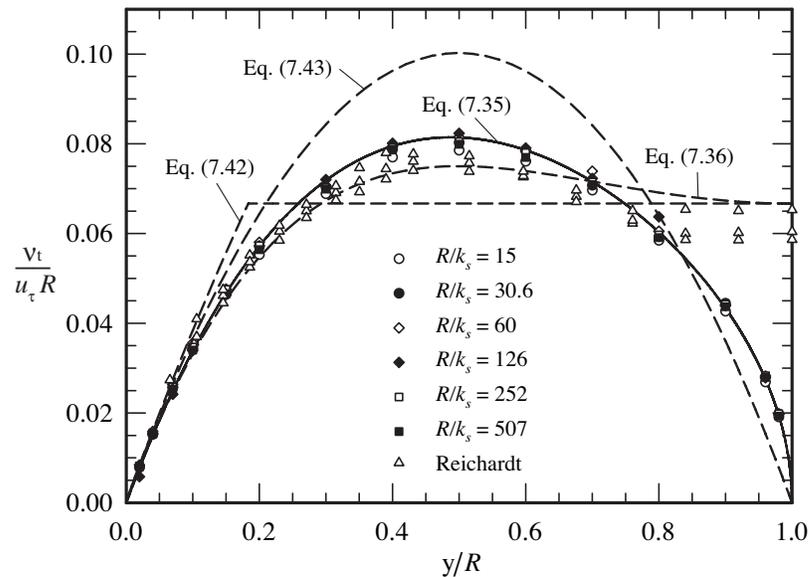


Fig. 7.5 Eddy-viscosity profiles in fully developed pipe flow.

Note that Eq. (7.36) stays fairly constant over the core of the pipe. Based on this observation, Kays and Crawford [96] suggest using “a simpler and adequate alternative” which uses a constant value of $\hat{\nu}_t = \kappa/6$ over the central region of the pipe combined with an approximation from mixing-length theory near the wall. A similar formulation that applies to rough walls can be developed as follows.

Near the wall of fully rough flow, the mixing length is a function of the distance from the wall and the wall roughness

$$\ell^+ = \kappa(y^+ + \gamma k_s^+) = \kappa[R_\tau(1 - \hat{r}) + \gamma k_s^+] \quad (7.37)$$

Using this in Eq. (7.16) gives

$$\nu^+ = \kappa^2(y^+ + \gamma k_s^+)^2 \left| \frac{du^+}{dy^+} \right| = \frac{\kappa^2 [R_\tau(1 - \hat{r}) + \gamma k_s^+]^2}{R_\tau} \left| \frac{du^+}{d\hat{r}} \right| \quad (7.38)$$

Combining this with Eq. (7.9) and solving for ν^+ gives an expression for the eddy viscosity near the wall

$$v^+ = \sqrt{\frac{1}{4} + F^2 \hat{r}} - \frac{1}{2}, \quad F = \kappa[R_\tau(1 - \hat{r}) + \gamma k_s^+] \quad (7.39)$$

Although the experimental data in the fully rough limit is limited, assuming that the eddy viscosity is a constant in the central region of the pipe yields the expression

$$v^+ = \begin{cases} (\kappa/6)R_\tau, & 0 \leq \hat{r} \leq \hat{r}_o \\ \sqrt{\frac{1}{4} + F^2 \hat{r}} - \frac{1}{2}, & \hat{r}_o \leq \hat{r} \leq 1 \end{cases}, \quad F = \kappa[R_\tau(1 - \hat{r}) + \gamma k_s^+] \quad (7.40)$$

where \hat{r}_o is the outer root of

$$\sqrt{\frac{1}{4} + F^2 \hat{r}} - \frac{1}{2} = (\kappa/6)R_\tau, \quad F = \kappa[R_\tau(1 - \hat{r}) + \gamma k_s^+] \quad (7.41)$$

Equation (7.40) can alternately be written as

$$\hat{v}_t = \begin{cases} \kappa/6, & 0 \leq \hat{r} \leq \hat{r}_o \\ \frac{1}{R_\tau} \sqrt{\frac{1}{4} + F^2 \hat{r}} - \frac{1}{2R_\tau}, & \hat{r}_o \leq \hat{r} \leq 1 \end{cases}, \quad F = \kappa[R_\tau(1 - \hat{r}) + \gamma k_s^+] \quad (7.42)$$

The relation given in Eq. (7.42) provides a simple approximation for the turbulent-eddy-viscosity profile and is also shown in Fig. 7.5. Values of $\kappa = 0.404$, $\gamma = 0.0341$, and $A^+ = 27$ give results that match experimental data very well.

One additional method that can be used to estimate the eddy-viscosity profile is through the use of the empirical correlation to Nikuradse's velocity profile data. Differentiating Eq. (7.32), using it in Eq. (7.33), and assuming $\hat{v} \ll \hat{v}_t$ gives the fully rough turbulent eddy viscosity that will match the velocity profile of Nikuradse's data

$$\hat{v}_t = \kappa \hat{r}^2 (1 - \hat{r} + 2\gamma k_r) \quad (7.43)$$

This relation is also included in Fig. 7.5 for comparison. Any eddy-viscosity distribution that matches this relation will yield the velocity profile given in Eq. (7.32) for high roughness Reynolds numbers.

Equations (7.35), (7.36), (7.42), and (7.43) give relations for what could be termed eddy-viscosity models for fully rough pipe flow. These can be used directly in Eq. (7.33) and integrated to yield a velocity profile, bulk velocity, and friction factor. The velocity profiles that results from these models will be presented in Chapter 9 for comparison with the results of the energy-vorticity model in consideration.

Because estimates for the eddy viscosity are strong functions of the velocity gradient and measurements for the velocity gradient near the pipe centerline are difficult to accurately obtain, the exact behavior of the eddy viscosity near the central region of fully rough pipe flow is not well established. Equations (7.35), (7.36), (7.42), and (7.43) provide four approximations for the eddy viscosity that have been presented in the literature. The approximations based on the findings of Nikuradse given in Eqs. (7.35) and (7.43) are not viable in the central region because it is impossible for the eddy viscosity to go to zero at the centerline, and for the derivative of the eddy viscosity to have a finite slope at the centerline. The work of Kays and Crawford presented in Eq. (7.42) is used as a reference in the subsequent development of the Phillips k - λ model. This was a viable option because it provides a model for the viscosity which matches most closely the assumptions that were inherent in the development of the velocity profile given in Eq. (7.32). Because Eq. (7.32) matches experimental data so well, it was expected that an eddy-viscosity profile similar to that given in Eq. (7.42) would yield results for the velocity profile and friction factor very near experimental data. However, the work of Reichardt presented in Eq. (7.36) may also be worth consideration in a future edition of a fully rough turbulence model.

Before leaving the topic of eddy viscosity, it is insightful to consider one additional aspect of mixing-length theory. Recall that a fundamental approximation for fully rough flow is that the molecular viscosity is negligible compared to the turbulent eddy viscosity. The mixing length near a wall is very accurate and is expressed in Eq. (7.37). Using Eq. (7.37) and (7.19) in Eq. (7.16) gives the ratio of turbulent eddy viscosity to molecular viscosity at the wall

$$\nu^+ \Big|_{y^+=0} \equiv \frac{V_t}{\nu} \Big|_{y=0} = \kappa(y^+ + \gamma k_s^+)$$

For fully rough flow, this ratio at the wall must be much greater than unity. Using the suggested values of $\kappa = 0.404$ and $\gamma = 0.0341$ for the accepted value of fully rough flow $k_s^+ = 70.0$, this yields $\nu^+ = 0.96$ at the wall. This result shows that at $k_s^+ = 70.0$, the molecular viscosity is about the same order of magnitude as the turbulent eddy viscosity at the wall. Using $k_s^+ = 100.0$ gives $\nu^+ = 1.4$. Based on this mixing-length

approximation, it is suggested that a better lower limit for fully rough pipe flow is about $k_s^+ = 1000$ which yields a viscosity ratio of $\nu^+ = 14$ at the wall.

III. The Phillips k - λ Model for Fully Rough Pipe Flow

The foundational equations for the Phillips k - λ model for steady, incompressible, fully rough pipe flow include the continuity equation

$$\nabla \cdot \bar{\mathbf{V}} = 0 \quad (7.44)$$

The Boussinesq-RANS equations

$$(\bar{\mathbf{V}} \cdot \nabla) \bar{\mathbf{V}} = -\frac{1}{\rho} \nabla(\hat{p}) + \nabla \cdot [2(\nu + \nu_t) \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})] \quad (7.45)$$

an algebraic relation for the eddy viscosity

$$\nu_t = \lambda \sqrt{k} \quad (7.46)$$

and the turbulent-kinetic-energy equation in terms of the mean vortex wavelength

$$\begin{aligned} (\bar{\mathbf{V}} \cdot \nabla) k &= 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \nu(C_\lambda k/\lambda^2 + 4\nabla \cdot \{\frac{1}{3} \nabla k - \nabla \cdot [\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})]\}) \\ &+ \nabla \cdot ((\nu + \nu_t/\sigma_k) \{\frac{5}{3} \nabla k - 2\nabla \cdot [\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}})]\}) \end{aligned} \quad (7.47)$$

Several possible closing equations for the mean vortex wavelength are included in Chapter 2. In general, the closing relation for the mean vortex wavelength could take the form of a transport equation similar to the turbulent-kinetic-energy equation. However, for fully rough pipe flow, it has been found that an algebraic relation may suffice based on the results of mixing-length theory mentioned in the previous section. An initial estimate for the mean vortex wavelength distribution was developed for fully rough pipe flow by combining the algebraic relation for the eddy viscosity given in Eq. (7.42) with the model equations given in Eqs. (7.44) – (7.47). This provided a complete model that could be solved to yield the mean vortex wavelength distribution given a relative roughness and roughness Reynolds number. After correlating several mean vortex wavelength distributions over a range of Reynolds numbers and relative roughness values, a general algebraic relation for the mean vortex wavelength was developed. It was found

that the algebraic relation could be a function of a bulk flow parameter that will be termed the core Reynolds number.

A. Core Reynolds Number

The velocity profile in the pipe is a function of the pressure gradient and eddy viscosity and can be written as

$$(v + \nu_t) \frac{d\bar{V}_z}{dr} = \frac{1}{2\rho} \frac{d\hat{p}}{dz} r \quad (7.48)$$

It has been shown that experimental data suggests that the eddy viscosity in the central region of the pipe is nearly constant. Assuming that the eddy viscosity is constant near the centerline of the pipe, Eq. (7.48) can be integrated from the pipe centerline to an arbitrary point in the constant-eddy-viscosity core

$$\bar{V}_z|_{r=r} = \bar{V}_z|_{r=0} + \frac{1}{4\rho(v + \nu_t)} \frac{d\hat{p}}{dz} r^2 \quad (7.49)$$

Note that the velocity profile in this central region of the pipe is parabolic. Dividing this expression by the velocity at the centerline, $\bar{V}_{\max} \equiv \bar{V}_z(0)$ gives the nondimensional equation

$$\frac{\bar{V}_z(r)}{\bar{V}_{\max}} = 1 + \frac{1}{4\rho(v + \nu_t)\bar{V}_{\max}} \frac{d\hat{p}}{dz} r^2 \quad (7.50)$$

The pressure gradient is constant and negative in fully developed flow. Therefore, Eq. (7.50) yields an important length scale associated with the core region of the flow

$$R_c \equiv \sqrt{-4\rho(v + \nu_t)\bar{V}_{\max} / \frac{d\hat{p}}{dz}} = R\sqrt{2(\hat{\nu} + \hat{\nu}_t)u_{\max}^+} \quad (7.51)$$

This will be called the core radius of the flow. The centerline velocity and the core radius are the important velocity and length scales in the core region of the pipe. These can be used to form what will be referred to here as the core Reynolds number

$$R_{ec} \equiv \frac{2\bar{V}_{\max}R_c}{\nu} \quad (7.52)$$

The turbulent properties in the core region of the pipe should also depend on the velocity and length scales used to develop the core Reynolds number. Therefore, the turbulent kinetic energy and mean vortex wavelength at the center of the pipe should be a function of the core Reynolds number.

B. Mean Vortex Wavelength Distribution

At the wall of a fully rough flow, the turbulent eddy viscosity can be evaluated from Eq. (7.42). Applying Eq. (7.12) gives

$$\left. \frac{v_t}{u_\tau R} \right|_{r=R} = \sqrt{(k_r/k_s^+)^2 + (2\kappa\gamma k_r)^2} - (k_r/k_s^+) \quad (7.53)$$

which can alternately be written as

$$\left. \frac{v_t}{u_\tau k_s} \right|_{r=R} = \sqrt{1/(2k_s^+)^2 + (\kappa\gamma)^2} - 1/(2k_s^+) \quad (7.54)$$

Therefore, the turbulent eddy viscosity at the wall is independent of the pipe radius. Also from Eq. (7.42), the turbulent eddy viscosity at the center of the pipe is assumed independent of the wall effects

$$\left. \frac{v_t}{u_\tau R} \right|_{r=0} = \kappa/6 \quad (7.55)$$

Note that the eddy viscosity is proportional to $u_\tau k_s$ at the wall and $u_\tau R$ at the centerline.

In general we can expect the mean vortex wavelength to be a function of both the core Reynolds number and the roughness Reynolds number. Results from this model compared to experimental results suggest that the mean vortex wavelength at the centerline is only a function of the core Reynolds number and the pipe radius

$$\lambda \Big|_{r=0} = C_{r1} R_{ec}^{a_{r1}} R \quad (7.56)$$

where C_{r1} and a_{r1} are model constants. At the wall, experimental data and results from this model suggest that the mean vortex wavelength is only a function of the surface roughness and roughness Reynolds number

$$\lambda \Big|_{r=R} = C_{r2} k_s^{+a_{r2}} k_s \quad (7.57)$$

where C_{r2} and a_{r2} are model constants. In the transition region, agreement with experimental data can be obtained by using the transition function

$$\lambda = \lambda|_{r=R} + C_{r3} R_{ec}^{a_{r3}} (R-r)^{a_{r4}} \quad (7.58)$$

where C_{r3} , a_{r3} , and a_{r4} are model constants. Combining Eqs. (7.56) – (7.58) gives the general algebraic relation for the mean vortex wavelength

$$\lambda = \min[C_{r1} R_{ec}^{a_{r1}} R, C_{r2} k_s^{+a_{r2}} k_s + C_{r3} R_{ec}^{a_{r3}} (1-r/R)^{a_{r4}} R] \quad (7.59)$$

where C_{r1} , a_{r1} , C_{r2} , a_{r2} , C_{r3} , a_{r3} , and a_{r4} are closure coefficients. Equation (7.59) forms the final equation for the Phillips k - λ model for fully rough pipe flow.

Combining Eqs. (7.46), (7.53), and (7.59) and solving for the value of the turbulent kinetic energy at the wall gives the fully rough-wall boundary condition on k

$$\frac{k}{u_\tau^2} \Big|_{r=R} = \left[\frac{\sqrt{(1/k_s^+)^2 + (2\gamma\kappa)^2} - 1/k_s^+}{2C_{r2} k_s^{+a_{r2}}} \right]^2 \quad (7.60)$$

which applies in the limit as k_s^+ approaches infinity. Note that the wall boundary condition on k is independent of the pipe radius.

C. Model Summary

The Phillips k - λ model for fully rough pipe flow in dimensional form comprises the following equations: the Boussinesq-RANS equation including the no-slip wall boundary condition

$$\frac{d\bar{V}_z}{dr} = -\frac{u_\tau^2}{(\nu + \nu_t)} \frac{r}{R}, \quad \bar{V}_z \Big|_{r=R} = 0 \quad (7.61)$$

the turbulent-kinetic-energy transport equation and associated boundary conditions

$$\begin{aligned} -\frac{5}{3r} \frac{d}{dr} \left[(\nu + \nu_t / \sigma_k) r \frac{dk}{dr} \right] &= \nu_t \left(\frac{\partial \bar{V}_z}{\partial r} \right)^2 - \nu \left[C_\lambda \frac{k}{\lambda^2} + \frac{4}{3r} \frac{d}{dr} \left(r \frac{dk}{dr} \right) \right] \\ \frac{dk}{dr} \Big|_{r=0} &= 0, \quad k \Big|_{r=R} = \left[\frac{\sqrt{(1/k_s^+)^2 + (2\gamma\kappa)^2} - 1/k_s^+}{2C_{r2} k_s^{+a_{r2}}} \right]^2 u_\tau^2 \end{aligned} \quad (7.62)$$

an algebraic expression for the mean vortex wavelength

$$\lambda = \min[C_{r1} R_{ec}^{a_{r1}} R, C_{r2} k_s^{+a_{r2}} k_s + C_{r3} R_{ec}^{a_{r3}} (1-r/R)^{a_{r4}} R] \quad (7.63)$$

and an algebraic expression for the turbulent eddy viscosity

$$\nu_t = \lambda \sqrt{k} \quad (7.64)$$

with definitions for the core Reynolds number

$$R_{ec} = \frac{\bar{V}_z(0)}{\nu} 2R \sqrt{2 \frac{\nu + \nu_t(0)}{u_\tau R} \frac{\bar{V}_z(0)}{u_\tau}} \quad (7.65)$$

and the roughness Reynolds number

$$k_s^+ \equiv u_\tau k_s / \nu \quad (7.66)$$

Applying the nondimensional definitions

$$\begin{aligned} \hat{r} &\equiv \frac{r}{R}, \quad R_\tau \equiv \frac{u_\tau R}{\nu}, \quad u^+ \equiv \frac{\bar{V}_z}{u_\tau}, \quad k_s^+ \equiv \frac{u_\tau k_s}{\nu} \\ k^+ &\equiv \frac{k}{u_\tau^2}, \quad \lambda^+ \equiv \frac{u_\tau \lambda}{\nu}, \quad \nu^+ \equiv \frac{\nu_t}{\nu} \end{aligned} \quad (7.67)$$

this model with the associated boundary conditions can be written in nondimensional form in terms of the independent variable \hat{r} as

$$\begin{aligned} \frac{du^+}{d\hat{r}} &= -\frac{R_\tau \hat{r}}{(1+\nu^+)} \\ -\frac{5}{3\hat{r}} \frac{d}{d\hat{r}} \left[(1+\nu^+/\sigma_k) \hat{r} \frac{dk^+}{d\hat{r}} \right] &= \nu^+ \left(\frac{\partial u^+}{\partial \hat{r}} \right)^2 - \left[R_\tau^2 C_\lambda \frac{k^+}{\lambda^{+2}} + \frac{4}{3\hat{r}} \frac{d}{d\hat{r}} \left(\hat{r} \frac{dk^+}{d\hat{r}} \right) \right] \\ \lambda^+ &= \min[C_{r1} R_{ec}^{a_{r1}} R_\tau, C_{r2} k_s^{+(1+a_{r2})} + C_{r3} R_{ec}^{a_{r3}} (1-\hat{r})^{a_{r4}} R_\tau] \\ \nu^+ &= \lambda^+ \sqrt{k^+} \\ R_{ec} &= [2u^+(0)]^{3/2} \{R_\tau [1+\nu^+(0)]\}^{1/2} \end{aligned} \quad (7.68)$$

with boundary conditions

$$u^+ \Big|_{\hat{r}=1} = 0, \quad \frac{dk^+}{d\hat{r}} \Big|_{\hat{r}=0} = 0, \quad k^+ \Big|_{\hat{r}=1} = \left[\frac{\sqrt{(1/k_s^+)^2 + (2\gamma\kappa)^2} - 1/k_s^+}{2C_{r2} k_s^{+a_{r2}}} \right]^2 \quad (7.69)$$

where the model constants σ_k , C_λ , C_{r1} , a_{r1} , C_{r2} , a_{r2} , C_{r3} , a_{r3} , and a_{r4} are as of yet unknown. Note that $\hat{r} = 0$ is the value at the centerline and $\hat{r} = 1$ is the value at the wall.

D. Solution Process

A Fortran code to solve this model is included in Appendix O. The solution process is developed as follows. Combining like terms, the k -transport equation in the model can be rearranged to yield

$$-\frac{d}{d\hat{r}} \left[\left(\frac{1}{3} + \frac{5\nu^+}{3\sigma_k} \right) \hat{r} \frac{dk^+}{d\hat{r}} \right] = \nu^+ \left(\frac{\partial u^+}{\partial \hat{r}} \right)^2 \hat{r} - R_\tau^2 C_\lambda \frac{k^+}{\lambda^{+2}} \hat{r} \quad (7.70)$$

Defining

$$Q^+ \equiv - \left(\frac{1}{3} + \frac{5\nu^+}{3\sigma_k} \right) \hat{r} \frac{dk^+}{d\hat{r}} \quad (7.71)$$

and using this definition along with the Boussinesq-RANS relation for the velocity gradient in Eq. (7.70), the second-order k -transport equation can be written as two first order transport equations with its associated boundary conditions

$$\begin{aligned} \frac{dk^+}{d\hat{r}} &= - \frac{Q^+}{\left(\frac{1}{3} + \frac{5\nu^+}{3\sigma_k} \right) \hat{r}}, \quad k^+ \Big|_{\hat{r}=1} = \left[\frac{\sqrt{(1/k_s^+)^2 + (2\gamma\kappa)^2} - 1/k_s^+}{2C_{r2}k_s^{+a_{r2}}} \right]^2 \\ \frac{dQ^+}{d\hat{r}} &= \frac{\nu^+ R_\tau^2}{(1+\nu^+)^2} \hat{r}^3 - \frac{C_\lambda R_\tau^2 k^+}{\lambda^{+2}} \hat{r}, \quad Q^+ \Big|_{\hat{r}=0} = 0 \end{aligned} \quad (7.72)$$

This gives a version of the model that can be solved using a direct numerical integration scheme such as the fourth-order Runge-Kutta method and comprises the set of algebraic and first-order equations with the appropriate boundary conditions

$$\begin{aligned} \frac{du^+}{d\hat{r}} &= - \frac{R_\tau \hat{r}}{(1+\nu^+)}, \quad u^+ \Big|_{\hat{r}=1} = 0 \\ \frac{dk^+}{d\hat{r}} &= - \frac{Q^+}{\left(\frac{1}{3} + \frac{5\nu^+}{3\sigma_k} \right) \hat{r}}, \quad k^+ \Big|_{\hat{r}=1} = \left[\frac{\sqrt{(1/k_s^+)^2 + (2\gamma\kappa)^2} - 1/k_s^+}{2C_{r2}k_s^{+a_{r2}}} \right]^2 \\ \frac{dQ^+}{d\hat{r}} &= \frac{\nu^+ R_\tau^2}{(1+\nu^+)^2} \hat{r}^3 - \frac{C_\lambda R_\tau^2 k^+}{\lambda^{+2}} \hat{r}, \quad Q^+ \Big|_{\hat{r}=0} = 0 \\ \lambda^+ &= \min[C_{r1}R_{ec}^{a_{r1}}R_\tau, C_{r2}k_s^{+(1+a_{r2})} + C_{r3}R_{ec}^{a_{r3}}(1-\hat{r})^{a_{r4}}R_\tau] \\ \nu^+ &= \lambda^+ \sqrt{k^+} \\ R_{ec} &= [2u^+(0)]^{3/2} \{R_\tau[1+\nu^+(0)]\}^{1/2} \end{aligned} \quad (7.73)$$

Note that the expression for the first derivative of k is singular at the centerline where $\hat{r} = 0$. In this region, the turbulent eddy viscosity can be assumed constant, and the Taylor series expansion for k^+ gives

$$k^+(\hat{r}) = k^+(0) + A_k k^{+2}(0) \hat{r}^2 - \frac{B_k - A_k k^{+3}(0)}{2} \hat{r}^4 - \frac{A_k k^+(0) [B_k - 2A_k^2 k^{+3}(0)]}{9} \hat{r}^6 + O(\hat{r}^8) + \dots \quad (7.74)$$

where

$$A_k \equiv \frac{3\sigma_k C_\lambda R_\tau^2}{4(\sigma_k + 5\nu^+) \nu^{+2}}, \quad B_k \equiv \frac{3\sigma_k \nu^+ R_\tau^2}{8(\sigma_k + 5\nu^+) (1 + \nu^+)^2} \quad (7.75)$$

Thus, the first derivative of k^+ can be approximated at the centerline as

$$\frac{dk^+}{d\hat{r}}(0) = 2A_k k^{+2}(0) \hat{r} - 2[B_k - A_k k^{+3}(0)] \hat{r}^3 - 2 \frac{A_k k^+(0) [B_k - 2A_k^2 k^{+3}(0)]}{3} \hat{r}^5 + O(\hat{r}^7) + \dots \quad (7.76)$$

In the case that $\nu^+ = 0$, Eq. (7.76) can be conditionally replaced with $dk^+/d\hat{r}(0) = 0$. Note that in order to use Eq. (7.76), estimates for k^+ and ν^+ at the centerline must be known. Additionally, the mean vortex wavelength is a function of the core Reynolds number, which is not known until a solution is obtained. For this purpose, initial estimates for k^+ and ν^+ at the centerline and the core Reynolds number can be obtained from

$$\begin{aligned} k^+(0) &\cong 0.205 R_\tau^{0.158} \\ \nu^+(0) &\cong C_{\ell 2} R_\tau \\ R_{cc} &\cong 21.26 R_\tau^{1.078} \end{aligned} \quad (7.77)$$

where the shear Reynolds number can be evaluated from the known case parameters

$$R_\tau = \frac{k_s^+}{2k_r} \quad (7.78)$$

Because two of the three boundary conditions for the model are specified at the wall, and one is specified at the centerline, a shooting method is used in conjunction with an integration scheme to obtain the solution to the system of equations. For such an approach, an initial estimate for Q^+ at the wall is needed. A useful initial estimate is

$$Q^+(1) \cong 9.4 k_r^{-0.152} \quad (7.79)$$

The solution procedure implemented in this work is as follows. Given case values for k_r and k_s^+ along with model constants, R_τ is evaluated from Eq. (7.78). Using the estimates given in Eqs. (7.77) and (7.79), a fourth-order Runge-Kutta integration scheme is used to integrate the first-order model equations from the wall ($\hat{r}=1$) to the centerline ($\hat{r}=0$). Near the centerline, ie. $\hat{r} < 0.002$, Eq. (7.76) is used instead of the model equation for k^+ given in Eq. (7.73). Once the integration is complete, estimates for the centerline values for Q^+ , k^+ , ν^+ , and the core Reynolds number are updated. This integration process is repeated using a secant method until the centerline boundary condition $Q^+(0)=0$ is adequately satisfied and the centerline values for k^+ and the core Reynolds number have stopped changing. It should be noted that a negative solution to any one of the case variables is non-physical, and a constraint to that end can also be applied during the solution process.

E. Initial Results

Initial estimates for the closure coefficients that match the velocity and eddy-viscosity profiles of mixing length theory are

$$\begin{aligned} \sigma_k = 0.01, \quad C_\lambda = 0.0004, \quad C_{r1} = 0.185, \quad a_{r1} = -0.0735, \quad C_{r2} = 0.0383, \\ a_{r2} = -0.1039, \quad C_{r3} = 1.072, \quad a_{r3} = -0.0753, \quad a_{r4} = 1.0 \end{aligned} \quad (7.80)$$

Figure 7.6 shows the velocity profiles for $k_s^+ = 1000$ over the range of relative roughness values reported by Nikuradse and Figure 7.7 shows the corresponding eddy-viscosity profiles. Note the velocity profiles resemble the relations given in Eqs. (7.31) and (7.32). Additionally, the eddy-viscosity distributions over this range closely resemble the relation give in Eq. (7.42). Even though the velocity and turbulent-eddy-viscosity profiles have the correct trends, the friction factors for high roughness Reynolds numbers do not consistently match the Colebrook equation. This can be seen in Fig. 7.8 which shows the results of the model using these coefficients over a range of relative roughness values and Reynolds numbers. If the model were correct, the black dots would fall directly on the black lines. Note that the model results are significantly lower than the results predicted by the Colebrook equation, and that the model predictions get worse at lower relative roughness values. The experimental data of Nikuradse and Shockling et al. is included in the plot for comparison.

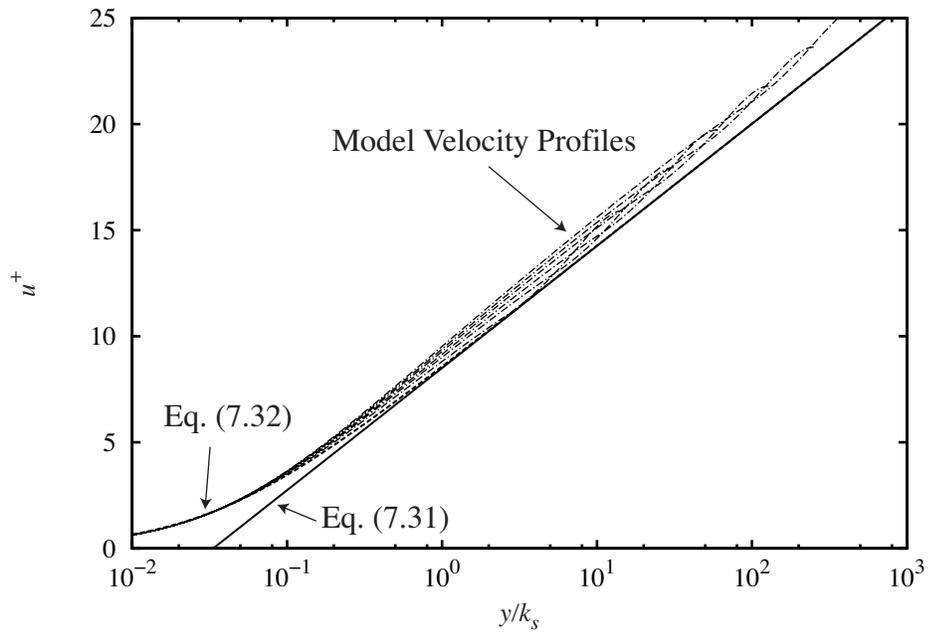


Fig. 7.6 Initial velocity profile results for the Phillips k - λ model.

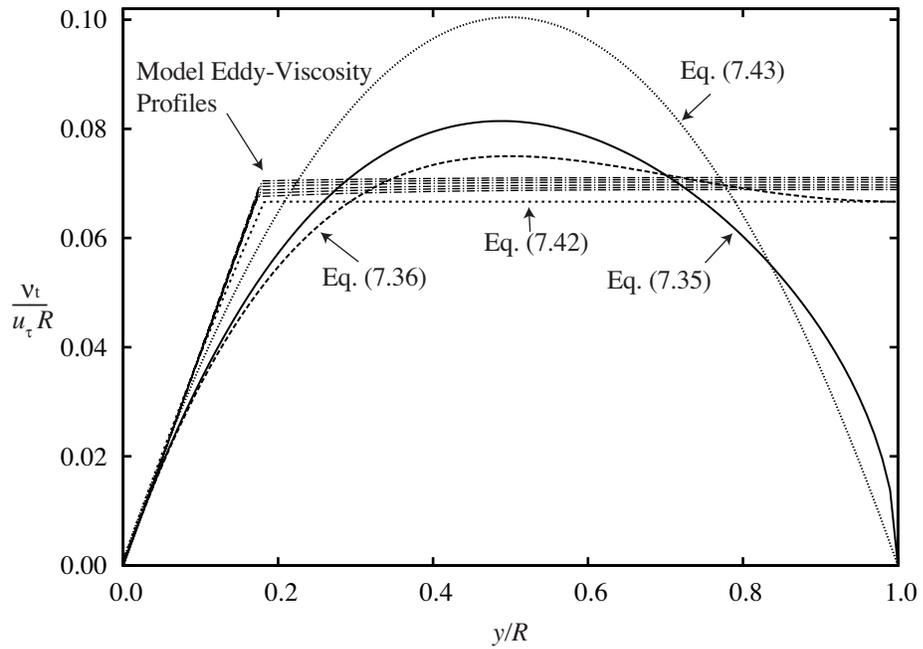


Fig. 7.7 Initial eddy-viscosity profile results for the Phillips k - λ model.

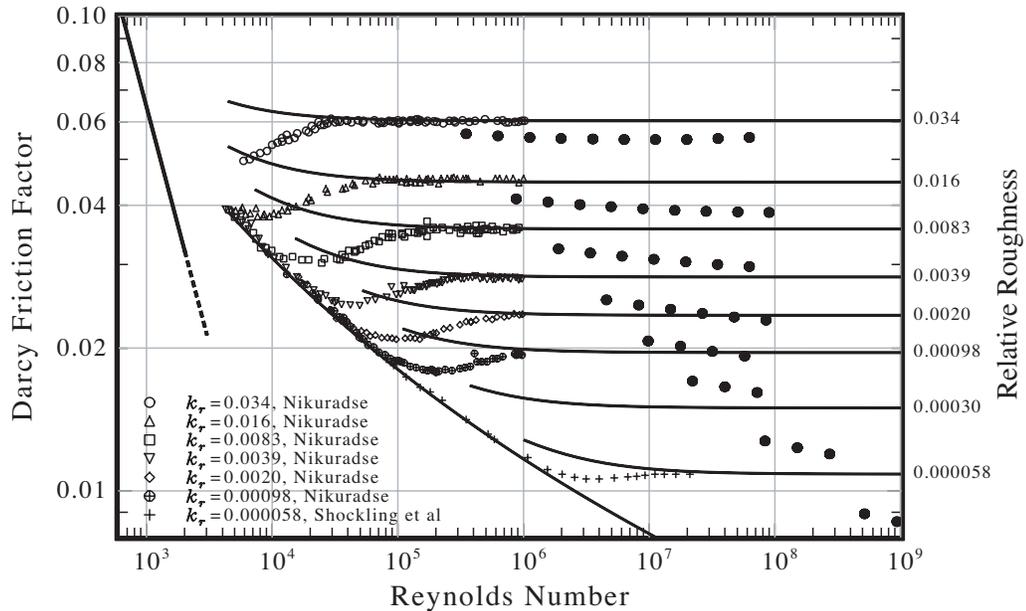


Fig. 7.8 Initial friction factor results for the Phillips k - λ model.

IV. Summary and Conclusions

The bulk of what is understood today about fully rough turbulent flow is based on the original work of Nikuradse [59] who performed a substantial set of experiments for a range of roughness Reynolds numbers. Nikuradse correlated the friction coefficient, wall roughness, and roughness Reynolds number results according to the expression given in Eq. (7.24) which in this work is called the Nikuradse number. Nikuradse found that for fully rough flows, the Nikuradse number was independent of roughness Reynolds number. This fully rough limit is so ingrained in subsequent work, that it has come to define the meaning of roughness.

Nikuradse's work suggests that fully rough flow begins around $k_s^+ \approx 70.0$ whereas the work of Colebrook [73] suggests that the fully rough region begins around $k_s^+ \approx 100.0$. However, it has been shown analytically that in order for the flow to truly be considered fully rough, it should have a roughness Reynolds number in the range $k_s^+ \geq 1000.0$.

The empirical relation of Kays and Crawford [96] for the eddy viscosity in fully rough pipe flow was used to develop estimates for the mean vortex wavelength distribution. From this, an algebraic relation for the mean vortex wavelength was developed along with initial values for the closure coefficients. This

complete system of equations constitutes the Phillips k - λ model for fully rough pipe flow. This model along with the associated boundary conditions is summarized in Eqs. (7.68) and (7.69). The system of equations can be solved using a fourth-order Runge-Kutta integration method, as explained above, and initial results for the model have been shown. Using the initial estimates for the closure coefficients, velocity profiles and eddy-viscosity distributions were obtained which match empirical relations. However, the friction factor predicted by the model is significantly in error. This error can be minimized by evaluating more appropriate closure coefficients for the model.

CHAPTER 8

PHILLIPS k - λ MODEL CLOSURE COEFFICIENT EVALUATION**I. Introduction**

The Phillips k - λ model for fully rough pipe flow in nondimensional form comprises the following equations and boundary conditions

$$\begin{aligned}
 \frac{du^+}{d\hat{r}} &= -\frac{R_\tau \hat{r}}{(1+\nu^+)} \\
 -\frac{5}{3\hat{r}} \frac{d}{d\hat{r}} \left[(1+\nu^+/\sigma_k) \hat{r} \frac{dk^+}{d\hat{r}} \right] &= \nu^+ \left(\frac{\partial u^+}{\partial \hat{r}} \right)^2 - \left[R_\tau^2 C_\lambda \frac{k^+}{\lambda^{+2}} + \frac{4}{3\hat{r}} \frac{d}{d\hat{r}} \left(\hat{r} \frac{dk^+}{d\hat{r}} \right) \right] \\
 \lambda^+ &= \min[C_{r1} R_{ec}^{a_{r1}} R_\tau, C_{r2} k_s^{+(1+a_{r2})} + C_{r3} R_{ec}^{a_{r3}} (1-\hat{r})^{a_{r4}} R_\tau] \\
 \nu^+ &= \lambda^+ \sqrt{k^+} \\
 R_{ec} &= [2u^+(0)]^{3/2} \{R_\tau [1+\nu^+(0)]\}^{1/2} \\
 u^+ \Big|_{\hat{r}=1} &= 0, \quad \frac{dk^+}{d\hat{r}} \Big|_{\hat{r}=0} = 0, \quad k^+ \Big|_{\hat{r}=1} = \left[\frac{\sqrt{(1/k_s^+)^2 + (2\gamma\kappa)^2} - 1/k_s^+}{2C_{r2} k_s^{+a_{r2}}} \right]^2
 \end{aligned} \tag{8.1}$$

where the model constants are σ_k , C_λ , C_{r1} , a_{r1} , C_{r2} , a_{r2} , C_{r3} , a_{r3} , and a_{r4} . Values for these model constants have a significant impact on the accuracy of the model. Therefore, accurate values for these constants must be evaluated. The influence of C_λ on the model solution can be analytically determined. However, in order to determine acceptable values for the other model constants, computer optimization was used. This chapter explains the process for evaluating the model constants.

II. The Influence of C_λ for Fully Rough Pipe Flow

The influence of the closure coefficient C_λ is most easily seen by considering the fundamental dimensional equations of the model

$$\frac{d\bar{V}_z}{dr} = -\frac{u_\tau^2}{(\nu + \nu_t)} \frac{r}{R} \tag{8.2}$$

$$-\frac{5}{3r} \frac{d}{dr} \left[(v + v_t / \sigma_k) r \frac{dk}{dr} \right] = v_t \left(\frac{\partial \bar{V}_z}{\partial r} \right)^2 - v \left[C_\lambda \frac{k}{\lambda^2} + \frac{4}{3r} \frac{d}{dr} \left(r \frac{dk}{dr} \right) \right] \quad (8.3)$$

$$\lambda = \min [C_{r1} R_{ec}^{a_{r1}} R, C_{r2} k_s^{+a_{r2}} k_s + C_{r3} R_{ec}^{a_{r3}} (1 - r/R)^{a_{r4}} R] \quad (8.4)$$

$$v_t = \lambda \sqrt{k} \quad (8.5)$$

Equation (8.4) shows that the mean vortex wavelength is greatest near the centerline and smallest near the wall. Because the dissipation of turbulent kinetic energy is inversely proportional to the square of the mean vortex wavelength, Eq. (8.3) shows that the dissipation of turbulent kinetic energy is greatest at the wall where the mean vortex wavelength is small. Also note that because the turbulent kinetic energy and mean vortex wavelength are large at the center of the pipe, Eq. (8.5) shows that the turbulent eddy viscosity is also large at the center of the pipe. Therefore, in the core region of the pipe, the diffusion and generation terms of the turbulent kinetic energy become the dominant terms. The dissipation term only becomes significant near the wall where the mean vortex wavelength is much smaller. Therefore, Eq. (8.3) shows that the turbulent kinetic energy that is generated in the core of the pipe is diffused toward the wall where it is dissipated. This makes sense from a physical point of view as the energy from the large eddies diffuses toward the wall where these eddies break down to form the smaller eddies. These small eddies have higher strain rates and are responsible for much of the dissipation of energy. Thus the bulk of the dissipation of turbulent kinetic energy takes place in a layer very near the wall. As the roughness Reynolds number is increased, this dissipation layer becomes thinner. At very high roughness Reynolds numbers, the molecular viscosity is negligible compared with the turbulent eddy viscosity across most of the flow, and the dissipation layer becomes infinitesimally thin. At that point, the solution becomes independent of the molecular viscosity, and Eqs. (8.2) and (8.3) simplify to

$$\frac{d\bar{V}_z}{dr} = -\frac{u_\tau^2}{v_t} \frac{r}{R} \quad (8.6)$$

$$-\frac{5}{3r} \frac{d}{dr} \left[\left(\frac{v_t}{\sigma_k} \right) r \frac{dk}{dr} \right] = v_t \left(\frac{\partial \bar{V}_z}{\partial r} \right)^2 \quad (8.7)$$

Using Eq. (8.6) in Eq. (8.7) gives

$$\frac{d}{dr} \left[v_t r \frac{dk}{dr} \right] = -\frac{3\sigma_k u_t^4 r^3}{5v_t R^2} \quad (8.8)$$

Therefore, at high roughness Reynolds numbers, the solution is independent of the model constant C_λ . At lower roughness Reynolds numbers, and particularly in the transition and smooth-wall limits, C_λ will significantly affect the solution. Note from Eq. (8.3) that the point that the molecular viscosity becomes negligible compared to the turbulent eddy viscosity is dependent on the value of C_λ . Thus, C_λ determines to some extent the transition point to fully rough flow. However, as the roughness Reynolds number is increased, the solution will eventually become independent of C_λ . Because this work focuses on a model for the fully rough flow asymptote, for the remainder of this work the solution is assumed independent of the value for C_λ .

III. Computer Optimization

Optimization routines are helpful in evaluating optimal values for a set of inputs for a given situation. These input variables are commonly called the design variables, and the range of viable values for the design variables is called the design space. In order for an optimization routine to evaluate the best solution for the design variables within the design space, the relationship between the design variables and the solution must be expressed quantitatively. Once this solution is obtained for a given set of design variables, the degree to which that solution matches the desired solution is commonly called the fitness of the solution. The fitness must be a single number that comes as a direct result from a given set of design variables. The routine that calculates the fitness as a function of the design variables is commonly called the fitness function. Fitness functions can be as simple as an algebraic expression, or as complicated as a turbulence model. The optimization routine itself treats the fitness function as a sort of a “black box” which given a set of design variables returns a fitness value. Thus, an early step to any optimization problem is to define the fitness function, which, if minimized, results in the optimal solution.

A. Model Fitness

It has been shown that the solution for this turbulence model in the fully rough regime is independent of the model constant C_λ . Therefore, the design variables for this case are the remaining model constants σ_k , C_{r1} , a_{r1} , C_{r2} , a_{r2} , C_{r3} , a_{r3} , and a_{r4} . Given values for each of these model constants, a fitness of the model must be evaluated. The fitness of any given set of closure coefficients was determined by evaluating how well the model matched the Colebrook equation in the fully rough limit.

For a given set of model constants, 46 cases were run over a range of values for k_r and k_s^+ that all fall in the fully rough regime. The values for k_r were taken to match the values for the experimental cases of Nikuradse [59] and Shockling, Allen, and Smits [61] with an additional k_r value to fill in the large gap between the data sets of the researchers. The values for k_s^+ began at $k_s^+ = 1000$ and were incremented up to values that resulted in bulk Reynolds numbers at or just above $R_e = 1.0e8$, or until at least 3 cases were run. This provided a range of cases that most nearly match the range of data sets available.

For each of the 46 cases, the resulting friction factor was compared with the friction factor given by the Colebrook equation at the Reynolds number computed by the model. The percent error in the friction factor for each case was evaluated, squared, and summed with the errors of the other cases. This total was divided by 46 and the square root of that value was taken to yield the root-mean-square (RMS) percent error of the model. This single value was used as the fitness for any given set of model constants.

For example, given the non-optimal model constants

$$\begin{aligned} \sigma_k = 1.0, \quad C_{r1} = 0.065, \quad a_{r1} = 0.0, \quad C_{r2} = 0.002, \quad a_{r2} = -0.004, \\ C_{r3} = 0.063, \quad a_{r3} = -0.003, \quad a_{r4} = 1.12 \end{aligned} \quad (8.9)$$

produces the results shown in Table 8.1.

Table 8.1 Sample results for the Phillips k - λ fully rough pipe flow model given a set of non-optimal model constants

k_r	k_s^+	R_c	Phillips $4C_f$	Colebrook $4C_f$	% Error
3.40E-02	1.00E+3.00	3.34E+05	6.20E-02	6.04E-02	2.77E-02
3.40E-02	1.00E+3.25	5.93E+05	6.22E-02	6.03E-02	3.16E-02
3.40E-02	1.00E+3.50	1.05E+06	6.23E-02	6.03E-02	3.37E-02
3.40E-02	1.00E+3.75	1.87E+06	6.24E-02	6.03E-02	3.47E-02
3.40E-02	1.00E+4.00	3.33E+06	6.24E-02	6.03E-02	3.50E-02
3.40E-02	1.00E+4.25	5.92E+06	6.24E-02	6.03E-02	3.50E-02
3.40E-02	1.00E+4.50	1.05E+07	6.24E-02	6.03E-02	3.47E-02
3.40E-02	1.00E+4.75	1.87E+07	6.23E-02	6.03E-02	3.43E-02
3.40E-02	1.00E+5.00	3.33E+07	6.23E-02	6.03E-02	3.39E-02
3.40E-02	1.00E+5.25	5.93E+07	6.23E-02	6.03E-02	3.34E-02
1.60E-02	1.00E+3.00	8.26E+05	4.58E-02	4.48E-02	2.25E-02
1.60E-02	1.00E+3.25	1.47E+06	4.59E-02	4.48E-02	2.59E-02
1.60E-02	1.00E+3.50	2.61E+06	4.60E-02	4.47E-02	2.77E-02
1.60E-02	1.00E+3.75	4.63E+06	4.60E-02	4.47E-02	2.84E-02
1.60E-02	1.00E+4.00	8.24E+06	4.60E-02	4.47E-02	2.85E-02
1.60E-02	1.00E+4.25	1.47E+07	4.60E-02	4.47E-02	2.83E-02
1.60E-02	1.00E+4.50	2.61E+07	4.60E-02	4.47E-02	2.78E-02
1.60E-02	1.00E+4.75	4.64E+07	4.60E-02	4.47E-02	2.73E-02
1.60E-02	1.00E+5.00	8.25E+07	4.59E-02	4.47E-02	2.66E-02
8.30E-03	1.00E+3.00	1.79E+06	3.64E-02	3.57E-02	2.06E-02
8.30E-03	1.00E+3.25	3.17E+06	3.65E-02	3.56E-02	2.36E-02
8.30E-03	1.00E+3.50	5.64E+06	3.65E-02	3.56E-02	2.50E-02
8.30E-03	1.00E+3.75	1.00E+07	3.65E-02	3.56E-02	2.55E-02
8.30E-03	1.00E+4.00	1.78E+07	3.65E-02	3.56E-02	2.54E-02
8.30E-03	1.00E+4.25	3.17E+07	3.65E-02	3.56E-02	2.50E-02
8.30E-03	1.00E+4.50	5.64E+07	3.65E-02	3.56E-02	2.44E-02
3.90E-03	1.00E+3.00	4.27E+06	2.88E-02	2.82E-02	2.01E-02
3.90E-03	1.00E+3.25	7.59E+06	2.89E-02	2.82E-02	2.26E-02
3.90E-03	1.00E+3.50	1.35E+07	2.89E-02	2.82E-02	2.38E-02
3.90E-03	1.00E+3.75	2.40E+07	2.89E-02	2.82E-02	2.41E-02
3.90E-03	1.00E+4.00	4.27E+07	2.89E-02	2.82E-02	2.38E-02
3.90E-03	1.00E+4.25	7.59E+07	2.89E-02	2.82E-02	2.33E-02
2.00E-03	1.00E+3.00	9.14E+06	2.39E-02	2.34E-02	2.02E-02
2.00E-03	1.00E+3.25	1.62E+07	2.40E-02	2.34E-02	2.25E-02
2.00E-03	1.00E+3.50	2.89E+07	2.40E-02	2.34E-02	2.35E-02
2.00E-03	1.00E+3.75	5.14E+07	2.40E-02	2.34E-02	2.36E-02
2.00E-03	1.00E+4.00	9.14E+07	2.40E-02	2.34E-02	2.32E-02
9.80E-04	1.00E+3.00	2.04E+07	1.99E-02	1.96E-02	2.02E-02
9.80E-04	1.00E+3.25	3.63E+07	2.00E-02	1.95E-02	2.23E-02
9.80E-04	1.00E+3.50	6.45E+07	2.00E-02	1.95E-02	2.32E-02
3.00E-04	1.00E+3.00	7.64E+07	1.52E-02	1.49E-02	1.82E-02
3.00E-04	1.00E+3.25	1.36E+08	1.52E-02	1.49E-02	2.03E-02
3.00E-04	1.00E+3.50	2.41E+08	1.53E-02	1.49E-02	2.10E-02
5.80E-05	1.00E+3.00	4.66E+08	1.09E-02	1.08E-02	8.96E-03
5.80E-05	1.00E+3.25	8.28E+08	1.10E-02	1.08E-02	1.15E-02
5.80E-05	1.00E+3.50	1.47E+09	1.10E-02	1.08E-02	1.24E-02

Squaring and summing each value in the “% Error” column, dividing by 46, and taking the square root gives 2.5808% RMS error. This is the fitness for this case. Figure 8.1 shows the results of this case with respect to the Colebrook equation. The results of the model using the constants given in Eq. (8.9) fall just above the Colebrook equation. A fitness of 0.0% RMS error would show the model results falling directly on the Colebrook equation.

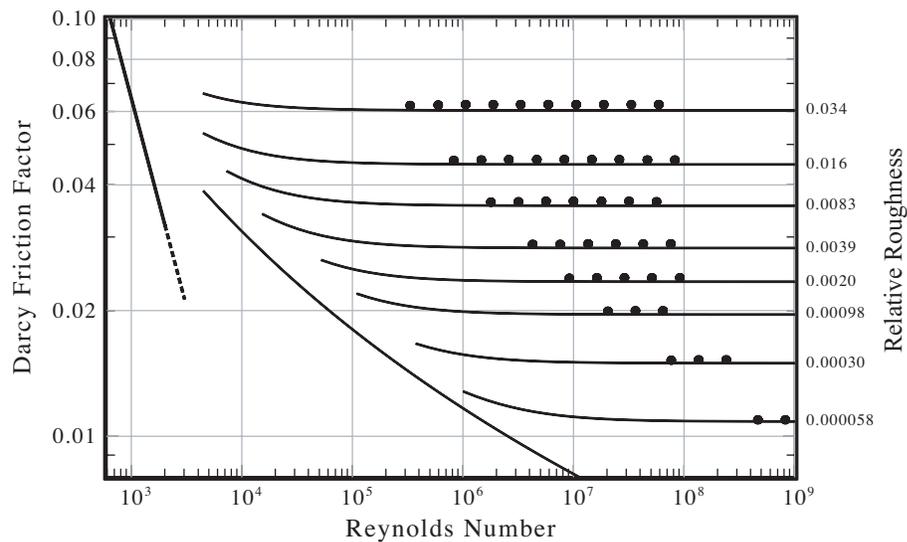


Fig. 8.1 Sample friction factor results for the Phillips $k-\lambda$ fully rough pipe flow model given a set of non-optimal model constants.

B. Common Optimization Algorithms

Once the fitness has been defined as a function of the design variables, an optimization algorithm can be used to determine the values for the variables that will minimize the fitness function. Many types of optimization methods exist which can generally be classified as gradient-based, discrete variable, or evolutionary methods. Gradient-based optimization algorithms are generally used to solve optimization problems for which the fitness is a continuous and differentiable function within the design space. Discrete variable algorithms are used when only discrete values for the design variables are viable options. Evolutionary routines, sometimes called genetic algorithms, comprise a relatively new branch of optimization and attempt to mimic nature in a sense of survival of the fittest. These schemes are very useful for optimization problems for which the design space is discontinuous, has multiple local saddle points and

minima, or has regions where the fitness function cannot be solved. Because the fitness function for this turbulence model is expected to be continuous throughout the design space, a gradient-based optimization algorithm was chosen for the optimization routine.

Three common types of gradient-based methods are steepest decent methods, Newton's method, and quasi-Newton methods. The advantages of steepest decent methods are that they make good progress toward an optimum when they are far from the optimum, they always progress downhill, and will always eventually converge to a local minimum. However, if the function to be minimized is eccentric, these methods can require an immense amount of time to reach convergence. The advantage of Newton's method is that it can quickly find a point for which the gradient is zero, especially if the function to be minimized is nearly quadratic. This is commonly made possible through the evaluation of second derivatives to construct the Hessian matrix of the fitness function. Thus, if it is near an optimum, it will quickly find the optimum regardless of the eccentricity of the function. However, Newton's method has several disadvantages. For example, it cannot differentiate between a maximum, minimum, or saddle point. Thus, it does not guarantee that it will progress downhill, and it may even converge to a maximum rather than a minimum. Additionally, it requires the evaluation of second derivatives, which can be computationally expensive. Quasi-Newton methods combine the advantages of steepest decent methods with those of Newton's method and eliminate many of the disadvantages. Quasi-Newton methods, sometimes termed "variable metric methods", begin as steepest decent methods, and store first derivative information in order to estimate the Hessian matrix as the solution progresses. As it approaches the optimum, a quasi-Newton method resembles a Newton's method where it can use the Hessian to progress more quickly to the optimum.

In general, quasi-Newton methods allow a direction matrix to be determined and refined as the solution progresses toward the optimum. This direction matrix is multiplied by the negative of the gradient in order to evaluate a search direction. In mathematical terms,

$$\bar{\mathbf{s}} = -\bar{\mathbf{N}}\nabla f(\bar{\mathbf{x}}) \quad (8.10)$$

where $\nabla f(\bar{\mathbf{x}})$ is the gradient of the vector of design variables, $\bar{\mathbf{N}}$ is the direction matrix, and $\bar{\mathbf{s}}$ is the search direction. Note that if the direction matrix is positive definite, the search direction will always point

downhill. If the direction matrix is equal to the identity matrix, then the method is a steepest decent method. If the direction matrix is equal to the inverse of the Hessian, then the method is equivalent to Newton's method. Most methods for updating the direction matrix begin with the identity matrix and attempt to construct the Hessian through the evaluation of first derivatives as the solution progresses. Multiple update methods have been developed which ensure that the direction matrix is always positive definite. One such update method was chosen for this work and an overview of this method is included here.

C. Broyden-Fletcher-Goldfarb-Shanno Update Method

One of the best quasi-Newton's methods to date is the Broyden-Fletcher-Goldfarb-Shanno (BFGS) update method named after Broyden [97], Fletcher [98], Goldfarb [99], and Shanno [100] who each came up with the update independently in the same year. This update method can be written as

$$\bar{\bar{\mathbf{N}}}_{i+1} = \bar{\bar{\mathbf{N}}}_i + \left(1 + \frac{\bar{\gamma}_i^T \bar{\bar{\mathbf{N}}}_i \bar{\gamma}_i}{\Delta \bar{\mathbf{x}}_i^T \bar{\gamma}_i} \right) \left(\frac{\Delta \bar{\mathbf{x}}_i \Delta \bar{\mathbf{x}}_i^T}{\Delta \bar{\mathbf{x}}_i^T \bar{\gamma}_i} \right) - \frac{\Delta \bar{\mathbf{x}}_i \bar{\gamma}_i^T \bar{\bar{\mathbf{N}}}_i + \bar{\bar{\mathbf{N}}}_i \bar{\gamma}_i \Delta \bar{\mathbf{x}}_i^T}{\Delta \bar{\mathbf{x}}_i^T \bar{\gamma}_i} \quad (8.11)$$

where

$$\begin{aligned} \Delta \bar{\mathbf{x}}_i &\equiv \bar{\mathbf{x}}_{i+1} - \bar{\mathbf{x}}_i \\ \bar{\gamma}_i &\equiv \nabla f(\bar{\mathbf{x}}_{i+1}) - \nabla f(\bar{\mathbf{x}}_i) \end{aligned} \quad (8.12)$$

and the subscript i is the iteration or step number. This update is commonly combined with a line search algorithm. Given a search direction, a line search can be performed in that direction until a minimum is found. The line search in this work included running several cases at specific intervals along the search direction and fitting a parabola to the three points that comprised the minimum point and the two points to either side of that minimum point. Once a parabola had been fit to those three points, the minimum of the parabola was taken as the next step or iteration in the BFGS algorithm. This algorithm is perhaps best understood by example.

Given the fitness function

$$f(\bar{\mathbf{x}}) = x_1^2 - 2x_1x_2 + 4x_2^2 \quad (8.13)$$

we seek values for the design variables x_1 and x_2 that minimize this function. Starting from the point

$$\bar{\mathbf{x}}_0 = \begin{Bmatrix} -3 \\ 1 \end{Bmatrix}, \quad \nabla f(\bar{\mathbf{x}}_0) = \begin{Bmatrix} -8 \\ 14 \end{Bmatrix}, \quad f(\bar{\mathbf{x}}_0) = 19 \quad (8.14)$$

For the first step, the identity matrix is used for the direction matrix

$$\bar{\bar{\mathbf{N}}}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (8.15)$$

This gives the search direction

$$\bar{\mathbf{s}}_0 = -\bar{\bar{\mathbf{N}}}_0 \nabla f(\bar{\mathbf{x}}_0) = -\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} -8 \\ 14 \end{Bmatrix} = \begin{Bmatrix} 8 \\ -14 \end{Bmatrix} \quad (8.16)$$

A line search is performed in this direction, and a parabola is fit to the minimum of the line search to give the next point

$$\bar{\mathbf{x}}_1 = \begin{Bmatrix} -2.030 \\ -0.698 \end{Bmatrix}, \quad \nabla f(\bar{\mathbf{x}}_1) = \begin{Bmatrix} -2.664 \\ -1.522 \end{Bmatrix}, \quad f(\bar{\mathbf{x}}_1) = 3.235 \quad (8.17)$$

Using Eqs. (8.14) and (8.17) in Eq. (8.12) gives

$$\begin{aligned} \Delta \bar{\mathbf{x}}_0 &\equiv \bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_0 = \begin{Bmatrix} 0.970 \\ -1.698 \end{Bmatrix} \\ \bar{\gamma}_0 &\equiv \nabla f(\bar{\mathbf{x}}_1) - \nabla f(\bar{\mathbf{x}}_0) = \begin{Bmatrix} 5.336 \\ -15.522 \end{Bmatrix} \end{aligned} \quad (8.18)$$

Using this in Eq. (8.11) gives the updated direction matrix

$$\bar{\bar{\mathbf{N}}}_1 = \bar{\bar{\mathbf{N}}}_0 + \left(1 + \frac{\bar{\gamma}_0^T \bar{\bar{\mathbf{N}}}_0 \bar{\gamma}_0}{\Delta \bar{\mathbf{x}}_0^T \bar{\gamma}_0} \right) \begin{pmatrix} \Delta \bar{\mathbf{x}}_0 \Delta \bar{\mathbf{x}}_0^T \\ \Delta \bar{\mathbf{x}}_0^T \bar{\gamma}_0 \end{pmatrix} - \frac{\Delta \bar{\mathbf{x}}_0 \bar{\gamma}_0^T \bar{\bar{\mathbf{N}}}_0 + \bar{\bar{\mathbf{N}}}_0 \bar{\gamma}_0 \Delta \bar{\mathbf{x}}_0^T}{\Delta \bar{\mathbf{x}}_0^T \bar{\gamma}_0} = \begin{bmatrix} 0.957 & 0.266 \\ 0.266 & 0.200 \end{bmatrix} \quad (8.19)$$

Using this in Eq. (8.10) gives the new search direction

$$\bar{\mathbf{s}}_1 = -\bar{\bar{\mathbf{N}}}_1 \nabla f(\bar{\mathbf{x}}_1) = -\begin{bmatrix} 0.957 & 0.266 \\ 0.266 & 0.200 \end{bmatrix} \begin{Bmatrix} -2.664 \\ -1.522 \end{Bmatrix} = \begin{Bmatrix} 2.954 \\ 1.015 \end{Bmatrix} \quad (8.20)$$

Performing a line search in this direction brings us within single precision computations of the optimum.

The process can be repeated until double precision is achieved. The optimum for this design space is

$$\bar{\mathbf{x}} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}, \quad \nabla f(\bar{\mathbf{x}}) = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}, \quad f(\bar{\mathbf{x}}) = 0 \quad (8.21)$$

At that point, the algorithm can exit. However, it is interesting to note that if the direction matrix is updated before exiting, it gives the inverse of the Hessian matrix

$$\bar{\bar{\mathbf{N}}} = \bar{\bar{\mathbf{H}}}^{-1} = \begin{bmatrix} 0.667 & 0.167 \\ 0.167 & 0.167 \end{bmatrix} \quad (8.22)$$

Because the direction matrix is a function of the previous direction matrix, if the nature of the design space is such that the optimization routine requires several iterations before an optimum is achieved, the direction matrix may contain information that is not applicable near an optimum. For this reason it can be helpful to reset the direction matrix to the identity matrix periodically. In this work, the direction matrix was reset to the identity matrix each time the resulting step size was

$$|\Delta \bar{x}_i| \leq 1.0e-12 \quad (8.23)$$

If the direction matrix was equal to the identity matrix and the criteria in Eq. (8.23) was met, the solution was considered finished and the algorithm exited.

The gradient required by the algorithm in Eq. (8.12) was computed numerically using a second-order central-differencing method. It was found that a step size of 1.0e-8 for the differencing provided a good trade-off between truncation and round-off error. This was the step size used for differencing in the code. Appendix P contains the Fortran code used for the optimization routine.

D. An Example Optimization Run

An example optimization run may be helpful to the reader. Beginning with the non-optimal values for the constants given in Eq. (8.9) and holding $\sigma_k = 1.0$, $C_{r1} = 0.065$, and $a_{r1} = 0.0$ constant, the optimization routine took the steps shown in Table 8.2. The resulting model constants for this case are

$$\begin{aligned} \sigma_k = 1.0, \quad C_{r1} = 0.065, \quad a_{r1} = 0.0, \quad C_{r2} = 2.0768E-03, \quad a_{r2} = -3.6480E-03, \\ C_{r3} = 6.2543E-02, \quad a_{r3} = -3.0563E-03, \quad a_{r4} = 1.1202E+00 \end{aligned} \quad (8.24)$$

which results in an RMS error of 0.2136%. Figure 8.2 shows the model results for the constants evaluated from the optimization run. Also included are the corresponding results for the Colebrook equation.

Table 8.2 Example optimization iterations using the BFGS update method

iteration	C_{r2}	a_{r2}	C_{r3}	a_{r3}	a_{r4}	Fitness
0	2.0000E-03	-4.0000E-03	6.3000E-02	-3.0000E-03	1.1200E+00	2.5808E+00
1	2.1749E-03	-3.9968E-03	6.2986E-02	-3.0155E-03	1.1200E+00	8.0540E-01
2	2.1999E-03	-4.1727E-03	6.3320E-02	-3.0288E-03	1.1202E+00	7.9584E-01
3	2.4371E-03	-8.7041E-03	6.9826E-02	-5.9085E-03	1.1258E+00	5.6610E-01
4	2.5871E-03	-1.2737E-02	7.1871E-02	-6.0652E-03	1.1295E+00	4.3093E-01
5	2.6995E-03	-1.9510E-02	7.2916E-02	-8.5437E-03	1.1236E+00	4.0144E-01
6	2.4684E-03	-1.4187E-02	6.8653E-02	-5.8296E-03	1.1242E+00	2.9050E-01
7	2.4222E-03	-1.4296E-02	6.7363E-02	-5.8726E-03	1.1210E+00	2.7447E-01
8	2.1622E-03	-5.3522E-03	6.4061E-02	-3.4922E-03	1.1218E+00	2.2605E-01
9	2.1307E-03	-4.4432E-03	6.3595E-02	-3.1936E-03	1.1219E+00	2.2352E-01
10	2.1252E-03	-4.6136E-03	6.3391E-02	-3.1418E-03	1.1217E+00	2.1715E-01
11	2.1122E-03	-5.0337E-03	6.3106E-02	-3.5625E-03	1.1198E+00	2.1455E-01
12	2.0867E-03	-3.9178E-03	6.2766E-02	-3.1851E-03	1.1203E+00	2.1377E-01
13	2.0800E-03	-3.7579E-03	6.2596E-02	-3.0954E-03	1.1202E+00	2.1360E-01
14	2.0800E-03	-3.7579E-03	6.2596E-02	-3.0954E-03	1.1202E+00	2.1360E-01
15	2.0799E-03	-3.7579E-03	6.2596E-02	-3.0954E-03	1.1202E+00	2.1360E-01
16	2.0800E-03	-3.7578E-03	6.2596E-02	-3.0950E-03	1.1202E+00	2.1360E-01
17	2.0798E-03	-3.7522E-03	6.2596E-02	-3.0960E-03	1.1202E+00	2.1360E-01
18	2.0775E-03	-3.6923E-03	6.2553E-02	-3.0762E-03	1.1201E+00	2.1359E-01
19	2.0768E-03	-3.6480E-03	6.2543E-02	-3.0563E-03	1.1202E+00	2.1359E-01
20	2.0768E-03	-3.6480E-03	6.2543E-02	-3.0563E-03	1.1202E+00	2.1359E-01
21	2.0768E-03	-3.6480E-03	6.2543E-02	-3.0563E-03	1.1202E+00	2.1359E-01
22	2.0768E-03	-3.6480E-03	6.2543E-02	-3.0563E-03	1.1202E+00	2.1359E-01

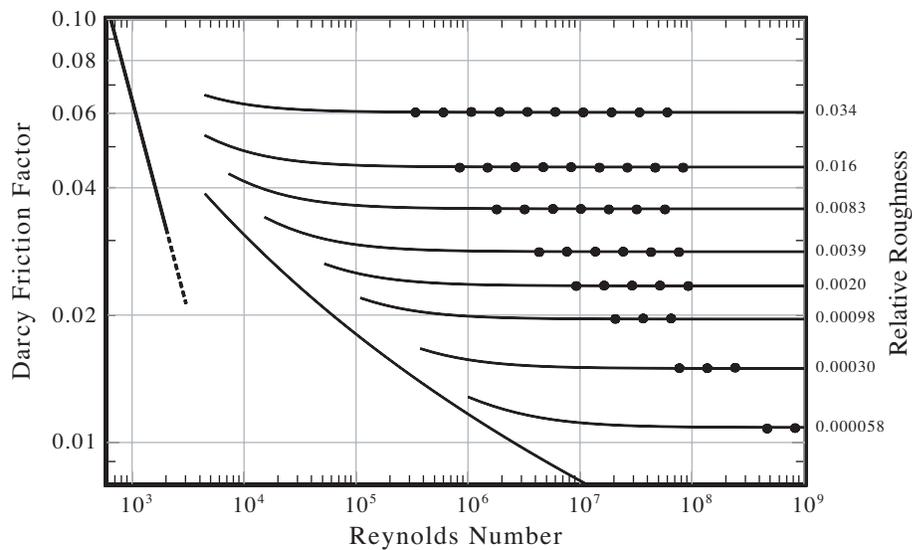


Fig. 8.2 Friction factor results for the Phillips k - λ fully rough pipe flow model with a set of optimal model constants given in Eq. (8.24).

Comparing Figs. 8.1 and 8.2, it is easy to see that these model constants provide a significant improvement over the original set of model constants given in Eq. (8.9) which resulted in an RMS error of 2.5808%.

E. Resulting Closure Coefficients

Initial estimates for the model constants were obtained by fitting the algebraic relation for λ^+ to results derived from the mixing length model. Early optimization runs were set to allow all the model constants to vary except for C_λ , which was held at the value $C_\lambda = 0.0004$. Results from these runs showed that excellent model fitness could be achieved over a significant range of values for σ_k and C_{r1} . Hundreds of optimization cases were run holding both σ_k and C_{r1} at specified values in the ranges of $0.10 \leq \sigma_k \leq 2.0$ and $0.0005 \leq C_{r1} \leq 0.25$. These ranges were chosen based on estimates gained from early optimization runs. Figure 8.3 shows the % RMS error as a function of C_{r1} for several values of σ_k .

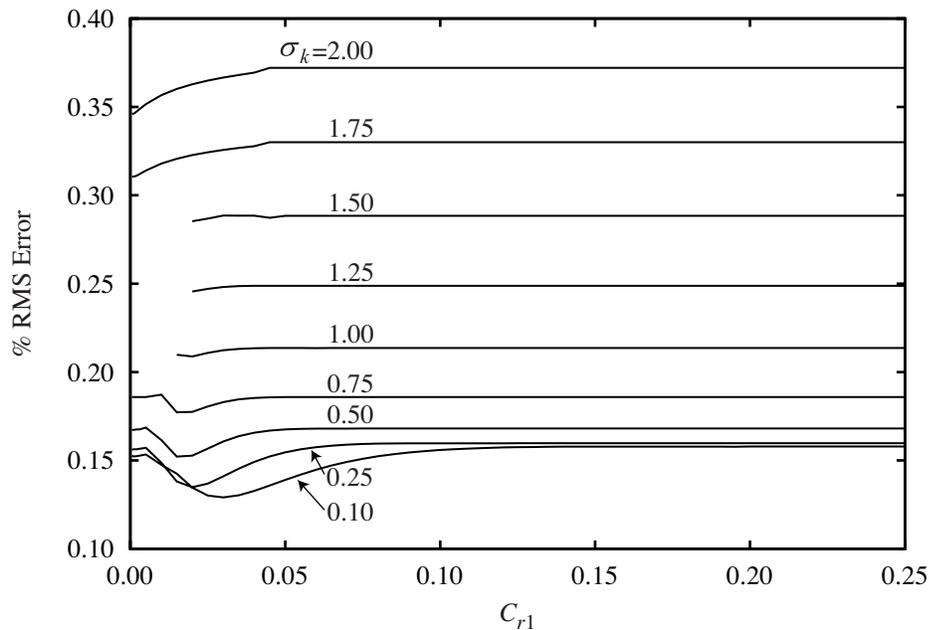


Fig. 8.3 % RMS error over a range of values for C_{r1} and σ_k .

Figure 8.3 shows two distinct regions of the design space. Note that for a given value of σ_k , the % RMS error is minimized at low values of C_{r1} . However, for high values of C_{r1} , the % RMS error is independent of C_{r1} . The region of the design space with high values of C_{r1} is attractive for two reasons.

First, if the solution at high roughness Reynolds numbers is independent of C_{r1} , the closure coefficient C_{r1} may be used as an extra degree of freedom for developing a model that transitions to lower roughness Reynolds numbers. Second, note that the % RMS error for high values of C_{r1} for any given value of σ_k is not significantly higher than the best % RMS error possible. In fact, the difference in % RMS error of these two regions is generally on the order of a few tenths of a percent. Therefore, it appears that using a solution in the region of small values for C_{r1} does not result in a model that is significantly better than using a solution with high values of C_{r1} .

In the region where the solution is independent of C_{r1} , it was found that the fitness is also independent of a_{r1} . To understand this, it is helpful to consider the algebraic relation for the mean vortex wavelength

$$\lambda^+ = \min[C_{r1}R_{ec}^{a_{r1}}R_\tau, C_{r2}k_s^{+(1+a_{r2})} + C_{r3}R_{ec}^{a_{r3}}(1-\hat{r})^{a_{r4}}R_\tau] \quad (8.25)$$

If C_{r1} and a_{r1} combine such that $C_{r1}R_{ec}^{a_{r1}}$ is large enough,

$$C_{r1}R_{ec}^{a_{r1}} > \frac{C_{r2}k_s^{+(1+a_{r2})}}{R_\tau} + C_{r3}R_{ec}^{a_{r3}}(1-\hat{r})^{a_{r4}} \quad (8.26)$$

over the entire pipe radius, and the solution becomes independent of both C_{r1} and a_{r1} . In the flat regions of Fig. 8.3, the optimized results for a_{r1} are identically 0.0 because a_{r1} was initialized to 0.0, and the results were found to be independent of both C_{r1} and a_{r1} . Initial optimization runs for which both C_{r1} and a_{r1} were allowed to vary, and for which the product $C_{r1}R_{ec}^{a_{r1}}$ was small enough to affect the results, often converged to solutions for a_{r1} that were on the order of 1.0e-4.

Again note from Fig. 8.3 that even for small values of C_{r1} where a_{r1} was also allowed to vary, the case fitness values do not significantly improve over the flat region that is independent of both C_{r1} and a_{r1} . Therefore, it appears that choosing values for the constants that give results in the lower regions of C_{r1} do not offer significant improvements over the results that can be obtained in the region where the solution is independent of both C_{r1} and a_{r1} . Overall, the case fitness appears to be a more significant function of σ_k than of C_{r1} and a_{r1} , and the case fitness is only a function of σ_k in the region that is perfectly flat.

Additional optimization cases were run extending the ranges of σ_k and C_{r1} to $0.02 \leq \sigma_k \leq 4.0$ and $0.0005 \leq C_{r1} \leq 0.50$. These ranges are expected to bound the domain of probable final values of the

coefficients. In traditional turbulence models, σ_k has ranged from 0.5 to 2.0. Thus the bound on σ_k is likely sufficiently large. Because the mean vortex wavelength is expected to be less than the pipe radius, it is expected that the upper bound for C_{r1} is about 1.0. Additionally, it was found that the solution is independent of C_{r1} for $C_{r1} > 0.50$ for all $\sigma_k \geq 0.02$. Therefore, optimization runs for $C_{r1} > 0.50$ are not needed.

In the region where the solution is independent of C_{r1} and a_{r1} , optimal values for the remaining constants and the % RMS error are only functions of σ_k . Because this region is perfectly flat with respect to C_{r1} , it will subsequently be referred to as “the flat.” Figures 8.4 through 8.8 show the optimization results for each of the remaining closure coefficients as a function of σ_k as well as curves that represent the lines of best fit for the results on the flat. Details on computing these lines of fit can be found in the Fortran subroutine included in Appendix Q that uses these lines of fit to calculate each of the closure coefficients as a function of σ_k . This method for computing the closure coefficients in this region as a function of σ_k will be referred to from here on as computing the closure coefficients from the flat.

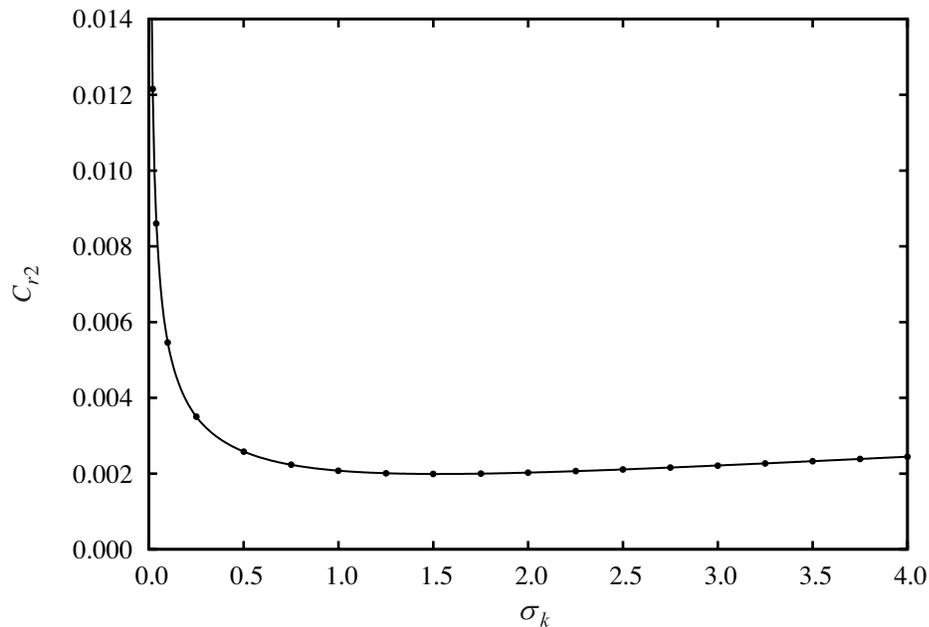


Fig. 8.4 Optimal value for C_{r2} as a function of σ_k .

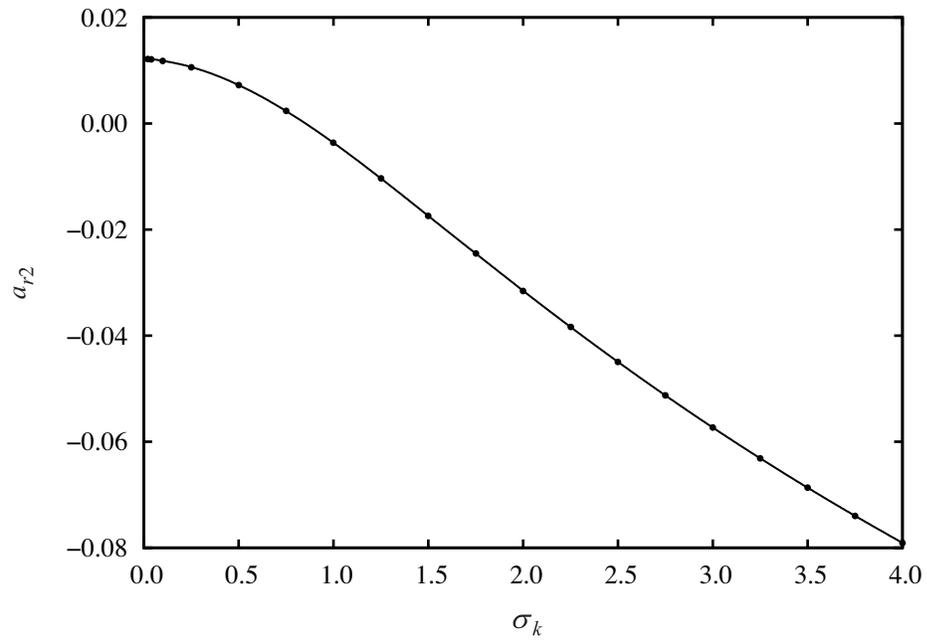


Fig. 8.5 Optimal value for a_{r2} as a function of σ_k .

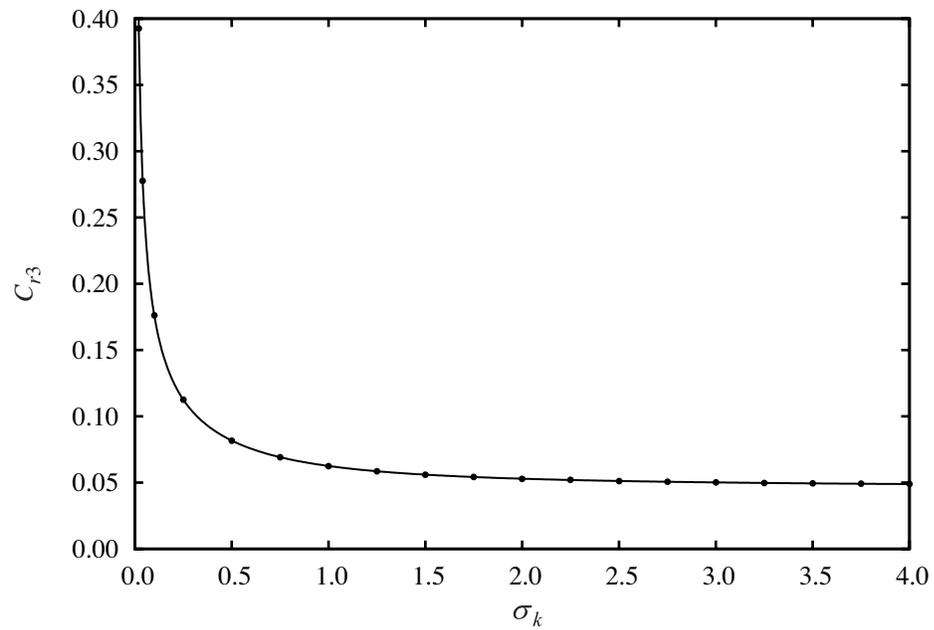


Fig. 8.6 Optimal value for C_{r3} as a function of σ_k .

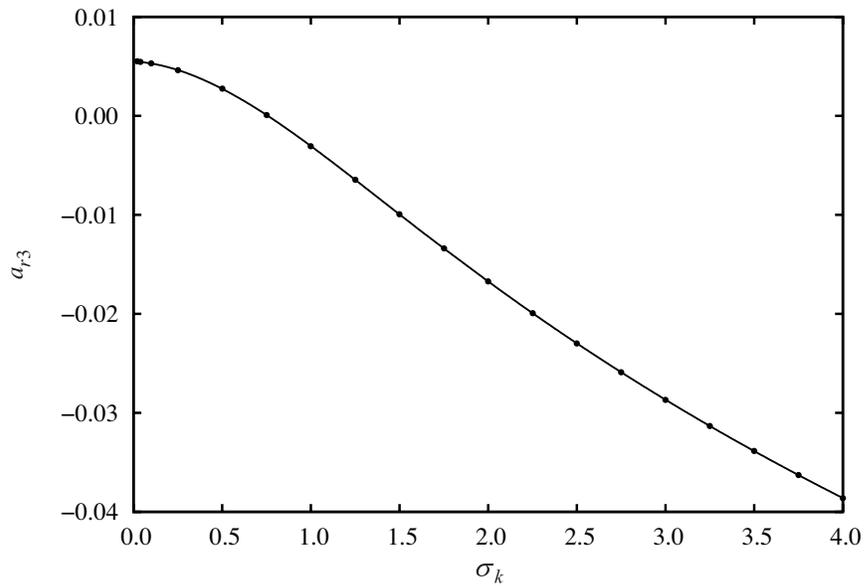


Fig. 8.7 Optimal value for a_{r3} as a function of σ_k .

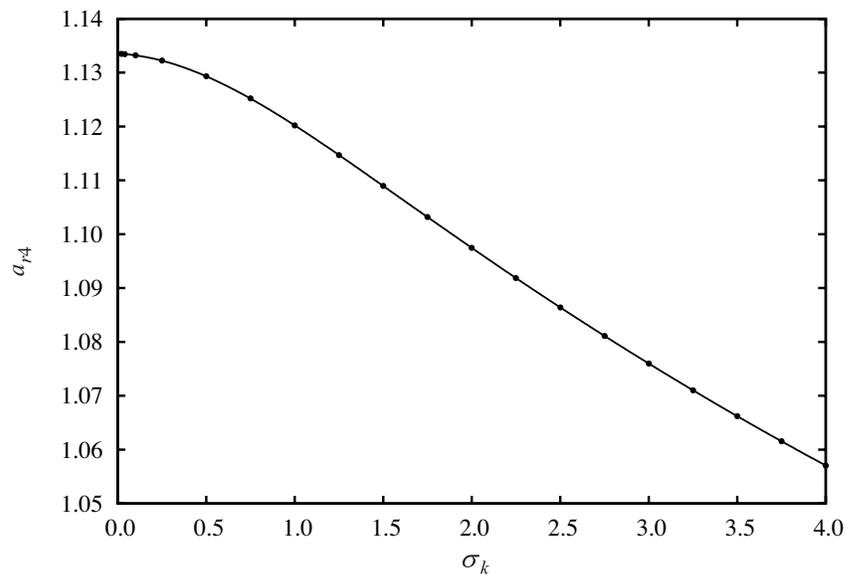


Fig. 8.8 Optimal value for a_{r4} as a function of σ_k .

A short discussion on the behaviors of the coefficients a_{r2} , a_{r3} , and a_{r4} may be of interest. Note from Fig. 8.7 that the value for a_{r3} is zero at $\sigma_k \cong 0.75$. Because $a_{r1} = 0.0$ was used for these optimization runs, choosing $\sigma_k \cong 0.75$ gives a model that is independent of the core Reynolds number. Similarly, from Fig. 8.5 it can be seen that choosing $\sigma_k \cong 0.85$ gives $a_{r2} = 0.0$ which results in a model where the mean vortex wavelength at the wall is a linear function of the roughness Reynolds number. Finally, note from Fig. 8.8

that $a_{r,4}$ is within 14% of 1.0 over the entire range included in this study. This result suggests that the mean vortex wavelength close to the wall is nearly a linear function of distance from the wall. In fact, in the region where the solution is independent of C_{r1} and a_{r1} , this result gives a model where the characteristic length of the turbulence is nearly a linear function of radius over the entire pipe cross section. This could be significant because of its similarity to Prandtl's original mixing-length theory. Prandtl hypothesized that near a smooth wall, the turbulent characteristic length is directly proportional to the distance from the wall. For roughened pipes, wall effects could be expected to extend further from the wall than for smooth pipes, affecting a larger region of the flow cross section. For fully rough pipe flow, it is probable that the wall effects extend across the entire pipe and that the characteristic length of the energy-bearing eddies would be directly proportional or very nearly proportional to the distance from the wall. Optimal results for the Phillips k - λ model suggest a value within 14% of $a_{r,4} = 1.0$ which results in a model where the mean vortex wavelength, or characteristic length of the energy-bearing eddies, is nearly directly proportional to the distance from the wall. Thus, this model for fully rough pipe flow resembles Prandtl's hypothesis for wall-bounded turbulence.

In the flat region of Fig. 8.3, the case fitness is a function of only σ_k . Figure 8.9 shows the % RMS error of the model as a function of σ_k . Note that even at the upper limit of $\sigma_k = 4.0$, the % RMS error is less than 1%.

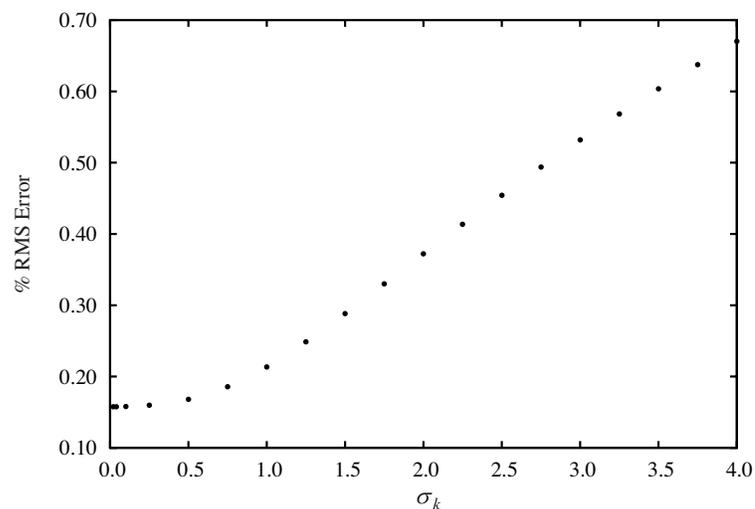


Fig. 8.9 % RMS error as a function of σ_k .

The relations shown in results of Figs. 8.4 through 8.9 were developed for the range where the solution is totally independent of C_{r1} and a_{r1} . However, if $C_{r1}R_{ec}^{a_{r1}}$ is small enough, the solution is affected as demonstrated in Fig. 8.3. Therefore, the relations shown in Figs. 8.4 through 8.9 cannot be expected to produce valid results for small values of C_{r1} . The discussion following Fig. 8.8 provides some justification as to why the fully rough solution may be expected to be independent of C_{r1} . However, at lower roughness Reynolds numbers, C_{r1} may become increasingly important as the flow transitions towards the smooth-wall asymptote, and the presence of the wall affects a smaller portion of the flow. Therefore, it is of worth to examine the % RMS error of the model for small values of C_{r1} using the coefficient relations from the flat. Figure 8.10 shows the % RMS error of the model as a function of C_{r1} for several values of σ_k . The closure coefficients over the entire range shown in Fig. 8.10 were evaluated using the flat with $a_{r1} = 0.0$. Note that the error of the model increases rapidly for low values of C_{r1} . It appears that the range of values $0.5 \leq \sigma_k \leq 1.0$ give the best model fitness over the widest range of values for C_{r1} .

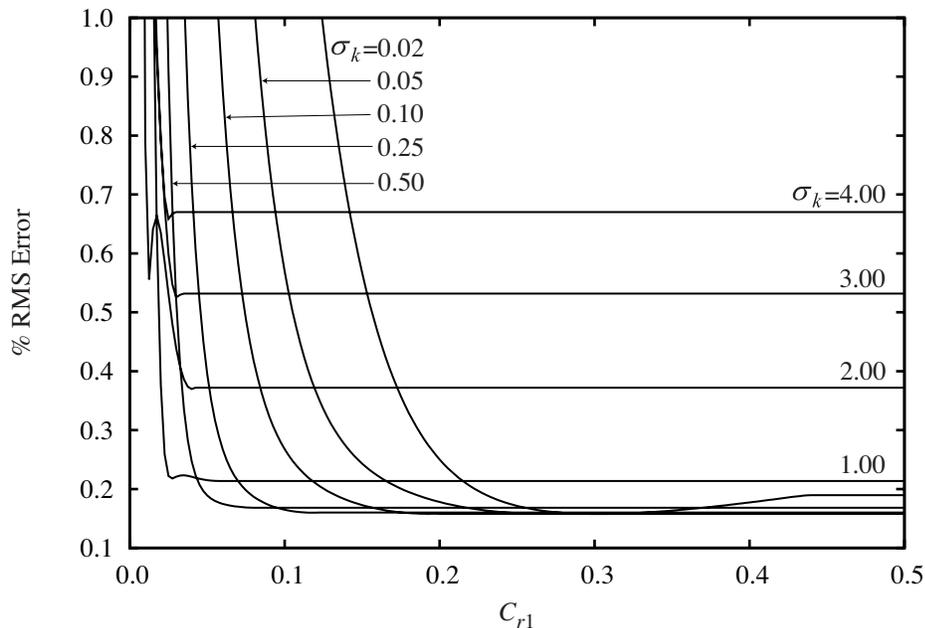


Fig. 8.10 % RMS error of the final model using coefficients evaluated from the flat over a range of values for C_{r1} and σ_k .

One point of interest apparent from Fig. 8.3 is the minimum % RMS error in the design space shown in that figure. This occurs at $\sigma_k = 0.1$ and $C_{r1} = 0.03$. This point is significant because it presents an optimal point where the value of C_{r1} is small enough to affect the solution. The % RMS error at that point is 0.129. It appears that this is very near the best solution possible with this turbulence model, and the model results at this point along with select points from the flat will be considered in the following chapter.

IV. Summary and Conclusions

The closure coefficients for the Phillips k - λ model are σ_k , C_λ , C_{r1} , a_{r1} , C_{r2} , a_{r2} , C_{r3} , a_{r3} , and a_{r4} . The closure coefficient C_λ only influences the solution through the dissipation term in the turbulent-kinetic-energy transport equation. Because the bulk of the dissipation of turbulent kinetic energy takes place in a very small region near the wall in fully rough pipe flow, the influence of C_λ is negligible for fully rough pipe flow. This has been shown analytically and confirmed numerically. However, at lower roughness Reynolds numbers, and especially in the smooth-wall limit, the influence of C_λ will be extremely important.

Values for the remaining closure coefficients, σ_k , C_{r1} , a_{r1} , C_{r2} , a_{r2} , C_{r3} , a_{r3} , and a_{r4} were evaluated using a quasi-Newton, gradient-based optimization algorithm. This algorithm employed the BFGS update method which begins as the steepest-descent method and stores information about the first derivative throughout the design space to estimate the Hessian. As the solution progresses, the stored information allows more intelligent steps to be taken until an optimum is reached.

Hundreds of optimization cases were run over a range of values for the closure coefficients. The results of this optimization study showed that the model results are independent of C_{r1} and a_{r1} over a wide range of the design space. In that region, the model predicts that the mean vortex wavelength is nearly directly proportional to the distance from the wall, which is similar to the fundamental hypothesis of Prandtl's mixing-length model. Additionally, in that region, optimal values for the remaining closure coefficients can be expressed as a function of σ_k . These relationships are shown in Figs. 8.4 through 8.8 and details on the lines of best fit for the closure coefficients can be found in Appendix Q. Choosing a value for σ_k in the

range of $0.5 \leq \sigma_k \leq 1.0$ appears to give a model with the best fit to experimental data over a wide range of values for C_{r1} . Additionally, choosing a value of $\sigma_k \cong 0.75$ gives $a_{r3} \cong 0.0$ which results in a model that is independent of the core Reynolds number.

Perhaps the best point in the design space is that for $\sigma_k = 0.1$ and $C_{r1} = 0.03$. The optimal values of the closure coefficients at this point result in a model with the minimum % RMS error in the design space of this study. The results of the model at this point along with points from the flat are considered in detail in the following chapter.

CHAPTER 9

PHILLIPS k - λ MODEL RESULTS AND CONCLUSIONS**I. Introduction**

The Phillips k - λ model and accompanying closure coefficient relations presented in the previous chapter provides an accurate method for predicting the friction factor of fully rough pipe flow. The results of the previous section show that there is a range of values for the closure coefficients that results in a model that has less than 1% RMS error for fully rough flows. If C_{r1} is large enough that it does not affect the solution, the closure coefficients that provide the best fit are only a function of σ_k . Four points of interest in the design space may be worth considering to evaluate the overall performance of the model. Three of these values are values on the flat for $\sigma_k = 0.1$, $\sigma_k = 4$, and $\sigma_k = 0.75$. These constitute a small value for σ_k , a large value for σ_k , and a convenient point in between that minimizes error while making the model nearly independent of the core Reynolds number. The final point of interest is the minimum % RMS error point in Fig. 8.3 with $\sigma_k = 0.1$ and $C_{r1} = 0.03$. Results for each of these points are discussed here and the model results are compared to results from the Wilcox 1998 and 2006 models.

II. Model Results at Points of Interest**A. Low Value on the Flat: $\sigma_k = 0.1$**

Setting $\sigma_k = 0.1$ and evaluating the other closure coefficients from the flat gives model results with a % RMS error of 0.158 over the range of fully rough flows studied. Figure 9.1 shows the friction factor results for these closure coefficient values. Figures 9.2 and 9.3 show the nondimensional velocity and eddy-viscosity profiles for $k_s^+ = 1000$ over a range of relative roughness values. The empirical correlations from Chapter 7 are also included for comparison. Note that although the friction factor results match the Colebrook equation extremely well and are easily within the scatter of the experimental data, the velocity profiles do not match the law of the wall and the eddy-viscosity profiles deviate significantly from the

empirical relations far from the wall. These characteristics of the model are typical for results computed on the flat as will be shown in the following subsections. The characteristics displayed in Figs. 8.12 and 8.13 reveal significant traits of the model and will be discussed once other plots computed from the flat have all been presented.

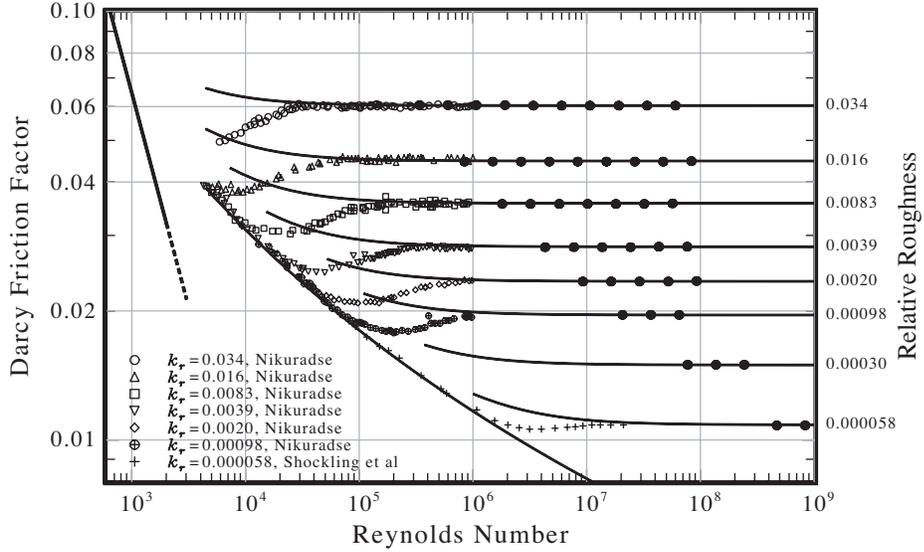


Fig. 9.1 Friction factor results for the Phillips k - λ model on the flat with $\sigma_k = 0.1$.

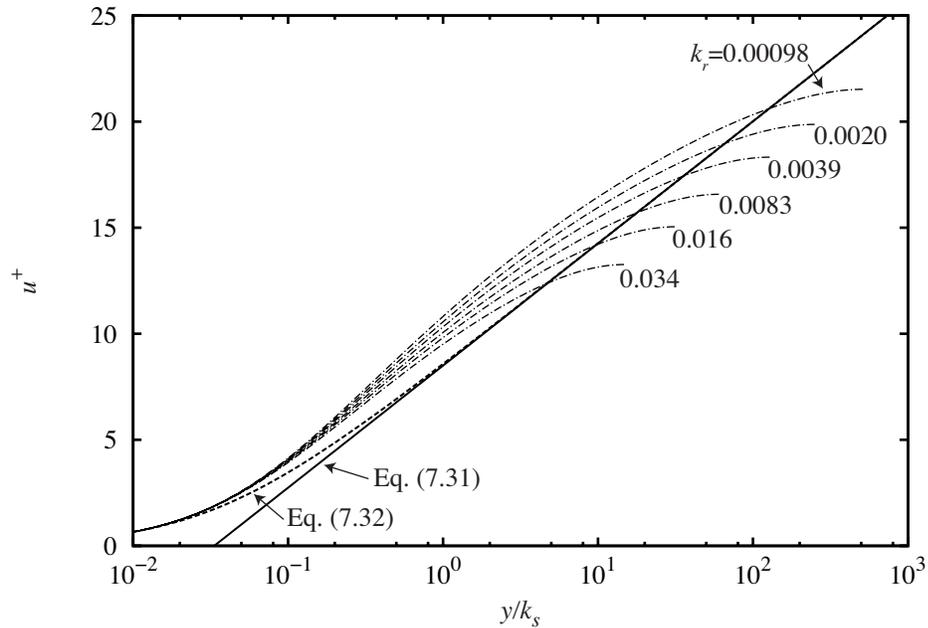


Fig. 9.2 Velocity results for the Phillips k - λ model on the flat with $\sigma_k = 0.1$ and $k_s^+ = 1000$.

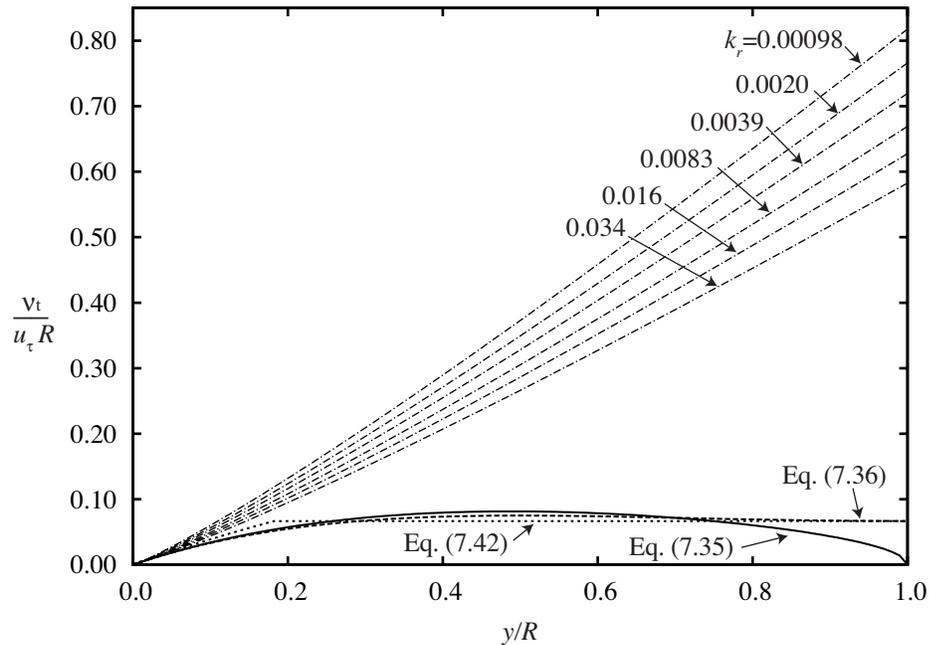


Fig. 9.3 Eddy-viscosity results for the Phillips k - λ model on the flat with $\sigma_k = 0.1$ and $k_s^+ = 1000$.

B. High Value on the Flat: $\sigma_k = 4$

Setting $\sigma_k = 4.00$ (which is likely larger than it will need to be to allow the model to transition to smoother walls) and evaluating the other closure coefficients from the flat gives friction factor results with a % RMS error of 0.670. Figure 9.4 shows the friction factor results along with published experimental data. Note that the friction factor results deviate noticeably more from the Colebrook equation than those for $\sigma_k = 0.1$. However, they could reasonably be considered within the scatter of the experimental data. Figures 9.5 and 9.6 show the velocity and eddy-viscosity profile results for $k_s^+ = 1000$. Again, note that the velocity profiles do not match the law of the wall, and the eddy-viscosity results do not match the empirical relations far from the wall.

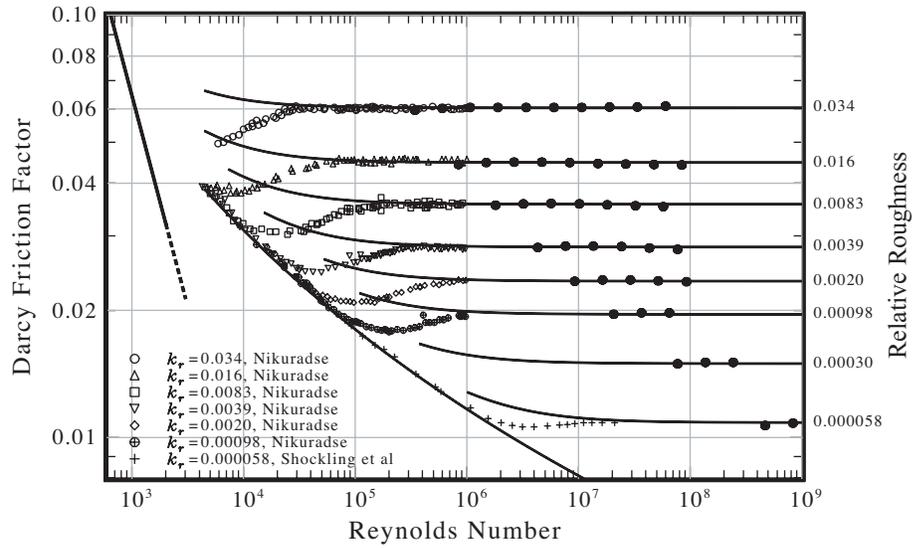


Fig. 9.4 Friction factor results for the Phillips k - λ model on the flat with $\sigma_k = 4$.

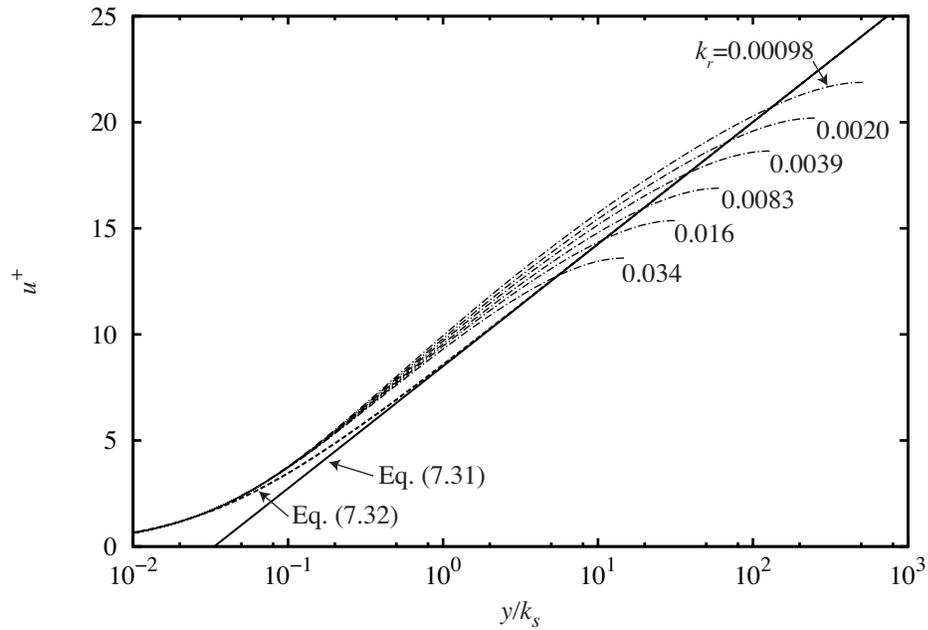


Fig. 9.5 Velocity results for the Phillips k - λ model on the flat with $\sigma_k = 4$ and $k_s^+ = 1000$.

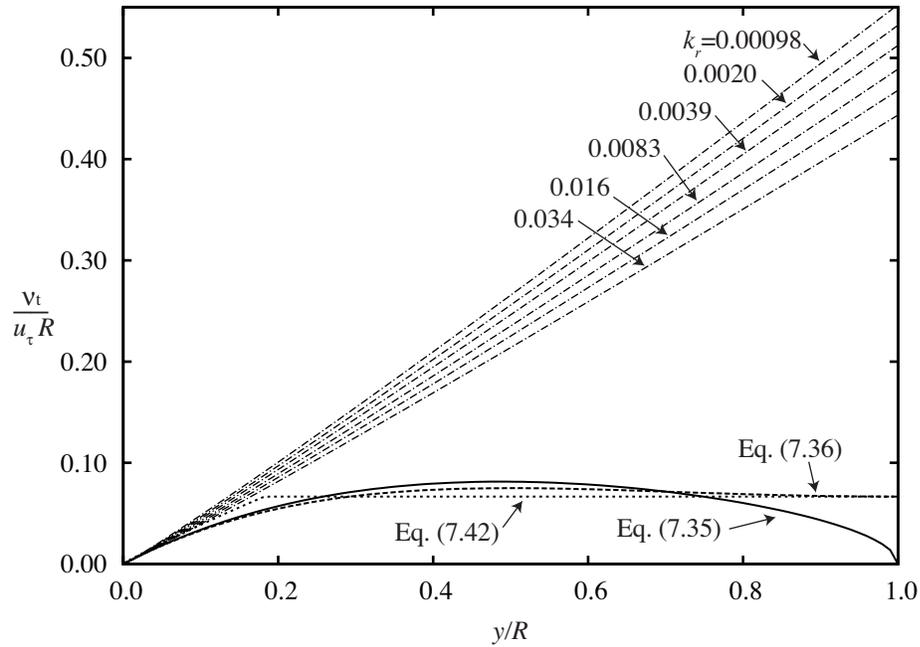


Fig. 9.6 Eddy-viscosity results for the Phillips $k\text{-}\lambda$ model on the flat with $\sigma_k = 4$ and $k_s^+ = 1000$.

C. Point of Interest on the Flat: $\sigma_k = 0.75$

Setting $\sigma_k = 0.75$ and evaluating the other closure coefficients from the flat gives friction factor results with a % RMS error of 0.186. The friction factor results are shown in Fig. 9.7. Again, the model results match the fully rough limit of the experimental data within the accuracy of the experimental data. Figures 9.8 and 9.9 show the results for the velocity and eddy-viscosity profiles.

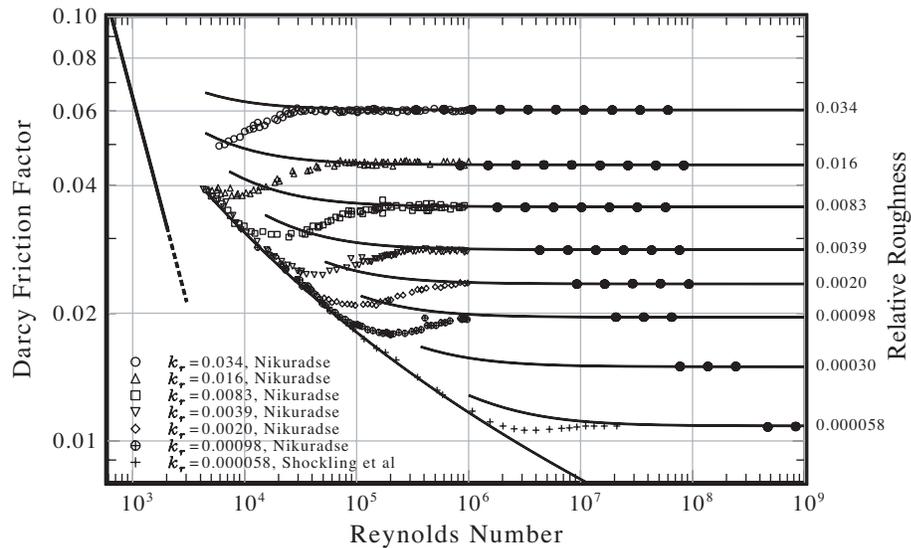


Fig. 9.7 Friction factor results for the Phillips $k\text{-}\lambda$ model on the flat with $\sigma_k = 0.75$.

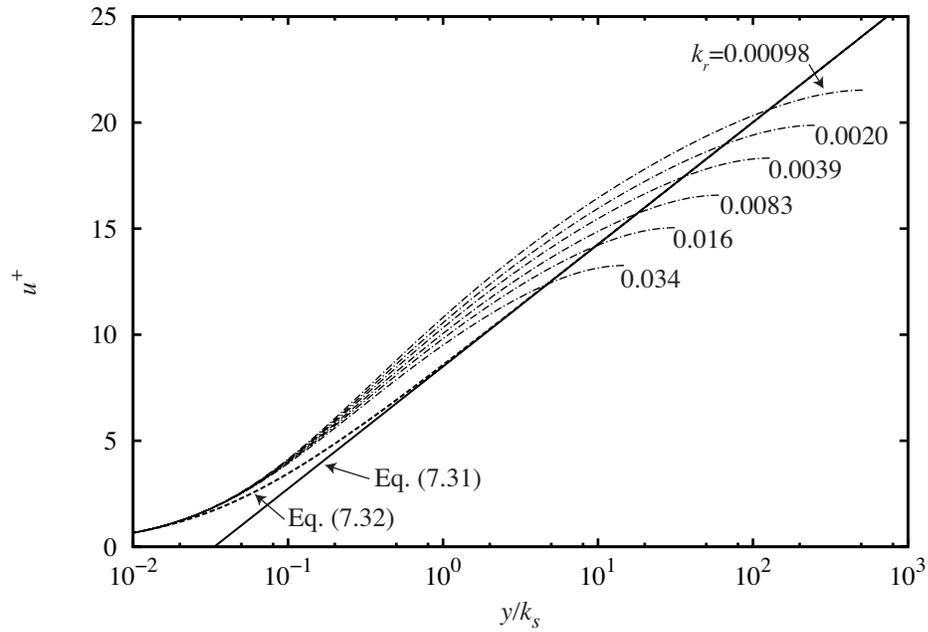


Fig. 9.8 Velocity results for the Phillips k - λ model on the flat with $\sigma_k = 0.75$ and $k_s^+ = 1000$.

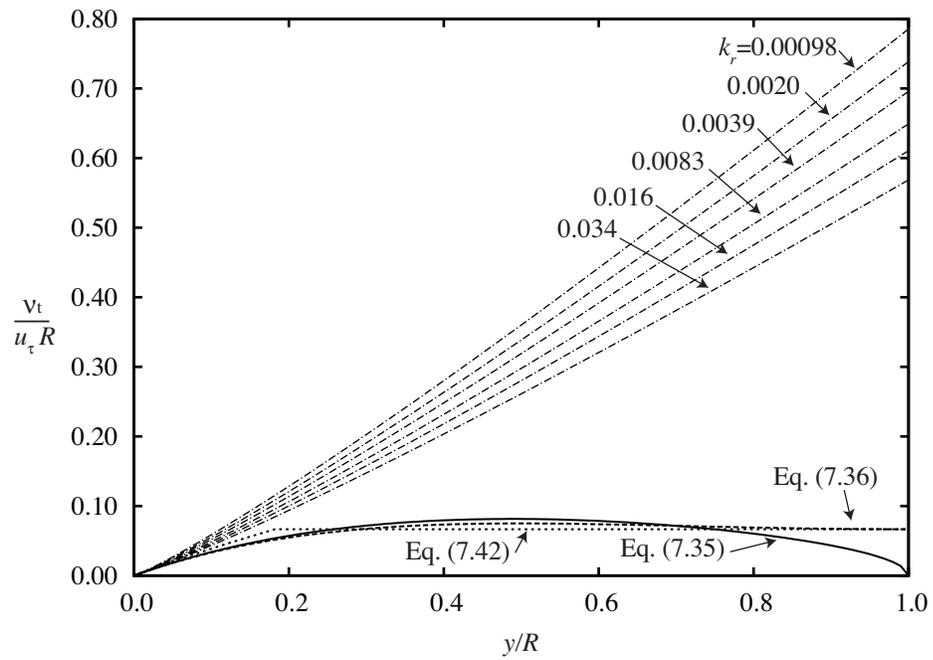


Fig. 9.9 Eddy-viscosity results for the Phillips k - λ model on the flat with $\sigma_k = 0.75$ and $k_s^+ = 1000$.

The results of the model with coefficients computed from the flat reveal some important characteristics of the model. Recall that the flat portion of the optimal design space is a region where the solution is independent of C_{r1} . If C_{r1} is large enough, the mean vortex wavelength is nearly a linear function of the distance from the wall, which results in a mean vortex wavelength distribution that does not satisfy the centerline symmetry boundary condition. Although this can be seen from Eq. (8.25), the fact that it did not satisfy the centerline symmetry boundary condition was not realized until the results for the eddy-viscosity profiles were considered. The fact that the mean vortex wavelength has a finite slope at the centerline causes the eddy-viscosity profile to have a finite slope at the centerline, which is obviously not correct. At the centerline, all the turbulence properties must satisfy the symmetry boundary condition which, in mathematical terms, means that the gradient at the centerline must be zero.

The fact that the velocity profiles on the flat do not match the law of the wall is very apparent in Figs. 9.2, 9.5, and 9.8. To understand why, recall that Eq. (7.32) was derived making two major assumptions. First, it was assumed that the eddy viscosity is a linear function of the distance from the wall, and second, it was assumed that y^+/R_τ is small compared to unity. Even though these assumptions do not match reality far from the wall, somehow the resulting equation for the velocity profile matches experimental data over the entire flow cross section as can be seen in Fig. 7.4. This fact suggests that the effects of one assumption must be cancelling the effects of the other assumption in the solution far from the wall. Therefore, if the eddy viscosity were assumed linear but the assumption that y^+/R_τ is small compared to unity were neglected, the closed-form solution for the velocity profile would likely be in significant error far from the wall. Such an assumption is similar to the results of this model with coefficients computed from the flat. Note that the eddy viscosity of the model is nearly linear over the entire flow cross section and that it matches the near-wall eddy viscosity of the empirical relations very well. However, in the numerical integration, the y^+/R_τ term is included, and the solution gives a velocity profile that does not match Eq. (7.32) in the region where y^+/R_τ is significant compared to unity.

These results show that although the results from the flat portion of the optimal design space match the friction factor very well, this region creates a non-physical mean vortex wavelength profile which in turn

causes the eddy-viscosity and velocity profiles to deviate significantly from experimental data and empirical relations. Therefore, a point in the optimum design space where C_{r1} is small enough to cause the mean vortex wavelength profile to satisfy the centerline symmetry boundary condition is worth considering.

D. Point of Minimum % RMS Error

The point with the smallest % RMS error for the friction factor shown in Fig. 8.3 is that for $\sigma_k = 0.1$ and $C_{r1} = 0.03$. This point is significant because it is not on the flat portion of the optimal design space, and therefore, the value for C_{r1} is small enough to affect the solution. The optimal closure coefficients for this point to three significant figures are

$$\begin{aligned} \sigma_k = 0.1, \quad C_{r1} = 0.03, \quad a_{r1} = 7.54E - 03, \quad C_{r2} = 4.96E - 03, \quad a_{r2} = 6.51E - 03, \\ C_{r3} = 1.70E - 01, \quad a_{r3} = 2.17E - 03, \quad a_{r4} = 1.12 \end{aligned} \tag{9.1}$$

Figure 9.10 shows the friction factor results for high roughness Reynolds numbers. The % RMS error for this set of closure coefficients is 0.129. Note that the results match the Colebrook equation extremely well.

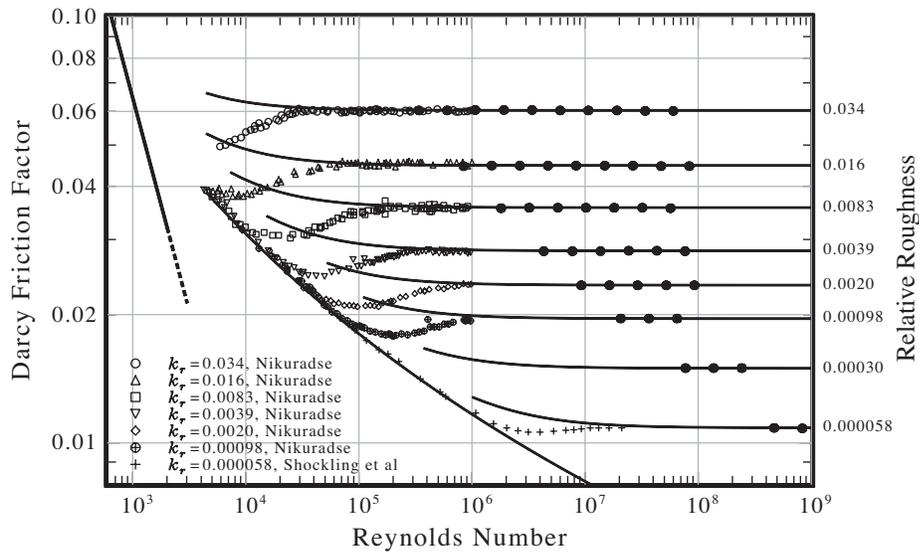


Fig. 9.10 Friction factor results for the Phillips $k-\lambda$ model at the point of minimum % RMS error.

Figures 9.11 and 9.12 show the velocity and eddy-viscosity profiles for the model with $k_s^+ = 1000$ over a range of values for the relative roughness. Note that the velocity profiles appear to match the law of the

wall marginally better than the results for the velocity profiles on the flat, with the greatest improvement being apparent near the centerline of the pipe. Also note from Fig. 9.12 that the eddy-viscosity profiles with this set of closure coefficients are much closer to the empirical relations than the profiles resulting from closure coefficients on the flat.

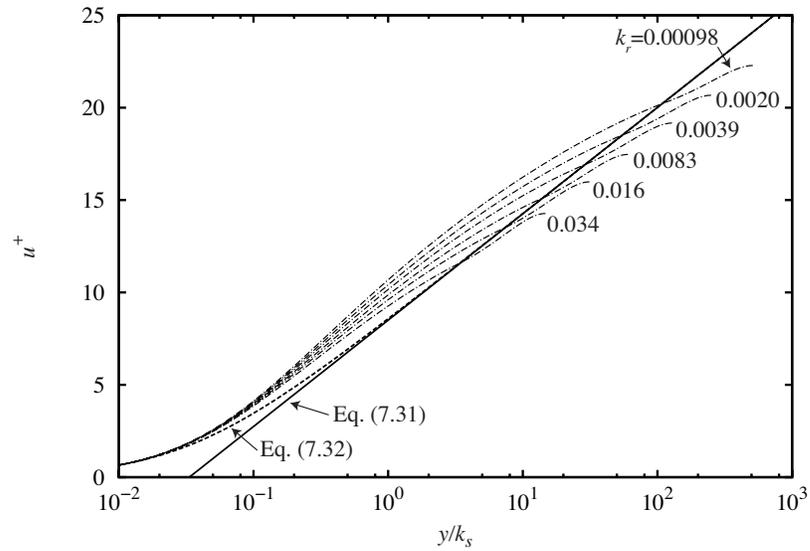


Fig. 9.11 Velocity results for the Phillips k - λ model at the point of minimum % RMS error with $k_s^+ = 1000$.

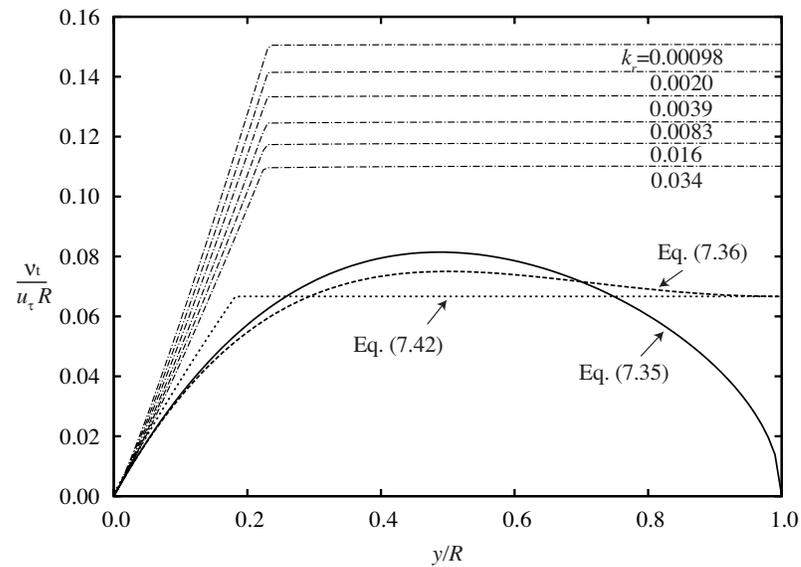


Fig. 9.12 Eddy-viscosity results for the Phillips k - λ model at the point of minimum % RMS error with $k_s^+ = 1000$.

Of particular note is that the region of the velocity profiles that deviates the most from the law of the wall corresponds to the region of the eddy-viscosity profile of $0.02 < y/R < 0.2$. Note that the model predicts an eddy-viscosity distribution in this region that is nearly linear while the experimental data shown in Fig. 7.5 displays a more parabolic distribution. Because the velocity distribution is a function of the eddy-viscosity distribution, if the law of the wall is to be correctly predicted, the eddy viscosity must also be accurately predicted in this region. Recall that the mean vortex wavelength function was based on matching the eddy-viscosity distribution to the approximation of Kays and Crawford in Eq. (7.42). Because the Kays and Crawford approximation is nearly linear, the resulting model has near-linear mean-vortex-wavelength and eddy-viscosity distributions in this region. Because the eddy viscosity is directly proportional to the mean vortex wavelength, if the law of the wall is to be obtained, the fundamental equation for the mean vortex wavelength given in Eq. (7.59) must be altered to a more general profile that would allow for a parabolic eddy-viscosity distribution. This appears to be the fundamental limitation of the model closure.

Near the center of the pipe where the velocity gradient is small, the dependence of the velocity profile on the eddy viscosity becomes weaker. In this region, there appears to be significant scatter in the data for the eddy viscosity as can be seen in Fig. 7.5. Therefore, it is expected that the exact distribution of the eddy viscosity and mean vortex wavelength in the region $y/R > 0.5$ is less important than that in the region $0.02 < y/R < 0.2$.

Given the current limitations of this turbulence model, it appears that the set of closure coefficients given in Eq. (9.1) are the best set of closure coefficients for this model. Significant improvements could be made to the model if the mean vortex wavelength equation were altered. However, without that flexibility, it is unlikely that any other set of closure coefficients would significantly improve the results of the model.

III. Comparison to Other Models

A. Wilcox k - ω Model

Traditionally, the k - ω model has been accepted as the model most capable of modeling rough-wall effects without implementing wall functions. The effects of surface roughness are commonly incorporated into the k - ω model by simply altering the surface boundary condition on ω . For example, Wilcox [35] suggests using the relation for ω^+ at a rough wall of

$$\omega^+ \Big|_{y^+=0} = \begin{cases} \left(\frac{200}{k_s^+} \right)^2, & k_s^+ \leq 5 \\ \frac{100}{k_s^+} + \left[\left(\frac{200}{k_s^+} \right)^2 - \frac{100}{k_s^+} \right] \exp(5 - k_s^+), & k_s^+ > 5 \end{cases} \quad (9.2)$$

The Wilcox 1998 k - ω model is likely the most widely used model for rough walls, and there is perhaps no better way to evaluate the Wilcox 1998 model than by using code distributed by Wilcox. Along with his book on turbulence modeling, Wilcox distributes a CD with a number of computer programs capable of running his model for various flows. Among these programs is a fully-developed-pipe-flow code which allows the user to model roughness effects by specifying ω^+ at the wall. This code was used to evaluate the Wilcox 1998 model in comparison to the experimental cases of Nikuradse. For each k_r value, cases were run starting at $R_\tau = 1.0$ and incrementing R_τ until the computer code would no longer converge. For each case, ω^+ at the wall was evaluated using Eq. (9.2) as suggested by Wilcox. It was found that grid-converged results for the code could be obtained using 801 nodes, and therefore, each case was run using this grid size. Figure 9.13 shows the results of this study. The results of this model appear to be nearly within the scatter of the experimental data for high Reynolds numbers. However, the results exhibit a trend with a positive slope that causes the deviation from the Colebrook equation to increase with increasing Reynolds number. Additionally, the code did not converge for higher Reynolds numbers than those shown in the figure.

In 2006 Wilcox published a revised model which has not as yet been as widely implemented as his 1998 model [35]. Using his code for his 2006 model, the computations explained above were repeated.

Figure 9.14 shows the results of this model compared to experimental data. Note that this revised model deviates significantly from the Colebrook equation, and the deviation increases with increasing Reynolds number. The code did not converge for Reynolds numbers higher than those shown in the figure.

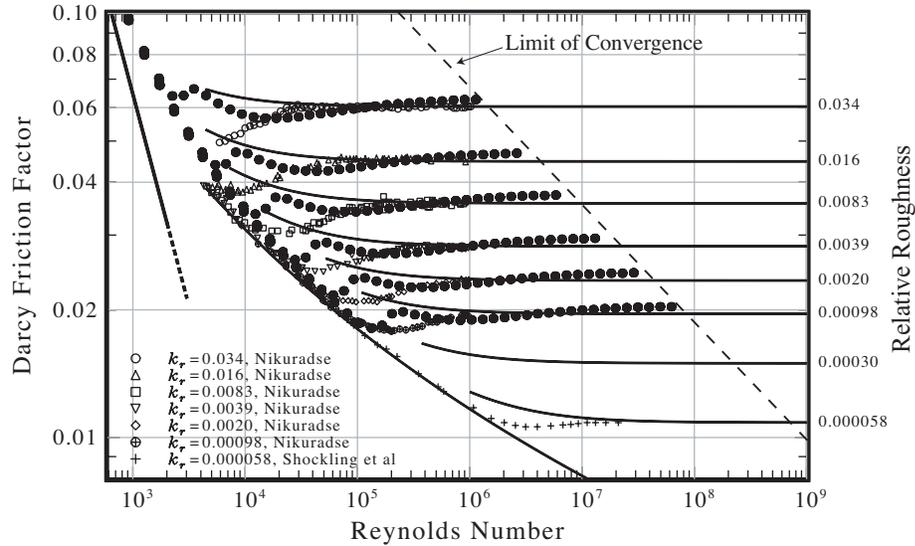


Fig. 9.13 Friction factor results for the Wilcox 1998 $k-\omega$ model.

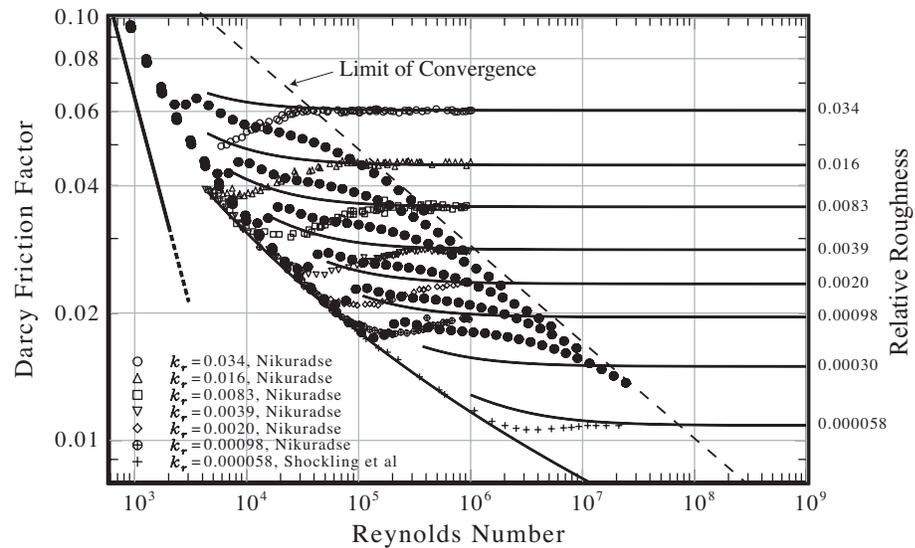


Fig. 9.14 Friction factor results for the Wilcox 2006 $k-\omega$ model.

Comparing the results of the Wilcox 1998 and 2006 $k-\omega$ models seen in Figs. 9.13 and 9.14 to the results of the Phillips $k-\lambda$ model seen in Figs. 9.1, 9.4, 9.7, and 9.10 shows that the Phillips $k-\lambda$ model provides a significant improvement over traditional methods for modeling the bulk-flow properties of fully rough pipe flow. Figure 9.15 shows the results of the model predictions for the Nikuradse number as a function of roughness Reynolds number. For these results, the optimal model constants given in Eq. (9.1) were used. The plot also includes the data sets of Nikuradse and Shockling, et al. Note that the Phillips $k-\lambda$ model is within the scatter of the experimental data for fully rough flows, whereas the $k-\omega$ models deviate significantly from the Colebrook equation and the experimental data.

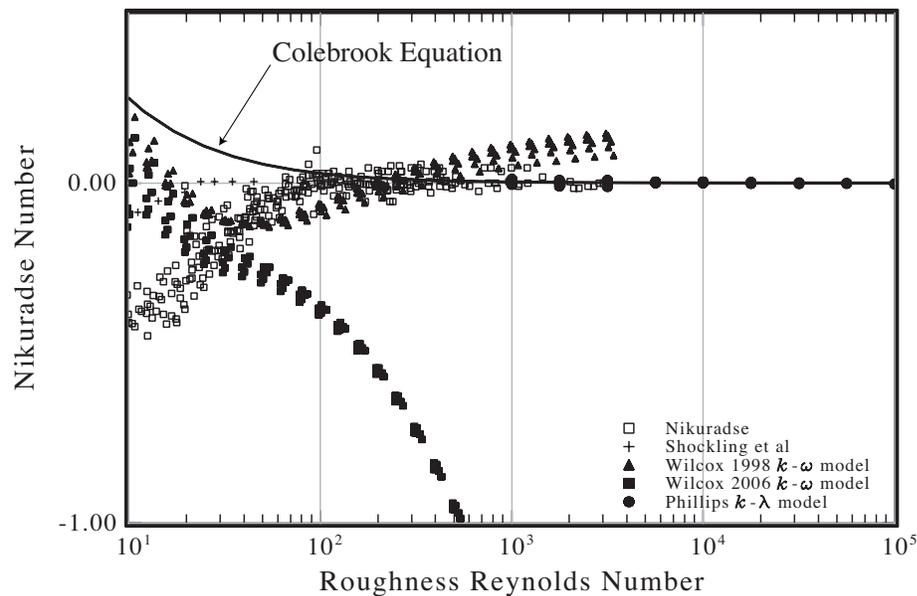


Fig. 9.15 Nikuradse number results of the Phillips $k-\lambda$ model with optimum model constants.

B. Eddy-Viscosity Models

Four eddy-viscosity models for fully rough pipe flow were given in Chapter 7. Equation (7.35) is a model that fits Nikuradse's eddy-viscosity estimates, Eq. (7.36) is a model that fits the estimates of Reichardt, Eq. (7.42) is a model suggested by Kays and Crawford, and Eq. (7.43) is a model that can be developed from the law of the wall. It is insightful to observe how well each of these eddy-viscosity profiles compares to the Phillips $k-\lambda$ model presented here. Because the deviation for each of these eddy-viscosity profiles as a function of roughness is almost imperceptible when shown on a plot similar to Fig. 7.5, the eddy-viscosity

profiles are not included here. Figures 9.16 – 9.19 show the velocity profiles produced by each of the models for a range of k_r values where $k_s^+ = 1000$. Each of these cases was run with the traditional values of $\kappa = 0.40$ and $\gamma = 0.0334$. The code to run each of these cases is included in Appendix O.

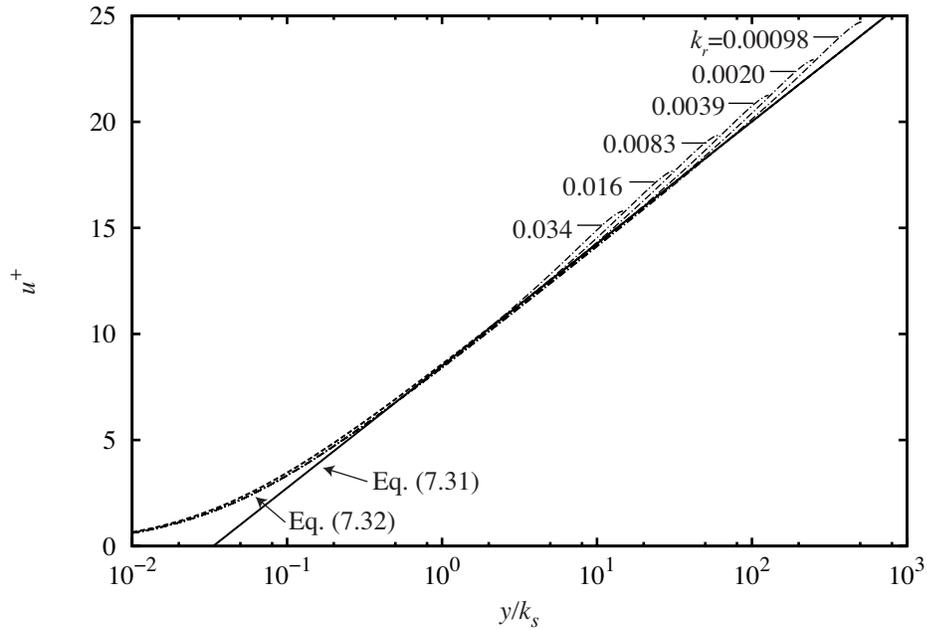


Fig. 9.16 Velocity results for the eddy-viscosity model given in Eq. (7.35) with $k_s^+ = 1000$.

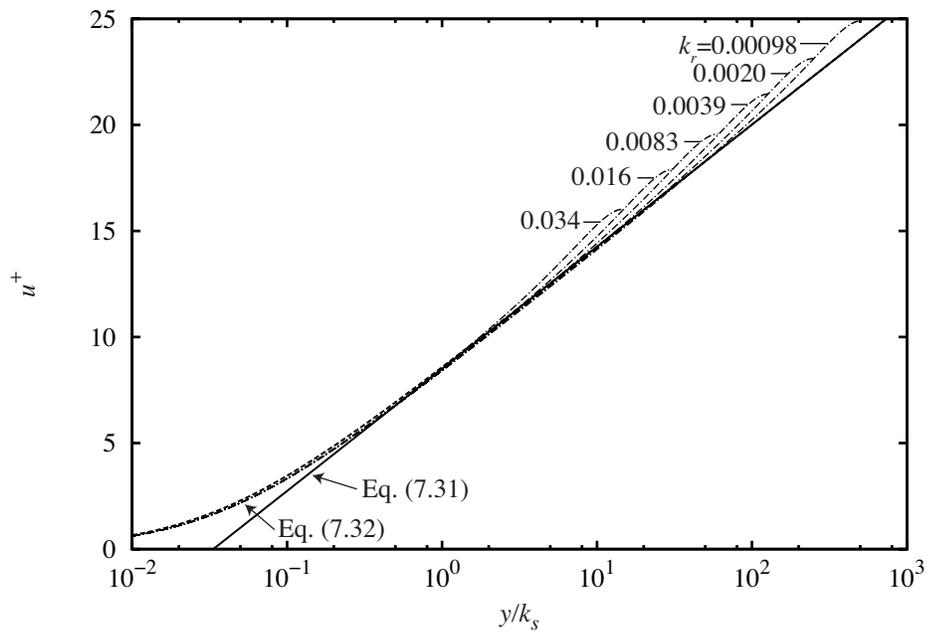


Fig. 9.17 Velocity results for the eddy-viscosity model given in Eq. (7.36) with $k_s^+ = 1000$.

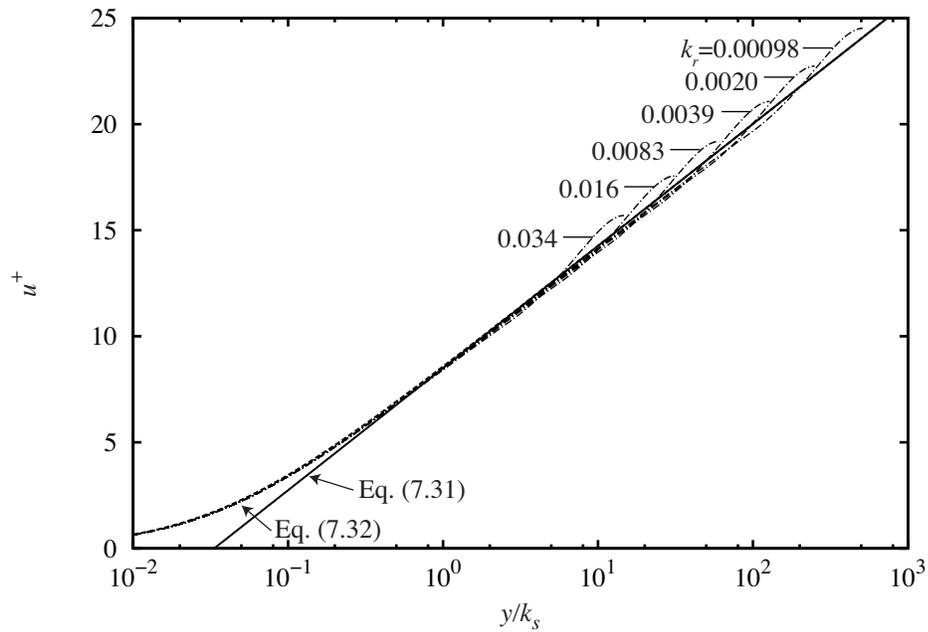


Fig. 9.18 Velocity results for the eddy-viscosity model given in Eq. (7.42) with $k_s^+ = 1000$.

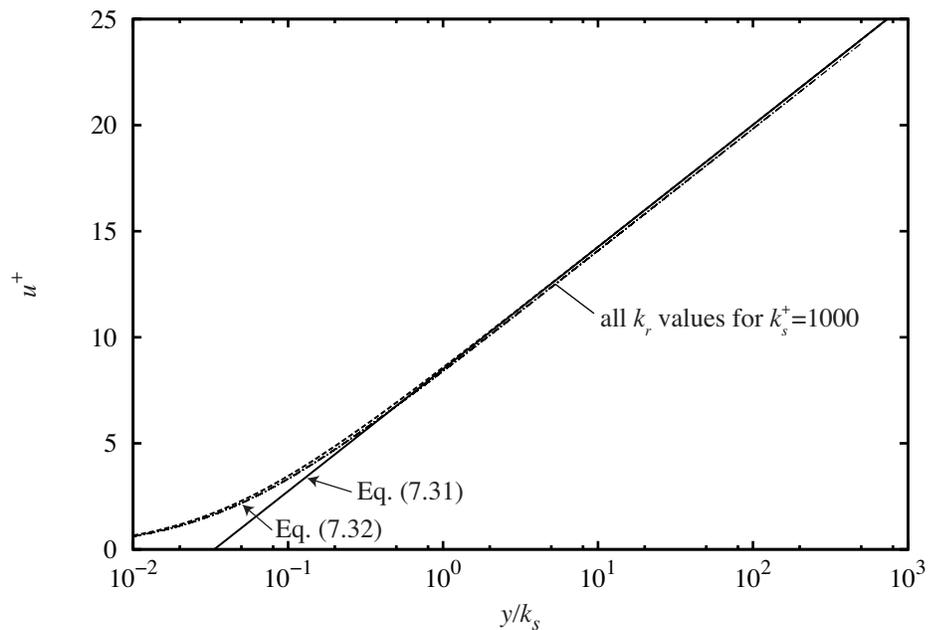


Fig. 9.19 Velocity results for the eddy-viscosity model given in Eq. (7.43) with $k_s^+ = 1000$.

Because each of these models was developed from velocity profile measurements, each model also reconstructs the velocity profile fairly well. Note particularly the results in Fig. 9.19. As the value for k_s^+ is increased, the model results approach the law-of-the-wall equations also included in the plot. This behavior

is not surprising because the model given in Eq. (7.43) was derived directly from the law-of-the-wall velocity profile with the fully rough flow assumption. Each of these velocity profiles from the eddy-viscosity models appears to be a significant improvement over the velocity profiles predicted by the Phillips k - λ model shown in Fig. 9.11.

Although each of these models reconstructs the velocity profile fairly well, the relationship between friction factor and Reynolds number is not extremely accurate. Figures 9.20 – 9.23 show the friction factor results for each of these models. Note that the models based on Nikuradse’s and Reichardt’s approximations for the eddy viscosity given in Eqs. (7.35) and (7.36) result in significant errors in the friction factor as a function of Reynolds number. These yield % RMS errors of 5.46 and 7.81 respectively. The model based on the work of Kays and Crawford given in Eq. (7.42) gives reasonable results with a % RMS error of 1.81, but is noticeably less accurate than the Phillips k - λ model. The model derived from the law of the wall with results shown in Fig. 9.23 has a % RMS error of 0.826. This model is a significant improvement over the other three models, but has a % RMS error that is six times that of the Phillips k - λ model. Therefore, the Phillips k - λ model included in this work provides a significant improvement over other models for predicting the relationship between the Reynolds number and friction factor for fully rough pipe flow.

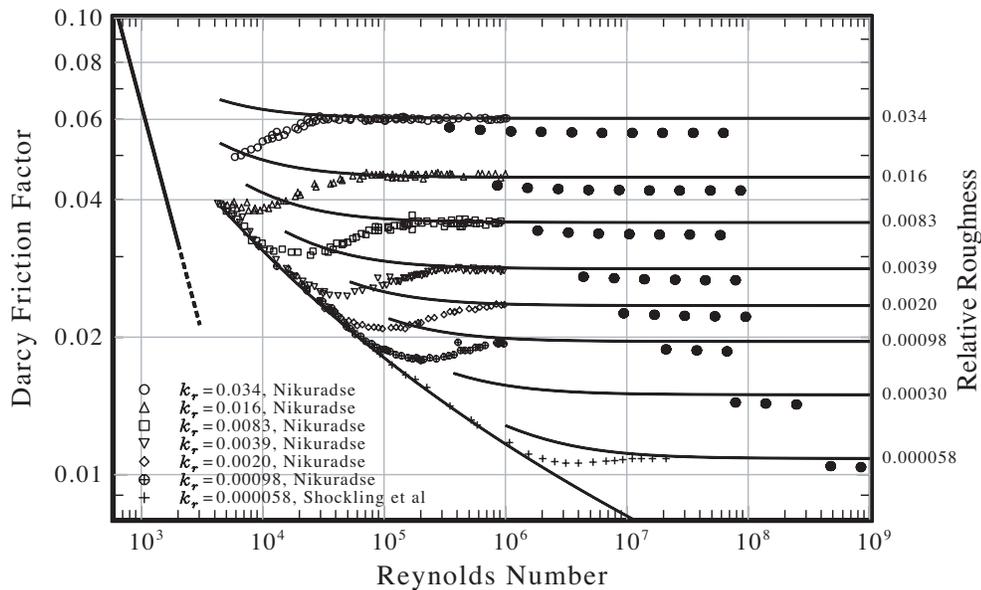


Fig. 9.20 Friction factor results for the eddy-viscosity model given in Eq. (7.35).

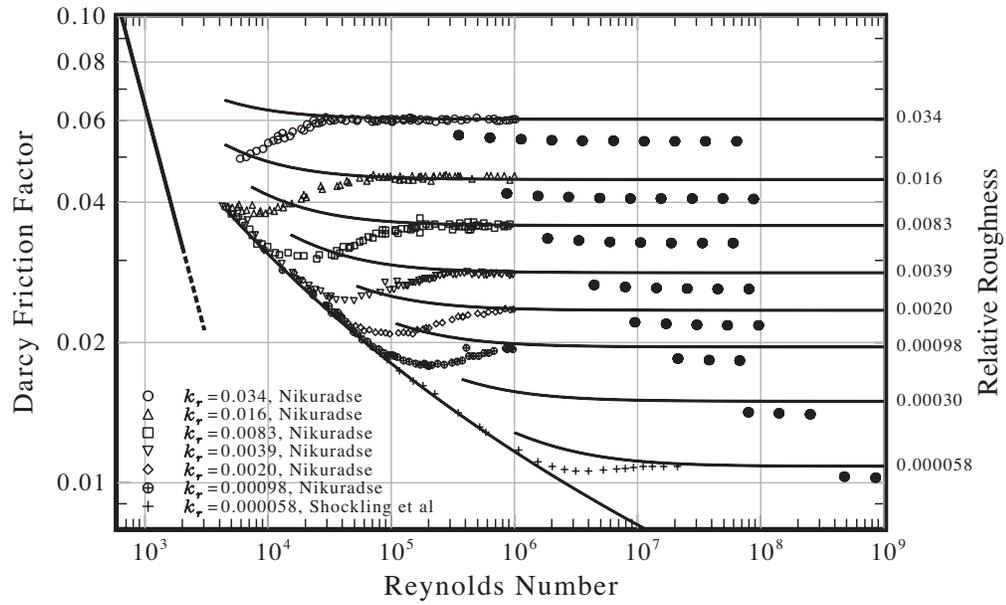


Fig. 9.21 Friction factor results for the eddy-viscosity model given in Eq. (7.36).

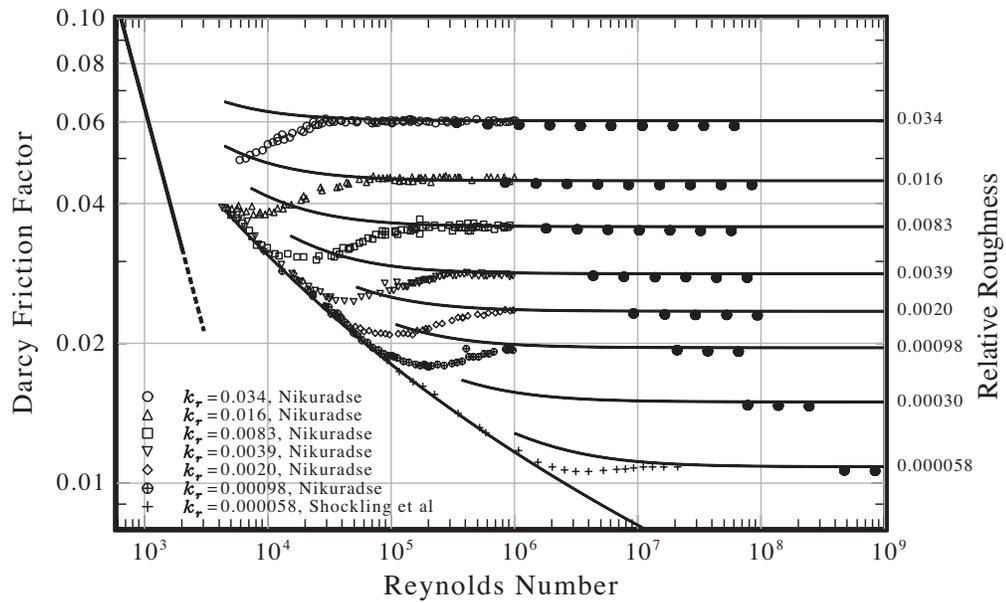


Fig. 9.22 Friction factor results for the eddy-viscosity model given in Eq. (7.42).

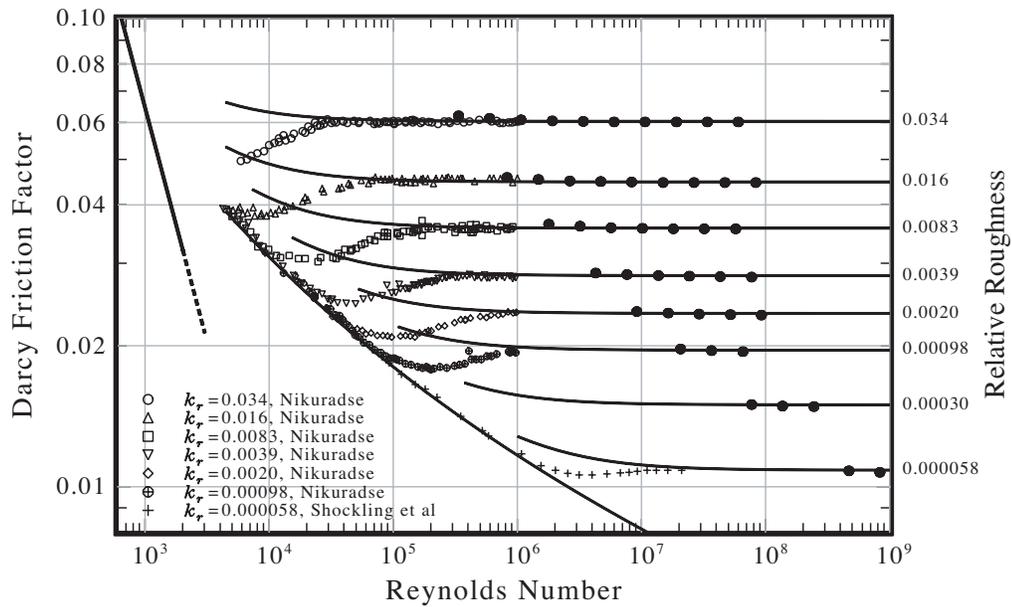


Fig. 9.23 Friction factor results for the eddy-viscosity model given in Eq. (7.43).

IV. Summary and Conclusions

The focus of this work has been to evaluate the performance of an alternate energy-vorticity turbulence model. One significant benefit to the current turbulence model is that it is based on modeling the vorticity of the flow. Many RANS-based turbulence models in use today model the dissipation of turbulent kinetic energy as a transport property. This is simply not physically correct because dissipation is not a transport property. Furthermore, it has been shown that traditional smooth-wall boundary conditions for dissipation-based turbulence models are incorrect. The correct smooth-wall boundary conditions for turbulence models were presented in Chapter 5, and are boundary conditions that force both k and its first derivative to zero at a smooth wall. The behavior of the present turbulence model near a smooth wall has been considered, and justification for developing a model for rough walls was given in Chapter 6. The development of the Phillips k - λ model for fully rough pipe flow was presented in detail in Chapter 7 and includes an algebraic relation for the mean vortex wavelength as a function of distance from the wall. A gradient-based optimization routine was presented that was used to evaluate optimal values for the closure coefficients. Finally, the results of the model have been presented and compared to experimental results, empirical correlations, and other rough-wall models.

The resulting model provides a very accurate method for predicting the bulk flow properties of fully rough pipe flow. The % RMS error of the model is less than 1% over a range of values for C_{r1} and σ_k as shown in Fig. 8.10. The model is significantly more accurate for predicting the relationship between Reynolds number and friction coefficient for fully rough pipe flow than the Wilcox 1998 and 2006 $k-\omega$ models as well as the eddy-viscosity models of Nikurase, Reichardt, and Kays and Crawford. However, the resulting velocity profiles of the Phillips $k-\lambda$ model do not match the expected law of the wall for fully rough pipe flow. One result of the model that will likely lead to future research is the fact that if C_{r1} is large enough that it does not affect the solution, the mean vortex wavelength equation does not satisfy the centerline symmetry boundary condition. This is a fundamental limitation in the model which must be addressed in a future revision of the model. The fact that the mean vortex wavelength does not satisfy the centerline boundary condition in this situation leads to a result for the eddy viscosity that also does not satisfy this centerline boundary condition. This is clearly a fundamental inconsistency with the model and should be addressed in future work.

The current limitations of the model suggest areas of future work that could produce promising results. First, a fully rough model must be developed which has the flexibility of matching the parabolic nature of the eddy-viscosity distribution in the law-of-the-wall region of the flow. This will most likely be achieved by developing a closing equation for the mean vortex wavelength that, when coupled with the turbulent-kinetic-energy transport equation, results in a parabolic distribution for the eddy viscosity. This can be accomplished using a method similar to that presented in Chapter 8, but using the distribution of Reichardt [95] or a distribution that matches the data of Nikuradse [59] rather than the approximation suggested by Kays and Crawford [96]. Once a relation for the mean vortex wavelength has been developed, the optimization routine included in this work can be used to evaluate optimal closure coefficients for the model. The resulting model will likely match the velocity and eddy-viscosity profiles better than the model presented here, and may even match the friction factor results given by the Colebrook equation better than the current model.

Once a robust model for fully rough pipe flow has been obtained, a transition model could be developed that allows the model to predict the flow properties at lower roughness Reynolds numbers. In the limit as the roughness approaches zero, this model should match results for hydraulically smooth walls. It is unlikely that a model will need to be developed for perfectly smooth walls, as no experimental data exists for such walls, and such a wall is currently impossible to manufacture.

Once a complete turbulence model for transitionally rough pipe flows has been developed, the model characteristics can be extended to channel and other boundary layer flows. It is possible that such an approach to developing future turbulence models can result in models that are more robust and accurate over a wide range of flow scenarios.

REFERENCES

- [1] Navier, C. L. M. H., “Mémoire sur les Loistu Movement des Fluides,” *Mém. de l’ Acad. R. Sci. Paris*, Vol. 6, 1823, pp. 389–416.
- [2] Stokes, G. G., “On the Theories of Internal Friction of Fluids in Motion,” *Transactions of the Cambridge Philosophical Society*, Vol. 8, 1845, pp. 287–305.
- [3] Hamilton, W. R., “On Quaternions: Or a New System of Imaginaries in Algebra,” *Philosophical Magazine*, Vol. 25, No. 3, 1844, pp. 489–495.
- [4] Hamilton, W. R., *Lectures on Quaternions*, Hodges and Smith, Dublin, 1853, pp. 1–736.
- [5] Hamilton, W. R., *Elements of Quaternions*, Longmans, Green and Co., London, 1866, pp. 1–762.
- [6] Boyer, C. B., and Merzback, U. C., *A History of Mathematics*, 2nd ed., Wiley, New York, 1989, pp. 656–762.
- [7] Moin, P., and Mahesh, K., “Direct Numerical Simulation – A Tool in Turbulence Research,” *Annual Review of Fluid Mechanics*, Vol. 30, 1998, pp. 539–578.
- [8] Goldstein, S., “Fluid Mechanics in the First Half of This Century,” *Annual Review of Fluid Mechanics*, Vol. 1, 1969, pp. 1–29.
- [9] Bradshaw, P., “Turbulence: The Chief Outstanding Difficulty of Our Subject,” *Experiments in Fluids*, Vol. 16, 1994, pp. 203–216.
- [10] Frisch, U., and Orszag, S., “Turbulence: Challenges for Theory and Experiment,” *Physics Today*, Vol. 43, 1990, pp. 24–32.
- [11] Holmes P. J., Berkooz G., and Lumley J. L., “Turbulence,” *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, Cambridge University Press, 1996, pp. 3–6.
- [12] Hinze, J. O., “General Introduction and Concepts,” *Turbulence*, Second Edition, McGraw-Hill, New York, 1975, pp. 1–2.
- [13] Phillips, W. F., “Turbulent Flow in Newtonian Fluids,” *Aerodynamics of Flight DRAFT*, Wiley, Hoboken, New Jersey, 2008, pp. 1–51.
- [14] Wilcox D. C., “Reynolds Averaging,” *Turbulence Modeling for CFD*, Third Edition, DCW Industries, Inc., La Cañada, California, 2006, pp. 34–38.
- [15] Jones, W. P. and Launder, B. E., “The Prediction of Laminarization with a Two-Equation Model of Turbulence,” *International Journal of Heat and Mass Transfer*, Vol. 15, 1972, pp. 301–314.
- [16] Kolmogorov, A. N., “Equations of Turbulent Motion of an Incompressible Fluid,” *Izvestia Academy of Sciences, USSR; Physics*, Vol. 6, Nos. 1 and 2, 1942, pp. 56–58.
- [17] Spalart, P. R., and Allmaras, S. R., “A One-Equation Turbulence Model for Aerodynamic Flows,” AIAA 92-439, 1992.
- [18] Baldwin, B. S., and Barth, T. J., “A One-Equation Turbulence Transport Model for High Reynolds Number Wall-Bounded Flows,” NASA TM-102847, 1990.

- [19] Durbin, P. A., "Near-Wall Turbulence Closure Modeling Without Damping Functions," *Theoretical and Computational Fluid Dynamics*, Vol. 3, No. 1, 1991, pp. 1–13.
- [20] Reynolds, O., "On the Dynamical Theory of Incompressible Viscous Fluids and the Determination of the Criterion," *Philosophical Transactions of the Royal Society of London, Series A*, Vol. 186, 1895, p. 123–164.
- [21] Boussinesq, J., "Théore de l'Écoulement Tourbillant," *Mém. Présentés par Divers Savants Acad. Sic. Inst. Fr.*, Vol. 23, 1877, pp. 46–50.
- [22] Prandtl, L., "Über die ausgebildete Turbulenz," *Z. angew. Math. u. Mech.*, Vol. 5, 1925, pp. 136–139.
- [23] Prandtl, L., "Über ein neues Formelsystem für die ausgebildete Turbulenz," *Nacr. Akad. Wiss Göttingen, Math-Phys. Kl.*, 1945, pp. 6–19.
- [24] Emmons, H. W., "Shear Flow Turbulence," Proceedings of the 2nd U. S. Congress of Applied Mechanics, ASME, 1954.
- [25] Glushko, G., "Turbulent Boundary Layer on a Flat Plate in an Incompressible Fluid," *Izvestia Akademiya Nauk SSSR, Mekh.*, No. 4, 1965, p. 13–30.
- [26] Wolfshtein, M., "Convection Processes in Turbulent Impinging Jets," Imperial College, Heat Transfer Section Report SF/R/2, 1967.
- [27] Bradshaw, P., Ferriss, D. H., and Atwell, N. P., "Calculation of Boundary Layer Development Using the Turbulent Energy Equation," *Journal of Fluid Mechanics*, Vol. 28, Pt. 3, 1967, pp. 593–616.
- [28] Nee, V. W., and Kovaszny, L. S. G., "The Calculation of the Incompressible Turbulent Boundary Layer by a Simple Theory," *Physics of Fluids*, Vol. 12, 1968, pp. 473.
- [29] Sekundov, A. N., "Application of the Differential Equation for Turbulent Viscosity to the Analysis of Plane Non-Self-Similar Flows," *Akademiya Nauk SSSR, Izvestiia, Mekhanika Zhidkosti i Gaza*, 1971, pp. 114–127 (in Russian).
- [30] Menter, F. R., "Eddy Viscosity Transport Equations and Their Relation to the $k-\epsilon$ Model," *Journal of Fluids Engineering*, Volume 119, Issue 4, 1997, pp. 876–884.
- [31] Rotta, J. C., "Statistische Theorie nichthomogener Turbulenz," *Zeitschrift für Physik*, Vol. 129, 1951, pp. 547–572.
- [32] Rotta, J. C., "Über eine Methode zur Berechnung turbulenter Scherströmungen," Aerodynamische Versuchsanstalt Göttingen, Rep. 69 A 14, 1968.
- [33] Zeierman, S., and Wolfshtein, M., "Turbulent Time Scale for Turbulent Flow Calculations," *AIAA Journal*, Vol. 24, No. 10, 1986, pp. 1606–1610.
- [34] Speziale, C. G., Abid, R., and Anderson, E. C., "A Critical Evaluation of Two-Equation Models for Near Wall Turbulence," AIAA 90-1481, 1990.
- [35] Wilcox D. C., "One-Equation and Two-Equation Models," *Turbulence Modeling for CFD*, Third Edition, DCW Industries, Inc., La Cañada, California, 2006, pp. 107–238.

- [36] Taylor, G. I., "The Spectrum of Turbulence," *Proceedings of the Royal Society of London, Series A*, Vol. 164, 1938, pp. 476–490.
- [37] Hinze, J. O., "Isotropic Turbulence," *Turbulence*, Second Edition, McGraw-Hill, New York, 1975, pp. 165–204.
- [38] Launder, B. E., Morse, A., Rodi, W., and Spalding, D. B., "Prediction of Free Shear Flows – A Comparison of the Performance of Six Turbulence Models," *Proceedings of NASA Conference on Free Turbulent Shear Flows*, Langley Research Center, Hampton, VA, Vol. 1, 1972, pp. 361–426.
- [39] Wilcox, D. C., "Multiscale Model for Turbulent Flows," *AIAA Journal*, Vol. 26, No. 11, 1988, pp. 1311–1320.
- [40] Wilcox, D. C., "Reassessment of the Scale Determining Equation for Advanced Turbulence Models," *AIAA Journal*, Vol. 26, No. 11, 1988, pp. 1299–1310.
- [41] Menter, F. R., "Improved Two-Equation k - ω Turbulence Models for Aerodynamic Flows," NASA TM-103975, 1992.
- [42] Peng, S. H., Davidson, L., and Holmberg, S., "A Modified Low-Reynolds Number k - ω Model for Recirculating Flows," *Journal of Fluids Engineering*, Vol. 119, 1997, pp. 867–875.
- [43] Kok, J. C., "Resolving the Dependence on Freestream Values for the k - ω Turbulence Model," *AIAA Journal*, Vol. 38, No. 7, 2000, pp. 1292–1295.
- [44] Hellsten, A., "New Advanced k - ω Turbulence Model for High-Lift Aerodynamics," *AIAA Journal*, Vol. 43, No. 9, 2005, pp. 1857–1869.
- [45] Wilcox, D. C., "One-Equation and Two-Equation Models," *Turbulence Modeling for CFD*, Second Edition, DCW Industries, Inc., La Cañada, California, 1998, pp. 103–226.
- [46] Robinson, D. E., Harris, J. E., and Hassan, H. A., "Unified Turbulence Closure Model for Axisymmetric and Planar Free Shear Flows," *AIAA Journal*, Vol. 33, No. 12, 1995, pp. 2325–2331.
- [47] Robinson, D. E., and Hassan, H. A., "Further Development of the k - ζ (Enstrophy) Turbulence Closure Model," *AIAA Journal*, Vol. 36, No. 10, 1998, pp. 1825–1833.
- [48] Townsend, A. A., *The Structure of Turbulent Shear Flow*, Second Ed., Cambridge University Press, Cambridge, England, 1976.
- [49] Durbin, P. A., and Pettersson Reif, B. A., "Analytical Solutions to the k - ϵ Model," *Statistical Theory and Modeling for Turbulent Flows*, John Wiley & Sons Ltd, Chichester, England, 2001, pp. 120–123.
- [50] Clark, J. A., "A Study of Incompressible Turbulent Boundary Layers in Channel Flows," *Journal of Basic Engineering*, Vol. 90, 1968, pp. 455.
- [51] Laufer, J., "Some Recent Measurements in a Two-Dimensional Turbulent Channel," *Journal of Aeronautical Sciences*, Vol. 17, 1950, pp. 277–287.
- [52] Briggs, D. A., Ferziger, J. H., Koseff, J. R., and Monismith, S. G., "Entrainment in a Shear-Free Turbulent Mixing Layer," *Journal of Fluid Mechanics*, Vol. 310, 1996, pp. 215–241.
- [53] Klebanoff, P. S., "Characteristics of Turbulence in a Boundary Layer with Zero Pressure Gradient," NACA TN-1247, 1955.

- [54] P. Bradshaw, "The turbulence structure of equilibrium boundary layers," *Journal of Fluid Mechanics*, Vol. 29, 1967, pp. 625–645.
- [55] Marusic, I., and Hutchins, N., "Study of the Log-Layer Structure in Wall Turbulence over a Very Large Range of Reynolds number," *Flow Turbulence and Combustion*, Vol. 81, 2008, pp. 115–130.
- [56] Guala, M., Metzger, M., and McKeon, B. J., "Interactions within the Turbulent Boundary Layer at High Reynolds Number," *Journal of Fluid Mechanics*, Vol. 666, 2011, pp. 573–604.
- [57] Laufer, J., "Investigation of Turbulent Flow in a Two-Dimensional Channel," NACA TR-1053, 1951.
- [58] Zanoun, E.-S., Durst, F., and Nagib, H., "Evaluating the Law of the Wall in Two-dimensional Fully Developed Turbulent Channel Flows," *Physics of Fluids*, Vol. 15, No. 10, October 2003, pp. 3079–3089.
- [59] Nikuradse, J., "Strömungsgesetze in rauhen Röhren," VDI Forschungsheft 361, 1933 [see also NACA TM-1292, 1950].
- [60] Laufer, J., "The Structure of Turbulence in Fully Developed Pipe Flow," NACA TR-1174, 1954.
- [61] Shockling, M. A., Allen, J. J., and Smits, A. J., "Roughness Effects in Turbulent Pipe Flow," *Journal of Fluid Mechanics*, Vol. 564, 2006, pp. 267–285.
- [62] Bradbury, L. J. S., "The Structure of a Self-Preserving Turbulent Plane Jet," *Journal of Fluid Mechanics*, Vol. 23, 1965, pp. 31–64.
- [63] Heskestad, G., "Hot-Wire Measurements in a Plane Turbulent Jet," *Journal of Applied Mechanics*, Vol. 32, No. 4, 1965, pp. 721–734.
- [64] Wagnanski, I., and Fiedler, H. E., "Some Measurements in the Self-Preserving Jet," *Journal of Fluid Mechanics*, Vol. 38, 1969, pp. 577–612.
- [65] Rodi, W., "A New Method of Analyzing Hot-Wire Signals in Highly Turbulent Flow, and Its Evaluation in a Round Jet," *Disa Information*, No. 17, 1975.
- [66] Pope, S. B., "An Explanation of the Turbulent Round-Jet/Plane-Jet Anomaly," *AIAA Journal*, Vol. 16, No. 3, 1978, pp. 279–281.
- [67] Durbin, P. A., "Application of a Near-Wall Turbulence Model to Boundary Layers and Heat Transfer," *International Journal of Heat and Fluid Flow*, Vol. 14, No. 4, 1993, pp. 316–323.
- [68] Lam, C. K. G. and Bremhorst, K. A., "Modified Form of the k - ϵ Model for Predicting Wall Turbulence," *ASME, Journal of Fluids Engineering*, Vol. 103, 1981, pp. 456–460.
- [69] Launder, B. E. and Sharma, B. I., "Application of the Energy Dissipation Model of Turbulence to the Calculation of Flow Near a Spinning Disc," *Letters in Heat and Mass Transfer*, Vol. 1, No. 2, 1974, pp. 131–138.
- [70] Huang, P. G., Coleman, G. H., and Bradshaw, P., "Compressible Turbulent Channel Flows: DNS Results and Modeling," *Journal of Fluid Mechanics*, Vol. 305, 1995, pp. 185–218.

- [71] Kreuzinger, J., Friedrich, R., and Gatski, T. B., "Compressibility Effects in the Solenoidal Dissipation Rate Equation: A Priori Assessment and Modeling," *International Journal of Heat and Fluid Flow*, Vol. 27, 2006, pp. 696–706.
- [72] Anderson, P. S., Kays, W. M., and Moffat, R. J., "Experimental Results for the Transpired Turbulent Boundary Layer in an Adverse Pressure Gradient," *Journal of Fluid Mechanics*, Vol. 69, 1975, pp. 353–375.
- [73] Colebrook, C. F., "Turbulent Flow in Pipes, with particular reference to the Transition Region between the Smooth and Rough Pipe Laws," *Journal of the Institution of Civil Engineers*, Vol. 11, 1939, pp. 133–156.
- [74] Rhie, C. M., "A Numerical Study of the Flow Past an Isolated Airfoil with Separation," Ph.D. Thesis, Dept. of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, 1981.
- [75] Shyy, W., Udaykumar, H. S., Rao, M. M., and Smith, R. W., "Governing Equations and Solution Procedure," *Computational Fluid Dynamics with Moving Boundaries*, Taylor and Francis, Philadelphia, PA, 1996, pp. 21–23.
- [76] Versteeg, H. K., and Malalasekera, W., "The Finite Volume Method for Convection-Diffusion Problems," *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, Pearson Education Limited, Harlow, England, 1995, pp. 103–134.
- [77] Ferziger, J. H., and Peric, M., "Finite Volume Methods," *Computational Methods for Fluid Dynamics*, Springer-Verlag Berlin Heidelberg, 1996, pp. 71–84.
- [78] Patankar, S. V. and Spalding, D. B., "A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-dimensional Parabolic Flows," *Int. Journal of Heat Mass Transfer*, Vol. 15, 1972, p. 1787–1806.
- [79] Versteeg, H. K., and Malalasekera, W., "Solution Algorithms for Pressure-Velocity Coupling in Steady Flows," *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, Pearson Education Limited, Harlow, England, 1995, pp. 135–155.
- [80] Rhie, C. M., and Chow, W. L., "Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation," *AIAA Journal*, Vol. 21, No. 11, Nov. 1983, pp. 1525–1532.
- [81] Richardson, L. F., "The Approximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations, with an Application to the Stresses in a Masonry Dam," *Transactions of the Royal Society of London, A*, Vol. 210, Jan. 1910, pp. 307–357.
- [82] Richardson, L. F., and Gaunt, J. A., "The Deferred Approach to the Limit," *Transactions of the Royal Society of London, A*, Vol. 226, Jan. 1927, pp. 299–361.
- [83] Phillips, W. F., Hunsaker, D. F., and Spall, R. E., "Smooth-Wall Boundary Conditions for Dissipation-Based Turbulence Models," 48th AIAA Aerospace Sciences Meeting, Orlando, Florida, AIAA 2010-1103, 2010.
- [84] Patel, V. C., Rodi, W., and Scheuerer, G., "Turbulence Models for Near-Wall and Low Reynolds Number Flows: A Review," *AIAA Journal*, Vol. 23, No. 9, 1985, pp. 1308–1319.
- [85] Fluent 6.3, "Standard $k-\omega$ Model," *FLUENT 6.3 User's Guide*, Fluent Inc., Lebanon, NH, Sept. 2006, pp. 12-26–12-31.

- [86] Wilcox, D. C., "Numerical Accuracy Near Boundaries," *Turbulence Modeling for CFD*, Second Edition, DCW Industries, Inc., La Cañada, California, 1998, pp. 341–344.
- [87] Kline, S. J., Cantwell, B. J., and Lilley, G. M. (eds.), *Proceedings of the 1980–81 AFOSR-HTTM-Stanford Conference on Complex Turbulent Flows*, Vol. 1, Stanford University Press, Stanford, CA, 1981.
- [88] Van Driest, E. R., "On Turbulent Flow Near a Wall," *Journal of Aerospace Sciences*, Vol. 23, 1956, pp. 1007–1011.
- [89] Metzger, M. M. and Klewicki, J. C., "A Comparative Study of Near-wall Turbulence in High and Low Reynolds Number Boundary Layers," *Physics of Fluids*, Vol. 13, Number 3, 2001, pp. 692–701.
- [90] Marusic, I., McKeon, B. J., Monkewitz, P. A., Nagib, H. M., Smits, A. J., and Sreenivasan K. R., "Wall-bounded Turbulent Flows at High Reynolds Numbers: Recent Advances and Key Issues," *Physics of Fluids*, Vol. 22, 2010, pp. 065103-1–065103-24.
- [91] Schlichting, H., "Experimentelle Untersuchungen zum Rauigkeitsproblem," *Ing.-Arch.* Vol. 7, 1936, pp. 1–34.
- [92] Schlichting, H., *Grenzschicht-Theorie*, vorm. G. Braunsche Hofbuchdruckerei u. Verlag GmbH, Karlsruhe, Germany, 1951.
- [93] Schlichting, H., "Rough Pipes and Equivalent Sand Roughness," *Boundary Layer Theory*, Fourth Edition, McGraw-Hill, New York, 1960, pp. 519–527.
- [94] Moody, L. F., "Friction Factors for Pipe Flow," *Transactions of the A.S.M.E.*, Vol. 66, 1944, pp. 671–684.
- [95] Reichardt, H., "Die Grundlagen des turbulenten Wärmeüberganges," *Arch. Ges. Wärmetechnik 2*, 1951, pp. 129–142.
- [96] Kays, W. M., and Crawford, M. E., "Momentum Transfer: Turbulent Flow in Tubes," *Convective Heat and Mass Transfer*, Third Edition, McGraw-Hill, Inc., 1993, pp. 244–254.
- [97] Broyden, C. G., "The Convergence of a Class of Double-rank Minimization Algorithms," *Journal of the Institute of Mathematics and Its Applications* Vol. 6, 1970, pp. 76–90.
- [98] Fletcher, R., "A New Approach to Variable Metric Algorithms," *The Computer Journal*, Vol. 13, 1970, pp. 317–322.
- [99] Goldfarb, D., "A Family of Variable Metric Methods Derived by Variational Means," *Mathematics of Computation*, Vol. 24, 1970, pp. 23–16.
- [100] Shanno, D. F., "Conditioning of Quasi-Newton Methods for Function Minimization," *Mathematics of Computation*, Vol. 24, 1970, pp. 647–656.
- [101] Launder, B. E., and Spalding, D. B., "The Numerical Computation of Turbulent Flows," *Computer Methods in Applied Mechanics and Engineering*, Vol. 3, 1974, pp. 269–289.
- [102] Chien, K.-Y., "Predictions of Channel and Boundary-Layer Flows with a Low-Reynolds-Number Turbulence Model," *AIAA Journal*, Vol. 20, No. 1, 1982, pp. 33–38.
- [103] Saffman, P. G., "A Model for Inhomogeneous Turbulent Flow," *Proceedings of the Royal Society of London, Series A*, Vol. 317, 1970, pp. 417–433.

- [104] Blasius, H., “Grenzschichten in Flüssigkeiten mit kleiner Reibung,” *Z. Angew. Math. Phys.*, Vol. 56, 1908, pp. 1–37 [English translation in NACA TM-1256, 1950].
- [105] White, F. M., “The Turbulent Boundary Layer On A Flat Plate,” *Viscous Fluid Flow*, Third Edition, McGraw Hill, New York, New York, 2006, pp. 433–440.
- [106] Kays, W. M., and Crawford, M. E., “Momentum Transfer: The Turbulent Boundary Layer,” *Convective Heat and Mass Transfer*, Third Edition, McGraw-Hill, Inc., 1993, pp. 192–243.
- [107] Schultz-Grunow, F., “New Frictional Resistance Law for Smooth Plates,” NACA TM-986, 1941.
- [108] Schlichting, H., “Laminare Strahlausbreitung,” *Zeitschrift für angewandte Mathematik und Mechanik*, Vol. 13, 1933, pp. 260–263.

APPENDICES

APPENDIX A

MATHEMATICAL IDENTITIES

I. Vector Identities

$$\nabla \cdot (\mathbf{UV}) = (\mathbf{U} \cdot \nabla)\mathbf{V} + \mathbf{V}(\nabla \cdot \mathbf{U}) \quad (\text{A.1})$$

$$\nabla \cdot (s\mathbf{V}) = (\mathbf{V} \cdot \nabla)s + s(\nabla \cdot \mathbf{V}) \quad (\text{A.2})$$

$$\nabla \cdot (\vec{\sigma} \cdot \mathbf{V}) = \vec{\sigma} \cdot (\nabla \mathbf{V}) + \mathbf{V}(\nabla \cdot \vec{\sigma}) \quad (\text{A.3})$$

$$\mathbf{U} \cdot \left[\frac{\partial \mathbf{U}}{\partial t} + (\mathbf{V} \cdot \nabla)\mathbf{U} \right] = \frac{\partial}{\partial t} \left(\frac{1}{2} U^2 \right) + (\mathbf{V} \cdot \nabla) \left(\frac{1}{2} U^2 \right) \quad (\text{A.4})$$

II. Ensemble Averaging Identities

$$\overline{\tilde{\varphi}} = 0 \quad (\text{A.5})$$

$$\begin{aligned} \overline{\varphi\psi} &= \overline{\varphi}\overline{\psi} + \overline{\tilde{\varphi}\tilde{\psi}} + \overline{\tilde{\varphi}}\overline{\tilde{\psi}} + \overline{\tilde{\varphi}}\overline{\tilde{\psi}} \\ &= \overline{\varphi}\overline{\psi} + \overline{\tilde{\varphi}\tilde{\psi}} \end{aligned} \quad (\text{A.6})$$

$$\overline{\varphi\psi\zeta} = \overline{\varphi}\overline{\psi}\overline{\zeta} + \overline{\tilde{\varphi}\tilde{\psi}\tilde{\zeta}} + \overline{\tilde{\varphi}}\overline{\tilde{\psi}}\overline{\tilde{\zeta}} + \overline{\tilde{\varphi}\tilde{\psi}}\overline{\tilde{\zeta}} + \overline{\tilde{\varphi}\tilde{\psi}\tilde{\zeta}} \quad (\text{A.7})$$

$$\frac{\partial \overline{f}}{\partial x} = \overline{\frac{\partial f}{\partial x}} \quad (\text{A.8})$$

III. Flowfield Properties

The strain-rate tensor is given by

$$\vec{\mathbf{S}}(\mathbf{V}) \equiv \frac{1}{2} \begin{bmatrix} \left(\frac{\partial V_x}{\partial x} + \frac{\partial V_x}{\partial x} \right) & \left(\frac{\partial V_x}{\partial y} + \frac{\partial V_y}{\partial x} \right) & \left(\frac{\partial V_x}{\partial z} + \frac{\partial V_z}{\partial x} \right) \\ \left(\frac{\partial V_y}{\partial x} + \frac{\partial V_x}{\partial y} \right) & \left(\frac{\partial V_y}{\partial y} + \frac{\partial V_y}{\partial y} \right) & \left(\frac{\partial V_y}{\partial z} + \frac{\partial V_z}{\partial y} \right) \\ \left(\frac{\partial V_z}{\partial x} + \frac{\partial V_x}{\partial z} \right) & \left(\frac{\partial V_z}{\partial y} + \frac{\partial V_y}{\partial z} \right) & \left(\frac{\partial V_z}{\partial z} + \frac{\partial V_z}{\partial z} \right) \end{bmatrix} \quad (\text{A.9})$$

The Jacobian tensor of a vector field is

$$\bar{\bar{\mathbf{J}}}(\mathbf{V}) \equiv \begin{bmatrix} \frac{\partial V_x}{\partial x} & \frac{\partial V_x}{\partial y} & \frac{\partial V_x}{\partial z} \\ \frac{\partial V_y}{\partial x} & \frac{\partial V_y}{\partial y} & \frac{\partial V_y}{\partial z} \\ \frac{\partial V_z}{\partial x} & \frac{\partial V_z}{\partial y} & \frac{\partial V_z}{\partial z} \end{bmatrix} \quad (\text{A.10})$$

The rotation tensor is

$$\bar{\bar{\mathbf{\Omega}}}(\mathbf{V}) \equiv \frac{1}{2} \begin{bmatrix} \left(\frac{\partial V_x}{\partial x} - \frac{\partial V_x}{\partial x} \right) & \left(\frac{\partial V_x}{\partial y} - \frac{\partial V_y}{\partial x} \right) & \left(\frac{\partial V_x}{\partial z} - \frac{\partial V_z}{\partial x} \right) \\ \left(\frac{\partial V_y}{\partial x} - \frac{\partial V_x}{\partial y} \right) & \left(\frac{\partial V_y}{\partial y} - \frac{\partial V_y}{\partial y} \right) & \left(\frac{\partial V_y}{\partial z} - \frac{\partial V_z}{\partial y} \right) \\ \left(\frac{\partial V_z}{\partial x} - \frac{\partial V_x}{\partial z} \right) & \left(\frac{\partial V_z}{\partial y} - \frac{\partial V_y}{\partial z} \right) & \left(\frac{\partial V_z}{\partial z} - \frac{\partial V_z}{\partial z} \right) \end{bmatrix} \quad (\text{A.11})$$

This can be written in terms of vorticity

$$\bar{\bar{\mathbf{\Omega}}}(\mathbf{V}) \equiv \frac{1}{2} \begin{bmatrix} 0 & -(\nabla \times \mathbf{V})_z & (\nabla \times \mathbf{V})_y \\ (\nabla \times \mathbf{V})_z & 0 & -(\nabla \times \mathbf{V})_x \\ -(\nabla \times \mathbf{V})_y & (\nabla \times \mathbf{V})_x & 0 \end{bmatrix} \quad (\text{A.12})$$

Note that

$$\bar{\bar{\mathbf{J}}}(\mathbf{V}) = \bar{\bar{\mathbf{S}}}(\mathbf{V}) + \bar{\bar{\mathbf{\Omega}}}(\mathbf{V}) \quad (\text{A.13})$$

The squared magnitude of the strain-rate tensor is given the symbol S_V^2 and is defined as

$$\begin{aligned} S_V^2 &\equiv \bar{\bar{\mathbf{S}}}(\mathbf{V}) \cdot \bar{\bar{\mathbf{S}}}(\mathbf{V}) \\ &= \left(\frac{\partial V_x}{\partial x} \right)^2 + \left(\frac{\partial V_y}{\partial y} \right)^2 + \left(\frac{\partial V_z}{\partial z} \right)^2 + \frac{1}{2} \left(\frac{\partial V_y}{\partial x} + \frac{\partial V_x}{\partial y} \right)^2 + \frac{1}{2} \left(\frac{\partial V_z}{\partial y} + \frac{\partial V_y}{\partial z} \right)^2 + \frac{1}{2} \left(\frac{\partial V_x}{\partial z} + \frac{\partial V_z}{\partial x} \right)^2 \end{aligned} \quad (\text{A.14})$$

The squared magnitude of the rotation tensor is given the symbol Ω_V^2 and is defined as

$$\Omega_V^2 \equiv \bar{\bar{\mathbf{\Omega}}}(\mathbf{V}) \cdot \bar{\bar{\mathbf{\Omega}}}(\mathbf{V}) = \frac{1}{2} \left(\frac{\partial V_z}{\partial y} - \frac{\partial V_y}{\partial z} \right)^2 + \frac{1}{2} \left(\frac{\partial V_x}{\partial z} - \frac{\partial V_z}{\partial x} \right)^2 + \frac{1}{2} \left(\frac{\partial V_y}{\partial x} - \frac{\partial V_x}{\partial y} \right)^2 \quad (\text{A.15})$$

This can also be written as one-half of the squared magnitude of the vorticity vector

$$\bar{\bar{\mathbf{\Omega}}}(\mathbf{V}) \cdot \bar{\bar{\mathbf{\Omega}}}(\mathbf{V}) = \frac{1}{2} (\nabla \times \mathbf{V}) \cdot (\nabla \times \mathbf{V}) \quad (\text{A.16})$$

APPENDIX B

TRADITIONAL TURBULENCE MODELS

I. The k - ε ModelA. The General k - ε Model Equations

The k - ε model was first made popular by Jones and Launder [15] and has become one of the most widely used turbulence models. In its general, steady-state, incompressible form, the complete k - ε model includes the continuity equation, the RANS equations, the turbulent kinetic energy equation, the dissipation transport equation, and the closing kinematic viscosity equation

$$\begin{aligned}
 \nabla \cdot \mathbf{V} &= 0 \\
 (\mathbf{V} \cdot \nabla) \mathbf{V} &= -\nabla \hat{p} / \rho + \nabla \cdot [2(\nu + \nu_t) \bar{\bar{\mathbf{S}}}(\mathbf{V})] \\
 (\bar{\mathbf{V}} \cdot \nabla) k &= 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \varepsilon + \nabla \cdot [(\nu + \nu_t / \sigma_k) \nabla k] \\
 (\bar{\mathbf{V}} \cdot \nabla) \varepsilon &= 2C_{\varepsilon 1} \nu_t \frac{\varepsilon}{k} \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - C_{\varepsilon 2} \frac{\varepsilon^2}{k} + \nabla \cdot [(\nu + \nu_t / \sigma_\varepsilon) \nabla \varepsilon] \\
 \nu_t &= C_\mu k^2 / \varepsilon
 \end{aligned} \tag{B.1}$$

This model has difficulty near wall boundaries and is generally modified in one of two ways depending on the flow Reynolds number. For high Reynolds number flows, Launder and Spalding [101] suggest the use of wall functions. For low Reynolds number flows, damping functions are employed which allow the model to be integrated to the wall. Because this work is more comparable to models capable of integration to the wall, special attention is given here to damping function models. Several damping function models exist including those by Jones and Launder [15], Launder and Sharma [69], Lam and Bremhorst [68], and Chien [102]. Patel et al. [84] compare the performance of several models.

Many low Reynolds number k - ε models can be written in the form

$$\begin{aligned}
 (\bar{\mathbf{V}} \cdot \nabla) k &= 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - \varepsilon - \varepsilon_o + \nabla \cdot [(\nu + \nu_t / \sigma_k) \nabla k] \\
 (\bar{\mathbf{V}} \cdot \nabla) \varepsilon &= 2C_{\varepsilon 1} f_1 \nu_t \frac{\varepsilon}{k} \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - C_{\varepsilon 2} f_2 \frac{\varepsilon^2}{k} + E + \nabla \cdot [(\nu + \nu_t / \sigma_\varepsilon) \nabla \varepsilon] \\
 \nu_t &= C_\mu f_\mu k^2 / \varepsilon
 \end{aligned} \tag{B.2}$$

where the models differ only in their definitions of f_μ , f_1 , f_2 , E , ε_o , and the closure constants, $C_{\varepsilon 1}$, $C_{\varepsilon 2}$, C_μ , σ_k , and σ_ε . In this work, only the low-Reynolds number models of Launder and Sharma [69] and Lam and Bremhorst [68] are included.

B. Fully Developed Channel Flow

The k - ε model with damping functions can be written in its general form for incompressible, steady-state flow as

$$(\bar{\mathbf{V}} \cdot \nabla)k = 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \varepsilon - \varepsilon_o + \nabla \cdot [(v + \nu_t / \sigma_k) \nabla k] \quad (\text{B.3})$$

$$(\bar{\mathbf{V}} \cdot \nabla)\varepsilon = 2C_{\varepsilon 1} f_1 \nu_t \frac{\varepsilon}{k} \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - C_{\varepsilon 2} f_2 \frac{\varepsilon^2}{k} + E + \nabla \cdot [(v + \nu_t / \sigma_\varepsilon) \nabla \varepsilon] \quad (\text{B.4})$$

$$\nu_t = C_\mu f_\mu k^2 / \varepsilon \quad (\text{B.5})$$

The k and ε transport equations can be written for 2-D flow in Cartesian coordinates as

$$\frac{d(\bar{V}_y k)}{dy} = 2\nu_t \left[\left(\frac{d\bar{V}_y}{dy} \right)^2 + \frac{1}{2} \left(\frac{d\bar{V}_x}{dy} \right)^2 \right] - \varepsilon - \varepsilon_o + \frac{d}{dy} \left[(v + \nu_t / \sigma_k) \frac{dk}{dy} \right] \quad (\text{B.6})$$

$$\frac{d(\bar{V}_y \varepsilon)}{dy} = 2C_{\varepsilon 1} f_1 \nu_t \frac{\varepsilon}{k} \left[\left(\frac{d\bar{V}_y}{dy} \right)^2 + \frac{1}{2} \left(\frac{d\bar{V}_x}{dy} \right)^2 \right] - C_{\varepsilon 2} f_2 \frac{\varepsilon^2}{k} + E + \frac{d}{dy} \left[(v + \nu_t / \sigma_\varepsilon) \frac{d\varepsilon}{dy} \right] \quad (\text{B.7})$$

For fully developed channel flow, the transport property gradients with respect to x are much smaller than those with respect to y . Therefore, eliminating the x -derivatives and using Eq. (C.20) gives the two-dimensional fully developed k and ε transport equations

$$-\frac{d}{dy} \left[(v + \nu_t / \sigma_k) \frac{dk}{dy} \right] = \nu_t \left(\frac{d\bar{V}_x}{dy} \right)^2 - \varepsilon - \varepsilon_o \quad (\text{B.8})$$

$$-\frac{d}{dy} \left[(v + \nu_t / \sigma_\varepsilon) \frac{d\varepsilon}{dy} \right] = C_{\varepsilon 1} f_1 \nu_t \frac{\varepsilon}{k} \left(\frac{d\bar{V}_x}{dy} \right)^2 - C_{\varepsilon 2} f_2 \frac{\varepsilon^2}{k} + E \quad (\text{B.9})$$

Using the nondimensional definitions

$$k^+ \equiv \frac{k}{u_\tau^2}, \quad \varepsilon^+ \equiv \frac{\varepsilon \nu}{u_\tau^4}, \quad \varepsilon_o^+ \equiv \frac{\varepsilon_o \nu}{u_\tau^4}, \quad E^+ \equiv \frac{E \nu^2}{u_\tau^6}, \quad R_\tau \equiv \frac{L u_\tau}{\nu} \quad (\text{B.10})$$

and substituting in the results of Eq. (C.31), gives the nondimensional fully developed k - ε model

$$\frac{d}{dy^+} \left[(1 + \nu^+ / \sigma_k) \frac{dk^+}{dy^+} \right] = \varepsilon^+ + \varepsilon_o^+ - \nu^+ \left(\frac{R_\tau - y^+}{R_\tau (1 + \nu^+)} \right)^2 \quad (\text{B.11})$$

$$\frac{d}{dy^+} \left[(1 + \nu^+ / \sigma_\varepsilon) \frac{d\varepsilon^+}{dy^+} \right] = C_{\varepsilon 2} f_2 \frac{\varepsilon^{+2}}{k^+} - C_{\varepsilon 1} f_1 \nu^+ \frac{\varepsilon^+}{k^+} \left(\frac{R_\tau - y^+}{R_\tau (1 + \nu^+)} \right)^2 - E^+ \quad (\text{B.12})$$

$$\nu^+ = C_\mu f_\mu k^{+2} / \varepsilon^+ \quad (\text{B.13})$$

In order to close this formulation, the damping functions f_μ , f_1 , f_2 , E^+ , and ε_o^+ must be specified. Various functions have been proposed, and the subsequent sections will discuss one such proposal. In addition to the definitions of the damping functions, four boundary conditions must be specified. These are

$$\frac{dk^+}{dy^+}(R_\tau) = 0, \quad \frac{d\varepsilon^+}{dy^+}(R_\tau) = 0, \quad k^+(0) = 0, \quad \frac{dk^+}{dy^+}(0) = 0 \quad (\text{B.14})$$

The first two conditions are a result of symmetry at the centerline of the channel. The third and fourth conditions are a result of the definition of k and the fact that the velocity fluctuations must go to zero at the wall. The final boundary condition for k is not widely recognized in the literature. The most common approaches include examining the near-wall behavior of ε and specifying this behavior as a boundary condition. However, such an approach is mathematically incorrect. The governing equations cannot be used to invoke a boundary condition at the wall as discussed in Chapter 5.

C. Fully Developed Pipe Flow

Several low-Reynolds number versions of the k - ε model exist and can be written in the form

$$\begin{aligned} \nu_t &= C_\mu f_\mu k^2 / \varepsilon \\ (\bar{\mathbf{V}} \cdot \nabla) k &= 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - \varepsilon - \varepsilon_o + \nabla \cdot [(\nu + \nu_t / \sigma_k) \nabla k] \\ (\bar{\mathbf{V}} \cdot \nabla) \varepsilon &= 2C_{\varepsilon 1} f_1 \nu_t \frac{\varepsilon}{k} \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - C_{\varepsilon 2} f_2 \frac{\varepsilon^2}{k} + E + \nabla \cdot [(\nu + \nu_t / \sigma_\varepsilon) \nabla \varepsilon] \end{aligned} \quad (\text{B.15})$$

where the models differ only in their definitions of f_μ , f_1 , f_2 , E , ε_o , and the closure constants, $C_{\varepsilon 1}$, $C_{\varepsilon 2}$, C_μ , σ_k , and σ_ε . The turbulent-kinetic-energy and dissipation-frequency transport equations can be written in cylindrical coordinates as

$$\begin{aligned}
\bar{V}_r \frac{\partial k}{\partial r} + \frac{\bar{V}_\theta}{r} \frac{\partial k}{\partial \theta} + \bar{V}_z \frac{\partial k}{\partial z} = 2\nu_t \left[\left(\frac{\partial \bar{V}_r}{\partial r} \right)^2 + \left(\frac{1}{r} \frac{\partial \bar{V}_\theta}{\partial \theta} + \frac{\bar{V}_r}{r} \right)^2 + \left(\frac{\partial \bar{V}_z}{\partial z} \right)^2 \right. \\
\left. + \frac{1}{2} \left(r \frac{\partial(\bar{V}_\theta/r)}{\partial r} + \frac{1}{r} \frac{\partial \bar{V}_r}{\partial \theta} \right)^2 + \frac{1}{2} \left(\frac{1}{r} \frac{\partial \bar{V}_z}{\partial \theta} + \frac{\partial \bar{V}_\theta}{\partial z} \right)^2 + \frac{1}{2} \left(\frac{\partial \bar{V}_r}{\partial z} + \frac{\partial \bar{V}_z}{\partial r} \right)^2 \right] \\
- \varepsilon - \varepsilon_o + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t/\sigma_k) r \frac{\partial k}{\partial r} \right] + \frac{1}{r} \frac{\partial}{\partial \theta} \left[(v + v_t/\sigma_k) \frac{1}{r} \frac{\partial k}{\partial \theta} \right] \\
+ \frac{\partial}{\partial z} \left[(v + v_t/\sigma_k) \frac{\partial k}{\partial z} \right]
\end{aligned} \tag{B.16}$$

$$\begin{aligned}
\bar{V}_r \frac{\partial \varepsilon}{\partial r} + \frac{\bar{V}_\theta}{r} \frac{\partial \varepsilon}{\partial \theta} + \bar{V}_z \frac{\partial \varepsilon}{\partial z} = 2C_{\varepsilon 1} f_1 \nu_t \frac{\varepsilon}{k} \left[\left(\frac{\partial \bar{V}_r}{\partial r} \right)^2 + \left(\frac{1}{r} \frac{\partial \bar{V}_\theta}{\partial \theta} + \frac{\bar{V}_r}{r} \right)^2 + \left(\frac{\partial \bar{V}_z}{\partial z} \right)^2 \right. \\
\left. + \frac{1}{2} \left(r \frac{\partial(\bar{V}_\theta/r)}{\partial r} + \frac{1}{r} \frac{\partial \bar{V}_r}{\partial \theta} \right)^2 + \frac{1}{2} \left(\frac{1}{r} \frac{\partial \bar{V}_z}{\partial \theta} + \frac{\partial \bar{V}_\theta}{\partial z} \right)^2 + \frac{1}{2} \left(\frac{\partial \bar{V}_r}{\partial z} + \frac{\partial \bar{V}_z}{\partial r} \right)^2 \right] \\
- C_{\varepsilon 2} f_2 \frac{\varepsilon^2}{k} + E + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t/\sigma_\varepsilon) r \frac{\partial \varepsilon}{\partial r} \right] + \frac{1}{r} \frac{\partial}{\partial \theta} \left[(v + v_t/\sigma_\varepsilon) \frac{1}{r} \frac{\partial \varepsilon}{\partial \theta} \right] \\
+ \frac{\partial}{\partial z} \left[(v + v_t/\sigma_\varepsilon) \frac{\partial \varepsilon}{\partial z} \right]
\end{aligned} \tag{B.17}$$

Applying the simplifications for fully developed flow in a pipe, these equations can be written as

$$\begin{aligned}
\bar{V}_r \frac{\partial k}{\partial r} = 2\nu_t \left[\left(\frac{\partial \bar{V}_r}{\partial r} \right)^2 + \left(\frac{\bar{V}_r}{r} \right)^2 + \frac{1}{2} \left(\frac{\partial \bar{V}_z}{\partial r} \right)^2 \right] - \varepsilon - \varepsilon_o + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t/\sigma_k) r \frac{\partial k}{\partial r} \right] \\
\bar{V}_r \frac{\partial \varepsilon}{\partial r} = 2C_{\varepsilon 1} f_1 \nu_t \frac{\varepsilon}{k} \left[\left(\frac{\partial \bar{V}_r}{\partial r} \right)^2 + \left(\frac{\bar{V}_r}{r} \right)^2 + \frac{1}{2} \left(\frac{\partial \bar{V}_z}{\partial r} \right)^2 \right] - C_{\varepsilon 2} f_2 \frac{\varepsilon^2}{k} + E + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t/\sigma_\varepsilon) r \frac{\partial \varepsilon}{\partial r} \right]
\end{aligned} \tag{B.18}$$

Applying Eq. (C.59) gives

$$\begin{aligned}
\frac{1}{r} \frac{\partial}{\partial r} \left[-(v + v_t/\sigma_k) r \frac{\partial k}{\partial r} \right] = \nu_t \left(\frac{\partial \bar{V}_z}{\partial r} \right)^2 - \varepsilon - \varepsilon_o \\
\frac{1}{r} \frac{\partial}{\partial r} \left[-(v + v_t/\sigma_\varepsilon) r \frac{\partial \varepsilon}{\partial r} \right] = C_{\varepsilon 1} f_1 \nu_t \frac{\varepsilon}{k} \left(\frac{\partial \bar{V}_z}{\partial r} \right)^2 - C_{\varepsilon 2} f_2 \frac{\varepsilon^2}{k} + E
\end{aligned} \tag{B.19}$$

The k - ε formulation with the remaining boundary conditions can be written as

$$\begin{aligned}
\frac{d\bar{V}_z}{dr} &= -\frac{u_\tau^2}{(\nu + \nu_t)} \frac{r}{R} \\
\nu_t &= C_\mu f_\mu k^2 / \varepsilon \\
\frac{1}{r} \frac{\partial}{\partial r} \left[-(\nu + \nu_t / \sigma_k) r \frac{\partial k}{\partial r} \right] &= \nu_t \left(\frac{\partial \bar{V}_z}{\partial r} \right)^2 - \varepsilon - \varepsilon_o \\
\frac{1}{r} \frac{\partial}{\partial r} \left[-(\nu + \nu_t / \sigma_\varepsilon) r \frac{\partial \varepsilon}{\partial r} \right] &= C_{\varepsilon 1} f_1 \nu_t \frac{\varepsilon}{k} \left(\frac{\partial \bar{V}_z}{\partial r} \right)^2 - C_{\varepsilon 2} f_2 \frac{\varepsilon^2}{k} + E \\
\bar{V}_z(R) = 0, \quad k(R) = 0, \quad \frac{dk}{dr}(R) = 0, \quad \frac{dk}{dr}(0) = 0, \quad \frac{d\varepsilon}{dr}(0) = 0
\end{aligned} \tag{B.20}$$

Applying the nondimensional parameter definitions

$$k^+ \equiv \frac{k}{u_\tau^2}, \quad \varepsilon^+ \equiv \frac{\varepsilon \nu}{u_\tau^4}, \quad \varepsilon_o^+ \equiv \frac{\varepsilon_o \nu}{u_\tau^4}, \quad E^+ \equiv \frac{E \nu^2}{u_\tau^6} \tag{B.21}$$

with those given in Eq. (C.68), the formulation can be written as

$$\begin{aligned}
\frac{du^+}{d\hat{r}} &= -\frac{R_\tau \hat{r}}{(1 + \nu^+)} \\
\nu^+ &= C_\mu f_\mu k^{+2} / \varepsilon^+ \\
\frac{1}{\hat{r}} \frac{d}{d\hat{r}} \left[-(1 + \nu^+ / \sigma_k) \hat{r} \frac{dk^+}{d\hat{r}} \right] &= \nu^+ \left(\frac{du^+}{d\hat{r}} \right)^2 - \varepsilon^+ - \varepsilon_o^+ \\
\frac{1}{\hat{r}} \frac{d}{d\hat{r}} \left[-(1 + \nu^+ / \sigma_\omega) \hat{r} \frac{d\omega^+}{d\hat{r}} \right] &= C_{\varepsilon 1} f_1 \nu^+ \frac{\varepsilon^+}{k^+} \left(\frac{du^+}{d\hat{r}} \right)^2 - C_{\varepsilon 2} f_2 \frac{\varepsilon^{+2}}{k^+} + E^+ \\
u^+(1) = 0, \quad k^+(1) = 0, \quad \frac{dk^+}{d\hat{r}}(1) = 0, \quad \frac{dk^+}{d\hat{r}}(0) = 0, \quad \frac{d\varepsilon^+}{d\hat{r}}(0) = 0
\end{aligned} \tag{B.22}$$

D. Sample Models

1. The Launder-Sharma k - ε Model

The Launder-Sharma [69] turbulence model is a special case of Eq. (B.2) where

$$\begin{aligned}
f_\mu &= \exp[-3.4/(1 + R_t/50)^2], \quad f_1 = 1.0, \quad f_2 = 1 - 0.3 \exp(-R_t^2), \\
R_t &= \frac{k^2}{\nu \varepsilon}, \quad \varepsilon_o = 2\nu \left(\frac{\partial \sqrt{k}}{\partial y} \right)^2, \quad E = 2\nu \nu_t \left(\frac{\partial^2 \bar{V}_x}{\partial y^2} \right)^2, \\
C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
\end{aligned} \tag{B.23}$$

where y is the coordinate measured normal to the wall and \bar{V}_x is the velocity tangential to the wall. This formulation can be written in terms of the k and ε transport equations given in Table F.1 and is a special case where

$$\begin{aligned}
 f_{k1} &= 2\nu_t, & f_{k2} &= \varepsilon + 2\nu \left(\frac{\partial \sqrt{k}}{\partial y} \right)^2, \\
 f_{\varepsilon 1} &= 2C_{\varepsilon 1} C_{\mu} f_{\mu} k, & f_{\varepsilon 2} &= C_{\varepsilon 2} [1 - 0.3 \exp(-R_t^2)] \frac{\varepsilon^2}{k} - 2\nu \nu_t \left(\frac{\partial^2 \bar{V}_x}{\partial y^2} \right)^2, \\
 R_t &= \frac{k^2}{\nu \varepsilon}, & \nu_t &= C_{\mu} f_{\mu} k^2 / \varepsilon, & f_{\mu} &= \exp[-3.4 / (1 + R_t / 50)^2], \\
 C_{\mu} &= 0.09, & C_{\varepsilon 1} &= 1.44, & C_{\varepsilon 2} &= 1.92, & \sigma_k &= 1.0, & \sigma_{\varepsilon} &= 1.3
 \end{aligned} \tag{B.24}$$

This model was designed so that $\varepsilon \rightarrow 0$ as $y \rightarrow 0$. Therefore, the near-wall approximation for ε is

$$f_{\varepsilon 0} = 0 \tag{B.25}$$

This is not a boundary condition. It is simply a near-wall approximation for ε and can be applied at a no-slip wall.

The Launder-Sharma turbulence model can be written for two-dimensional fully developed channel flow as

$$\begin{aligned}
 \frac{\partial}{\partial y} \left[(v + \nu_t) \frac{du}{dy} \right] &= \frac{1}{\rho} \frac{d\hat{p}}{dx} \\
 -\frac{d}{dy} \left[(v + \nu_t / \sigma_k) \frac{dk}{dy} \right] &= \nu_t \left(\frac{du}{dy} \right)^2 - \varepsilon - \frac{\nu}{2k} \left(\frac{\partial k}{\partial y} \right)^2 \\
 -\frac{d}{dy} \left[(v + \nu_t / \sigma_{\varepsilon}) \frac{d\varepsilon}{dy} \right] &= C_{\varepsilon 1} f_1 \nu_t \frac{\varepsilon}{k} \left(\frac{du}{dy} \right)^2 - C_{\varepsilon 2} f_2 \frac{\varepsilon^2}{k} + 2\nu \nu_t \left(\frac{\partial^2 u}{\partial y^2} \right)^2
 \end{aligned} \tag{B.26}$$

where

$$\begin{aligned}
 f_{\mu} &= \exp[-3.4 / (1 + R_t / 50)^2], & f_1 &= 1.0, & f_2 &= 1 - 0.3 \exp(-R_t^2), \\
 \nu_t &= C_{\mu} f_{\mu} \frac{k^2}{\varepsilon}, & R_t &= \frac{k^2}{\nu \varepsilon}
 \end{aligned} \tag{B.27}$$

and the closure constants are

$$C_{\mu} = 0.09, C_{\varepsilon 1} = 1.44, C_{\varepsilon 2} = 1.92, \sigma_k = 1.0, \sigma_{\varepsilon} = 1.3 \tag{B.28}$$

Using the change of variables, $h = \varepsilon$, Eq. (B.26) is a special case of Eq. (E.1) where

$$\begin{aligned}
f_v &= 0.09 f_\mu \frac{k^2}{h}, & f_u &= -\frac{1}{\rho} \frac{d\hat{p}}{dx}, & f_{k1} &= f_v, & f_{k2} &= h, & f_{k3} &= v \\
f_{h1} &= 0.1296 f_\mu k, & f_{h2} &= 1.92[1 - 0.3 \exp(-R_t^2)] \frac{h^2}{k}, & f_{h3} &= 2\nu\nu_t, \\
f_\mu &= \exp[-3.4/(1 + R_t/50)^2], & R_t &= \frac{k^2}{\nu h}
\end{aligned}
\tag{B.29}$$

Because h is not singular at the wall, the near-wall solution only needs to be used for the value of h at the wall. Therefore, we set $m_h = 0$. In the limit as $y \rightarrow 0$, the k equation reduces to

$$h_0 = 0 \tag{B.30}$$

Figures B.1–B.3 show the nondimensional results of a grid-resolution study for the model as a function of y^+ where $\mu = .001$, $\rho = .1$, $l = 0.5$, and $d\hat{p}/dx = -7.2$. The logscale axes are helpful for seeing how the model behaves near the wall.

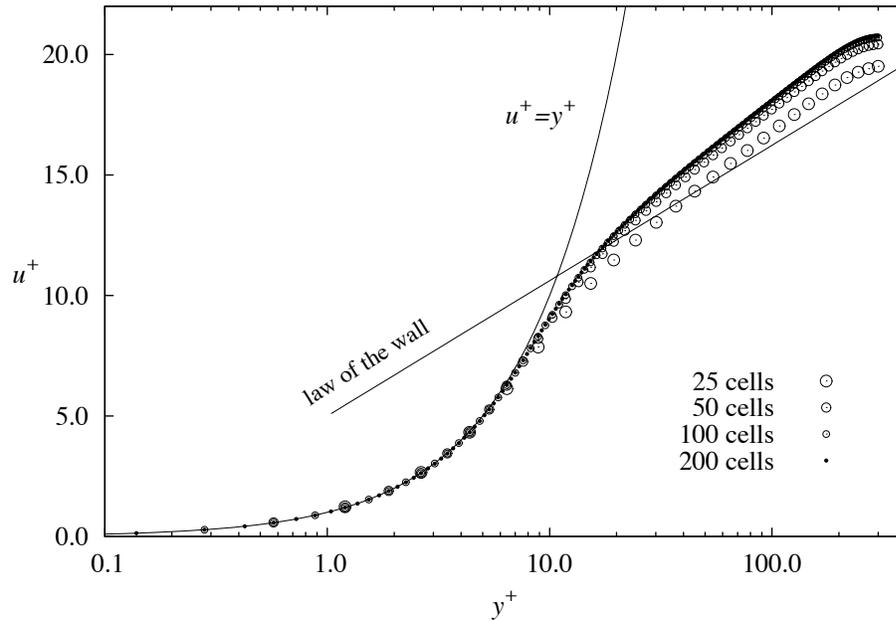


Fig. B.1 Nondimensional mean velocity results from the Launder-Sharma model.

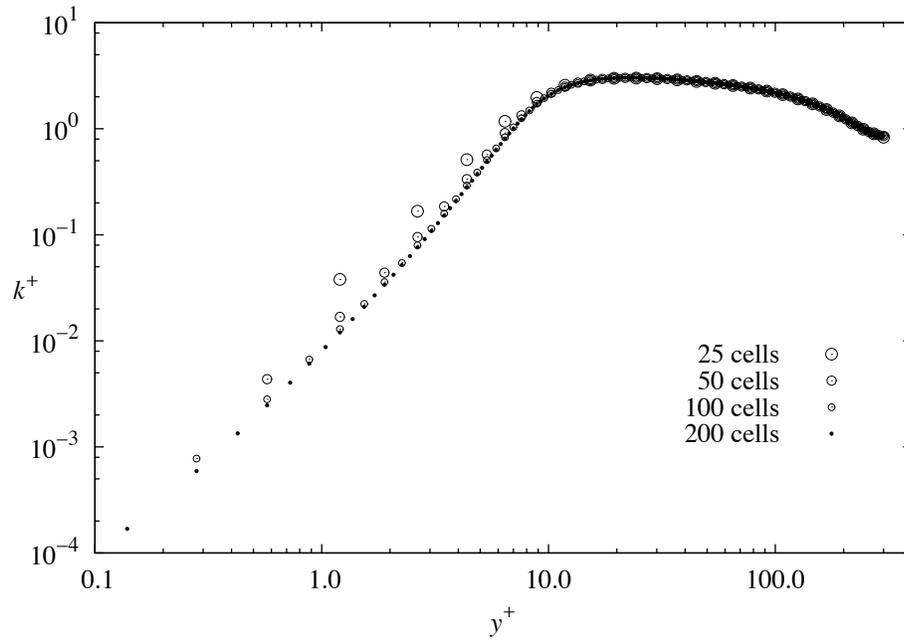


Fig. B.2 Nondimensional turbulent kinetic energy results from the Launder-Sharma model.

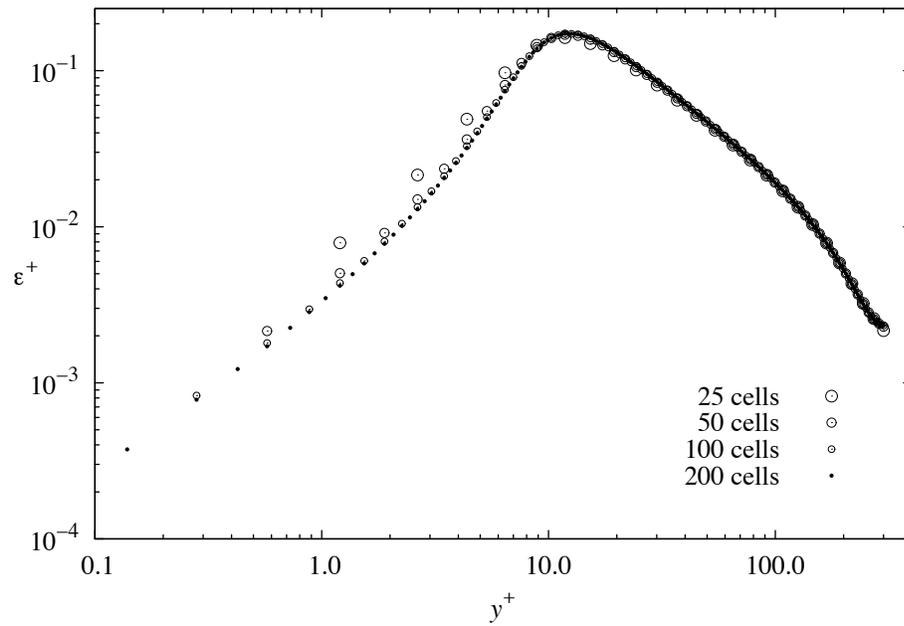


Fig. B.3 Nondimensional dissipation results from the Launder-Sharma model.

The Launder-Sharma turbulence model is a special case of Eq. (B.20) where

$$\begin{aligned}
 f_\mu &= \exp[-3.4/(1+R_t/50)^2], \quad f_1 = 1.0, \quad f_2 = 1 - 0.3 \exp(-R_t^2), \\
 R_t &= \frac{k^2}{\nu \varepsilon}, \quad \varepsilon_o = 2\nu \left(\frac{\partial \sqrt{k}}{\partial r} \right)^2 = \frac{\nu}{2k} \left(\frac{\partial k}{\partial r} \right)^2, \quad E = 2\nu \nu_t \left(\frac{\partial^2 \bar{V}_z}{\partial r^2} \right)^2, \\
 C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
 \end{aligned} \tag{B.31}$$

In nondimensional form, it is a special case of Eq. (B.22) where

$$\begin{aligned}
 f_\mu &= \exp[-3.4/(1+R_t/50)^2], \quad f_1 = 1.0, \quad f_2 = 1 - 0.3 \exp(-R_t^2), \\
 R_t &= \frac{k^{+2}}{\varepsilon^+}, \quad \varepsilon_o^+ = \frac{2}{R_\tau^2} \left(\frac{\partial \sqrt{k^+}}{\partial \hat{r}} \right)^2 = \frac{1}{2k^+ R_\tau^2} \left(\frac{\partial k^+}{\partial \hat{r}} \right)^2, \quad E^+ = \frac{2\nu^+}{R_\tau^4} \left(\frac{\partial^2 u^+}{\partial \hat{r}^2} \right)^2, \\
 C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
 \end{aligned} \tag{B.32}$$

From the differential equations, it can be shown that the near-wall asymptotic solution for ε^+ is

$$\varepsilon^+(0) = 0 \tag{B.33}$$

This model is a special case of Eq. (D.1) where

$$\begin{aligned}
 h^+ &= \varepsilon^+ \\
 f_\nu &= C_\mu f_\mu k^{+2} / \varepsilon^+ \\
 S_k &= \nu^+ \left(\frac{du^+}{d\hat{r}} \right)^2 - \varepsilon^+ - \varepsilon_o^+ \\
 S_h &= C_{\varepsilon 1} C_\mu f_\mu k^+ \left(\frac{du^+}{d\hat{r}} \right)^2 - C_{\varepsilon 2} f_2 \frac{\varepsilon^{+2}}{k^+} + E^+ \\
 f_\mu &= \exp[-3.4/(1+R_t/50)^2], \quad f_2 = 1 - 0.3 \exp(-R_t^2), \\
 R_t &= \frac{k^{+2}}{\varepsilon^+}, \quad \varepsilon_o^+ = \frac{2}{R_\tau^2} \left(\frac{\partial \sqrt{k^+}}{\partial \hat{r}} \right)^2 = \frac{1}{2k^+ R_\tau^2} \left(\frac{\partial k^+}{\partial \hat{r}} \right)^2, \quad E^+ = \frac{2\nu^+}{R_\tau^4} \left(\frac{\partial^2 u^+}{\partial \hat{r}^2} \right)^2, \\
 C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
 \end{aligned} \tag{B.34}$$

The algorithms given in Eqs. (D.18)–(D.20) can be used to solve this model where the functions are defined as

$$\begin{aligned}
R_t &= \frac{k_j^{+2}}{\varepsilon_j^+} \\
f_v &= C_\mu \exp[-3.4/(1+R_t/50)^2] k_j^{+2} / \varepsilon_j^+ \\
S_k &= v_j^+ u_j^{+ \prime 2} - \varepsilon_j^+ - \frac{k_j^{+ \prime 2}}{2k_j^+ R_\tau^2} \\
S_h &= C_{\varepsilon 1} C_\mu \exp[-3.4/(1+R_t/50)^2] k_j^+ u_j^{+ \prime 2} - C_{\varepsilon 2} [1 - 0.3 \exp(-R_t^2)] \frac{\varepsilon_j^{+2}}{k_j^+} + \frac{2v_j^+}{R_\tau^4} u_j^{+ \prime 2} \\
f_{h0} &= 0.0 \\
C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
\end{aligned} \tag{B.35}$$

and $m_h = 1$ should be used.

2. The Lam-Bremhorst k - ε Model

The Lam-Bremhorst [68] turbulence model is a special case of Eq. (E.2) where

$$\begin{aligned}
f_\mu &= [1 - \exp(-0.165R_y)]^2 [1 + 20.5/R_t], \quad f_1 = 1 + (.05/f_\mu)^3, \quad f_2 = 1 - \exp(-R_t^2), \\
v_t &= C_\mu f_\mu \frac{k^2}{\varepsilon}, \quad R_t = \frac{k^2}{\nu \varepsilon}, \quad R_y = \frac{y\sqrt{k}}{\nu}, \quad \varepsilon_o = 0, \quad E = 0, \\
C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
\end{aligned} \tag{B.36}$$

where y is the coordinate measured normal to the wall. This formulation can be written in terms of the k and ε transport equations given in Table F.1 and is a special case where

$$\begin{aligned}
f_{k1} &= 2v_t, \quad f_{k2} = \varepsilon, \\
f_{\varepsilon 1} &= 2C_{\varepsilon 1} C_\mu f_\mu [1 + (.05/f_\mu)^3] k, \quad f_{\varepsilon 2} = C_{\varepsilon 2} [1 - \exp(-R_t^2)] \frac{\varepsilon^2}{k}, \\
v_t &= C_\mu f_\mu \frac{k^2}{\varepsilon}, \quad R_t = \frac{k^2}{\nu \varepsilon}, \quad R_y = \frac{y\sqrt{k}}{\nu}, \quad f_\mu = [1 - \exp(-0.165R_y)]^2 [1 + 20.5/R_t], \\
C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
\end{aligned} \tag{B.37}$$

In the limit as $y \rightarrow 0$, the k equation reduces to give the near-wall approximation for ε

$$f_{\varepsilon 0} = \left(\frac{d^2 k}{dy^2} \right)_0 \tag{B.38}$$

This is not a boundary condition. It is simply a near-wall approximation for ε and can be applied at a smooth, no-slip wall. Some have implemented $d\varepsilon/dy(0) = 0$ as a boundary condition. However, this is mathematically incorrect and has no physical basis as pointed out in Chapter 5. Using Eqs. (F.10) and

(F.11) along with the approximations given in Eqs. (H.22) and (H.23), Eq. (B.38) can be evaluated along the south boundary as

$$\begin{aligned}
f_{\varepsilon 0} &= \left(\frac{d^2 k}{dy^2} \right)_0 = \left[\frac{d}{dy} \left(\frac{dk}{dy} \right) \right]_0 \\
&= \left[J_{X,\xi} \frac{d}{d\eta} \left(J_{X,\xi} \frac{dk}{d\eta} \right) \right]_0 \\
&= \left[J_{X,\xi}^2 \left(\frac{dJ}{d\eta} \frac{dk}{d\eta} + J \frac{d^2 k}{d\eta^2} \right) \right]_0 \\
&= J_0 x_{,\xi 0}^2 \left[\left(\frac{-8J_0 + 9J_1 - J_2}{3\Delta\eta} \right) \left(\frac{-8k_0 + 9k_1 - k_2}{3\Delta\eta} \right) + J_0 \left(\frac{72k_0 - 120k_1 + 60k_2 - 12k_3}{15\Delta\eta^2} \right) \right]
\end{aligned} \tag{B.39}$$

where the subscript 0 is the value at the wall, and the subscripts 1, 2, and 3 represent the values at the first, second, and third cells off of the wall respectively. As a final note, it is important to mention that near the wall, R_t and R_y approach zero and cause some of the functions in Eq. (B.37) to be singular. Therefore, near-wall approximations must be used. The following criteria were applied on a 32-bit computer performing double-precision computations for the case of $R_t < 20.5E-16$

$$\begin{aligned}
f_\mu &= 20.5(0.0165)^2 \frac{y^2 \varepsilon}{\nu k}, \quad \nu_t = 1.845(0.0165)^2 \frac{y^2 k}{\nu}, \\
f_{\varepsilon 1} &= \frac{0.1296(0.05)^3}{(20.5)^2 (0.0165)^4} \frac{k^3 \nu^2}{y^4 \varepsilon^2}, \quad f_{\varepsilon 2} = 1.92 \frac{k^3}{\nu^2}
\end{aligned} \tag{B.40}$$

The Lam-Bremhorst turbulence model can be written for two-dimensional fully developed channel flow as

$$\begin{aligned}
\frac{\partial}{\partial y} \left[(v + \nu_t) \frac{du}{dy} \right] &= \frac{1}{\rho} \frac{d\hat{p}}{dx} \\
-\frac{d}{dy} \left[(v + \nu_t / \sigma_k) \frac{dk}{dy} \right] &= \nu_t \left(\frac{du}{dy} \right)^2 - \varepsilon \\
-\frac{d}{dy} \left[(v + \nu_t / \sigma_\varepsilon) \frac{d\varepsilon}{dy} \right] &= C_{\varepsilon 1} f_1 \nu_t \frac{\varepsilon}{k} \left(\frac{du}{dy} \right)^2 - C_{\varepsilon 2} f_2 \frac{\varepsilon^2}{k}
\end{aligned} \tag{B.41}$$

where

$$\begin{aligned}
f_\mu &= [1 - \exp(-0.0165R_y)]^2 [1 + 20.5/R_t], & f_1 &= 1 + (.05/f_\mu)^3, & f_2 &= 1 - \exp(-R_t^2), \\
v_t &= C_\mu f_\mu \frac{k^2}{\varepsilon}, & R_t &= \frac{k^2}{\nu \varepsilon}, & R_y &= \frac{y\sqrt{k}}{\nu}
\end{aligned} \tag{B.42}$$

and the closure constants are

$$C_\mu = 0.09, C_{\varepsilon 1} = 1.44, C_{\varepsilon 2} = 1.92, \sigma_k = 1.0, \sigma_\varepsilon = 1.3 \tag{B.43}$$

Using the change of variables, $h = \varepsilon$, Eq. (B.41) is a special case of Eq. (E.1) where

$$\begin{aligned}
f_v &= 0.09 f_\mu \frac{k^2}{h}, & f_u &= -\frac{1}{\rho} \frac{d\hat{p}}{dx}, & f_{k1} &= f_v, & f_{k2} &= h, & f_{k3} &= 0 \\
f_{h1} &= 0.1296 [1 + (.05/f_\mu)^3] f_\mu k, & f_{h2} &= 1.92 [1 - \exp(-R_t^2)] \frac{h^2}{k}, & f_{h3} &= 0, \\
f_\mu &= [1 - \exp(-0.0165R_y)]^2 [1 + 20.5/R_t], & R_t &= \frac{k^2}{\nu h}, & R_y &= \frac{y\sqrt{k}}{\nu}
\end{aligned} \tag{B.44}$$

Because h is not singular at the wall, the near-wall solution only needs to be used for the value of h at the wall. Therefore, we set $m_h = 0$. In the limit as $y \rightarrow 0$, the k equation reduces to

$$h_0 = \left(\frac{d^2 k}{dy^2} \right)_0 \tag{B.45}$$

which can be evaluated from Eq. (E.49). As a final note on the Lam-Bremhorst model, it is important to mention that near the wall, R_t and R_y approach zero and cause some of the functions in Eq. (B.44) to be singular. Therefore, the near-wall approximations must be used. The following criteria were applied on a 32-bit computer performing double-precision computations for the case of $R_t < 20.5E-16$

$$\begin{aligned}
f_\mu &= 20.5(0.0165)^2 \frac{y^2 h}{\nu k}, & f_v &= 1.845(0.0165)^2 \frac{y^2 k}{\nu} \\
f_{h1} &= \frac{0.1296(0.05)^3}{(20.5)^2 (0.0165)^4} \frac{k^3 \nu^2}{y^4 h^2}, & f_{h2} &= 1.92 \frac{k^3}{\nu^2}
\end{aligned} \tag{B.46}$$

Figures B.4–B.6 show the nondimensional results of a grid-resolution study for the model as a function of y^+ where $\mu = .001$, $\rho = .1$, $l = 0.5$, and $d\hat{p}/dx = -7.2$.

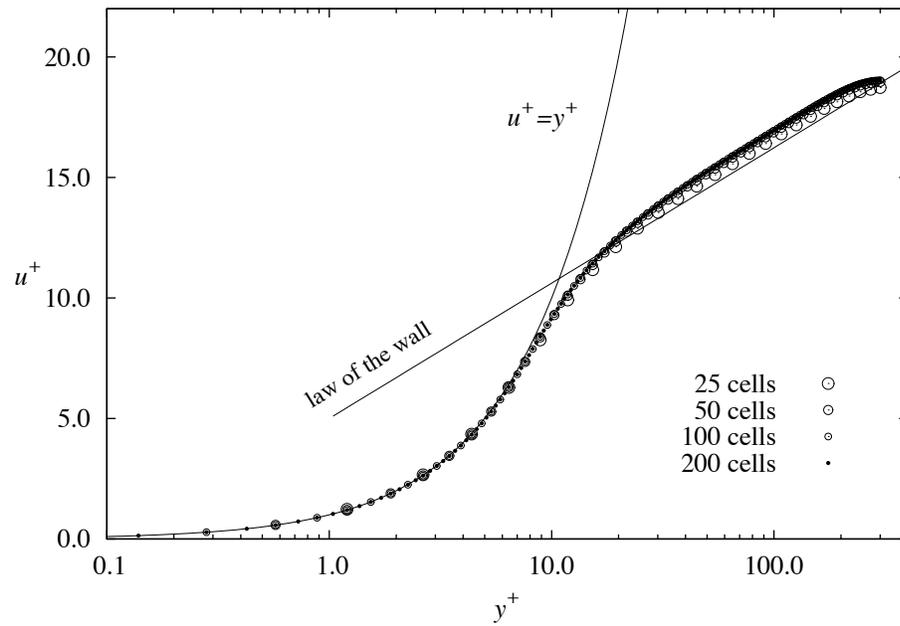


Fig. B.4 Nondimensional mean velocity results from the Lam-Bremhorst model.

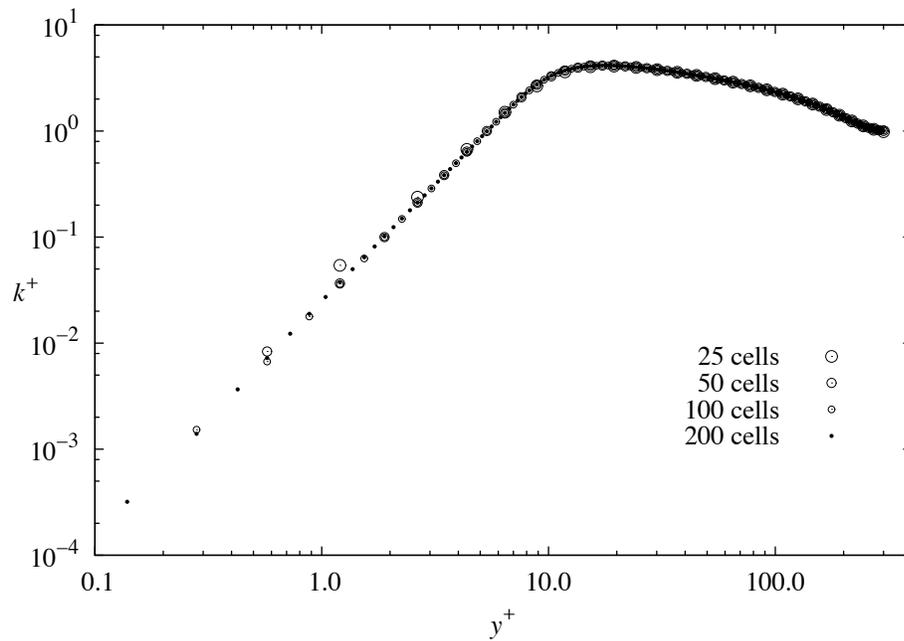


Fig. B.5 Nondimensional turbulent kinetic energy results from the Lam-Bremhorst model.

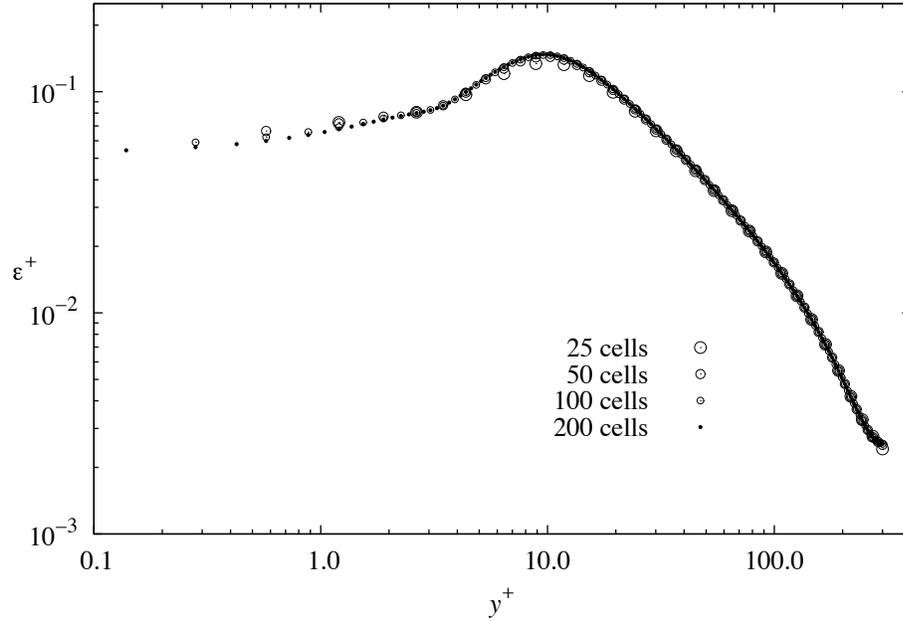


Fig. B.6 Nondimensional dissipation results from the Lam-Bremhorst model.

The Lam-Bremhorst turbulence model is a special case of Eq. (B.20) where

$$\begin{aligned}
 f_\mu &= [1 - \exp(-0.0165R_y)]^2 (1 + 20.5/R_t), \quad f_1 = 1 + (0.05/f_\mu)^3, \quad f_2 = 1 - \exp(-R_t^2), \\
 R_t &= \frac{k^2}{\nu \mathcal{E}}, \quad R_y = \frac{y\sqrt{k}}{\nu}, \quad \varepsilon_o = 0, \quad E = 0, \\
 C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
 \end{aligned}
 \tag{B.47}$$

In nondimensional form, it is a special case of Eq. (B.22) where

$$\begin{aligned}
 f_\mu &= [1 - \exp(-0.0165R_y)]^2 (1 + 20.5/R_t), \quad f_1 = 1 + (0.05/f_\mu)^3, \quad f_2 = 1 - \exp(-R_t^2), \\
 R_t &= \frac{k^{+2}}{\varepsilon^+}, \quad R_y = y^+ \sqrt{k^+}, \quad \varepsilon_o^+ = 0, \quad E^+ = 0, \\
 C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
 \end{aligned}
 \tag{B.48}$$

From the differential equations, it can be shown that the near-wall asymptotic solution for ε^+ is

$$\varepsilon^+(0) = k^{+''}(0)
 \tag{B.49}$$

where the double-prime represents a derivative with respect to y^+ . This model is a special case of Eq. (D.1)

where

$$\begin{aligned}
h^+ &= \varepsilon^+ \\
f_v &= C_\mu f_\mu k^{+2} / \varepsilon^+ \\
S_k &= \nu^+ \left(\frac{du^+}{d\hat{r}} \right)^2 - \varepsilon^+ \\
S_h &= C_{\varepsilon 1} C_\mu f_1 f_\mu k^+ \left(\frac{du^+}{d\hat{r}} \right)^2 - C_{\varepsilon 2} f_2 \frac{\varepsilon^{+2}}{k^+} \\
f_\mu &= [1 - \exp(-0.0165 R_y)]^2 (1 + 20.5/R_t), \quad f_1 = 1 + (0.05/f_\mu)^3, \\
f_2 &= 1 - \exp(-R_t^2), \quad R_t = \frac{k^{+2}}{\varepsilon^+}, \quad R_y = y^+ \sqrt{k^+}, \\
C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
\end{aligned} \tag{B.50}$$

The algorithms given in Eqs. (D.18)–(D.20) can be used to solve this model where the functions are defined as

$$\begin{aligned}
R_t &= \frac{k_j^{+2}}{\varepsilon_j^+}, \quad R_y = k_j^{+1/2} y_j^+ \\
f_v &= C_\mu R_t [1 - \exp(-0.0165 R_y)]^2 (1 + 20.5/R_t) \\
S_k &= \nu_j^+ u_j^{+2} - \varepsilon_j^+ \\
S_h &= C_{\varepsilon 1} C_\mu [1 + (0.05/f_\mu)^3] [1 - \exp(-0.0165 R_y)]^2 (1 + 20.5/R_t) k_j^+ u_j^{+2} - C_{\varepsilon 2} [1 - \exp(-R_t^2)] \frac{\varepsilon_j^{+2}}{k_j^+} \\
f_{h0} &= k_j^{+''} \\
C_\mu &= 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3
\end{aligned} \tag{B.51}$$

and $m_h = 1$ should be used. As a final note on the Lam-Bremhorst model, it is important to mention that near the wall, R_t and R_y approach zero and cause some of the functions to be singular. Therefore, near-wall approximations must be used for these functions. The leading-order terms in the near-wall Taylor series expansion for k and the near-wall asymptotic solution for ε give

$$\begin{aligned}
k^+ &= \frac{k^{+''}(0) y^{+2}}{2} + \dots \\
\varepsilon^+ &= k^{+''}(0) + \dots
\end{aligned} \tag{B.52}$$

Using these leading-order terms in the definitions of R_t and R_y shows that as $y^+ \rightarrow 0$

$$\begin{aligned}
 R_t &\equiv \frac{k^{+2}}{\varepsilon^+} = \frac{k^{+''}(0)y^{+4}}{4} \\
 R_y &\equiv k^{+1/2} y^+ = \frac{\sqrt{k^{+''}(0)y^{+2}}}{\sqrt{2}}
 \end{aligned}
 \tag{B.53}$$

This shows that as $y^+ \rightarrow 0$, $R_t \rightarrow 0$ as y^{+4} and $R_y \rightarrow 0$ as y^{+2} . Looking at the asymptotic behavior of f_μ for double-precision computations gives

$$\begin{aligned}
 \text{if } R_y < 2.71\text{E-}6 \quad f_\mu &= (0.0165R_y)^2 [1 + 20.5/R_t] \\
 \text{if } (R_t < 20.5\text{E-}15) \text{ and } (R_y < 2.71\text{E-}6) \quad f_\mu &= (0.0165R_y)^2 \frac{20.5}{R_t}
 \end{aligned}
 \tag{B.54}$$

These approximations can be used in Eq. (B.50) to give conditional statements for f_ν

$$\begin{aligned}
 \text{if } R_y < 2.71\text{E-}6 \\
 f_\mu &= (0.0165R_y)^2 [1 + 20.5/R_t] \\
 f_\nu &= C_\mu f_\mu \frac{k^{+2}}{\varepsilon^+} \\
 \text{if } (R_t < 20.5\text{E-}15) \text{ and } (R_y < 2.71\text{E-}6) \\
 f_\nu &= C_\mu (0.0165R_y)^2 \frac{20.5}{R_t} \frac{k^{+2}}{\varepsilon^+} \\
 &= 20.5C_\mu (0.0165R_y)^2
 \end{aligned}
 \tag{B.55}$$

Approximations from Eq. (B.54) can be used in Eq. (B.50) to give conditional statements for $f_1 f_\mu k^+$.

Details are shown here.

if $R_y < 2.71E-6$

$$f_1 f_\mu k^+ = [1 + (0.05/f_\mu)^3] (0.0165 R_y)^2 [1 + 20.5/R_t] k^+$$

if $(R_t < 20.5E-15)$ and $(R_y < 2.71E-6)$

$$\begin{aligned} f_1 f_\mu k^+ &= \left\{ 1 + \frac{(0.05)^3}{\left[(0.0165 R_y)^2 \frac{20.5}{R_t} \right]^3} \right\} (0.0165 R_y)^2 \frac{20.5}{R_t} k^+ \\ &= \left\{ 1 + \frac{(0.05)^3}{\left[(0.0165 k^{+1/2} y^+) ^2 \frac{20.5 \varepsilon^+}{k^+} \right]^3} \right\} (0.0165 k^{+1/2} y^+)^2 \frac{20.5}{k^+} \varepsilon^+ k^+ \\ &= (0.0165)^2 y^{+2} 20.5 \varepsilon^+ \left\{ 1 + \frac{(0.05)^3 k^{+3}}{(0.0165)^6 y^{+6} (20.5)^3 \varepsilon^{+3}} \right\} \\ &= (0.0165)^2 y^{+2} 20.5 \varepsilon^+ \left\{ 1 + \frac{(0.05)^3 \left(\frac{k^{+''}(0) y^{+2}}{2} \right)^3}{(0.0165)^6 y^{+6} (20.5)^3 \left(k^{+''}(0) \right)^3} \right\} \\ &= (0.0165)^2 y^{+2} 20.5 \varepsilon^+ \left\{ 1 + \frac{(0.05)^3}{8(0.0165)^6 (20.5)^3} \right\} \end{aligned} \tag{B.56}$$

Likewise, double-precision computations gives

$$\text{if } R_t < 2.11E-4 \quad [1 - \exp(-R_t^2)] \frac{\varepsilon^{+2}}{k^+} = k^{+3} \tag{B.57}$$

II. The $k-\omega$ Model

A. The General $k-\omega$ Model Equations

The $k-\omega$ model was first developed by Kolmogorov [16] and has been revised by many others including Saffman [103], Wilcox [35,39,40,45], Speziale et al. [34], Peng et al. [42], Kok [43], and Hellsten [44]. In its general, steady-state, incompressible form, the complete $k-\omega$ model includes the continuity equation, the RANS equations, the turbulent kinetic energy equation, the dissipation frequency transport equation, and the closing kinematic viscosity equation

$$\begin{aligned}
\nabla \cdot \mathbf{V} &= 0 \\
(\mathbf{V} \cdot \nabla) \mathbf{V} &= -\nabla \hat{p} / \rho + \nabla \cdot [2(\nu + \nu_t) \bar{\bar{\mathbf{S}}}(\mathbf{V})] \\
(\bar{\mathbf{V}} \cdot \nabla) k &= 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - C_\mu k \omega + \nabla \cdot [(\nu + \nu_t / \sigma_k) \nabla k] \\
(\bar{\mathbf{V}} \cdot \nabla) \omega &= 2C_{\omega 1} \nu_t \frac{\omega}{k} \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - C_{\omega 2} \omega^2 + \nabla \cdot [(\nu + \nu_t / \sigma_\omega) \nabla \omega] \\
\nu_t &= \frac{k}{\omega}
\end{aligned} \tag{B.58}$$

where C_μ , $C_{\omega 1}$, $C_{\omega 2}$, σ_k , and σ_ω are the closure coefficients. Because this model does not naturally exhibit the correct behavior in the near-wall region, functions are typically included that are similar to the wall damping functions employed in the k - ε model. These damping functions modify the equations above to yield a variation of the turbulent kinetic energy equation, the dissipation frequency transport equation, and the closing kinematic viscosity equation

$$\begin{aligned}
(\bar{\mathbf{V}} \cdot \nabla) k &= 2\nu_t \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - C_\mu f_k k \omega + \nabla \cdot [(\nu + \nu_t / \sigma_k) \nabla k] \\
(\bar{\mathbf{V}} \cdot \nabla) \omega &= 2C_{\omega 1} f_1 \nu_t \frac{\omega}{k} \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) - C_{\omega 2} f_2 \omega^2 + \nabla \cdot [(\nu + \nu_t / \sigma_\omega) \nabla \omega] \\
\nu_t &= f_\mu \frac{k}{\omega}
\end{aligned} \tag{B.59}$$

B. Fully Developed Channel Flow

The k and ω transport equations can be written for 2-D flow in Cartesian coordinates as

$$\begin{aligned}
\frac{\partial(\bar{V}_x k)}{\partial x} + \frac{\partial(\bar{V}_y k)}{\partial y} &= 2\nu_t \left[\left(\frac{\partial \bar{V}_x}{\partial x} \right)^2 + \left(\frac{\partial \bar{V}_y}{\partial y} \right)^2 + \frac{1}{2} \left(\frac{\partial \bar{V}_y}{\partial x} + \frac{\partial \bar{V}_x}{\partial y} \right)^2 \right] - C_\mu f_k k \omega \\
&+ \frac{\partial}{\partial x} \left[(\nu + \nu_t / \sigma_k) \frac{\partial k}{\partial x} \right] + \frac{\partial}{\partial y} \left[(\nu + \nu_t / \sigma_k) \frac{\partial k}{\partial y} \right]
\end{aligned} \tag{B.60}$$

$$\begin{aligned}
\frac{\partial(\bar{V}_x \omega)}{\partial x} + \frac{\partial(\bar{V}_y \omega)}{\partial y} &= 2C_{\omega 1} f_1 \nu_t \frac{\omega}{k} \left[\left(\frac{\partial \bar{V}_x}{\partial x} \right)^2 + \left(\frac{\partial \bar{V}_y}{\partial y} \right)^2 + \frac{1}{2} \left(\frac{\partial \bar{V}_y}{\partial x} + \frac{\partial \bar{V}_x}{\partial y} \right)^2 \right] - C_{\omega 2} f_2 \omega^2 \\
&+ \frac{\partial}{\partial x} \left[(\nu + \nu_t / \sigma_\omega) \frac{\partial \omega}{\partial x} \right] + \frac{\partial}{\partial y} \left[(\nu + \nu_t / \sigma_\omega) \frac{\partial \omega}{\partial y} \right]
\end{aligned} \tag{B.61}$$

For fully developed channel flow, the transport property gradients with respect to x are much smaller than those with respect to y . Therefore, eliminating the x -derivatives and using Eq. (C.20), gives the two-dimensional fully developed k and ω transport equations

$$-\frac{\partial}{\partial y} \left[(v + \nu_t / \sigma_k) \frac{\partial k}{\partial y} \right] = \nu_t \left(\frac{\partial \bar{V}_x}{\partial y} \right)^2 - C_\mu f_k k \omega \quad (\text{B.62})$$

$$-\frac{\partial}{\partial y} \left[(v + \nu_t / \sigma_\omega) \frac{\partial \omega}{\partial y} \right] = C_{\omega 1} f_1 \nu_t \frac{\omega}{k} \left(\frac{\partial \bar{V}_x}{\partial y} \right)^2 - C_{\omega 2} f_2 \omega^2 \quad (\text{B.63})$$

Using the nondimensional definitions

$$k^+ \equiv \frac{k}{u_\tau^2}, \quad \omega^+ \equiv \frac{\omega \nu}{u_\tau^2} \quad (\text{B.64})$$

and substituting in the results of Eq. (C.31), gives the nondimensional fully developed k - ω model

$$\frac{d}{dy^+} \left[(1 + \nu^+ / \sigma_k) \frac{dk^+}{dy^+} \right] = C_\mu f_k k^+ \omega^+ - \nu^+ \left(\frac{R_\tau - y^+}{R_\tau (1 + \nu^+)} \right)^2 \quad (\text{B.65})$$

$$\frac{\partial}{\partial y^+} \left[(1 + \nu^+ / \sigma_\omega) \frac{\partial \omega^+}{\partial y^+} \right] = C_{\omega 2} f_2 \omega^{+2} - C_{\omega 1} f_1 \nu^+ \frac{\omega^+}{k^+} \left(\frac{R_\tau - y^+}{R_\tau (1 + \nu^+)} \right)^2 \quad (\text{B.66})$$

$$\nu^+ = f_\mu \frac{k^+}{\omega^+} \quad (\text{B.67})$$

In order to close this formulation, the damping functions f_μ , f_1 , f_2 , and f_k , must be specified. Various functions have been proposed, and the subsequent sections will discuss one such proposal. In addition to the definitions of the damping functions, four boundary conditions must be specified. These are

$$\frac{dk^+}{dy^+}(R_\tau) = 0, \quad \frac{d\omega^+}{dy^+}(R_\tau) = 0, \quad k^+(0) = 0, \quad \frac{dk^+}{dy^+}(0) = 0 \quad (\text{B.68})$$

Again, the final boundary condition in Eq. (B.68) is not widely recognized in the literature, and the near-wall behavior of ω is often referred to as a boundary condition. This is incorrect as discussed in Chapter 5.

C. Fully Developed Pipe Flow

The k - ω model includes an expression for the kinematic eddy viscosity, the turbulent-kinetic-energy equation, and the dissipation-frequency-transport equation

$$\begin{aligned}
v_t &= f_\mu k / \omega \\
(\bar{\mathbf{V}} \cdot \nabla)k &= 2\nu_t \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - C_\mu f_k k \omega + \nabla \cdot [(v + \nu_t / \sigma_k) \nabla k] \\
(\bar{\mathbf{V}} \cdot \nabla)\omega &= 2C_{\omega 1} f_1 \nu_t \frac{\omega}{k} \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) - C_{\omega 2} f_2 \omega^2 + \nabla \cdot [(v + \nu_t / \sigma_\omega) \nabla \omega]
\end{aligned} \tag{B.69}$$

The turbulent-kinetic-energy and dissipation-frequency transport equations can be written in cylindrical coordinates as

$$\begin{aligned}
\bar{V}_r \frac{\partial k}{\partial r} + \frac{\bar{V}_\theta}{r} \frac{\partial k}{\partial \theta} + \bar{V}_z \frac{\partial k}{\partial z} &= 2\nu_t \left[\left(\frac{\partial \bar{V}_r}{\partial r} \right)^2 + \left(\frac{1}{r} \frac{\partial \bar{V}_\theta}{\partial \theta} + \frac{\bar{V}_r}{r} \right)^2 + \left(\frac{\partial \bar{V}_z}{\partial z} \right)^2 \right. \\
&\quad \left. + \frac{1}{2} \left(r \frac{\partial (\bar{V}_\theta / r)}{\partial r} + \frac{1}{r} \frac{\partial \bar{V}_r}{\partial \theta} \right)^2 + \frac{1}{2} \left(\frac{1}{r} \frac{\partial \bar{V}_z}{\partial \theta} + \frac{\partial \bar{V}_\theta}{\partial z} \right)^2 + \frac{1}{2} \left(\frac{\partial \bar{V}_r}{\partial z} + \frac{\partial \bar{V}_z}{\partial r} \right)^2 \right] \\
&\quad - C_\mu f_k k \omega + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + \nu_t / \sigma_k) r \frac{\partial k}{\partial r} \right] + \frac{1}{r} \frac{\partial}{\partial \theta} \left[(v + \nu_t / \sigma_k) \frac{1}{r} \frac{\partial k}{\partial \theta} \right] \\
&\quad + \frac{\partial}{\partial z} \left[(v + \nu_t / \sigma_k) \frac{\partial k}{\partial z} \right]
\end{aligned} \tag{B.70}$$

$$\begin{aligned}
\bar{V}_r \frac{\partial \omega}{\partial r} + \frac{\bar{V}_\theta}{r} \frac{\partial \omega}{\partial \theta} + \bar{V}_z \frac{\partial \omega}{\partial z} &= 2C_{\omega 1} f_1 \nu_t \frac{\omega}{k} \left[\left(\frac{\partial \bar{V}_r}{\partial r} \right)^2 + \left(\frac{1}{r} \frac{\partial \bar{V}_\theta}{\partial \theta} + \frac{\bar{V}_r}{r} \right)^2 + \left(\frac{\partial \bar{V}_z}{\partial z} \right)^2 \right. \\
&\quad \left. + \frac{1}{2} \left(r \frac{\partial (\bar{V}_\theta / r)}{\partial r} + \frac{1}{r} \frac{\partial \bar{V}_r}{\partial \theta} \right)^2 + \frac{1}{2} \left(\frac{1}{r} \frac{\partial \bar{V}_z}{\partial \theta} + \frac{\partial \bar{V}_\theta}{\partial z} \right)^2 + \frac{1}{2} \left(\frac{\partial \bar{V}_r}{\partial z} + \frac{\partial \bar{V}_z}{\partial r} \right)^2 \right] \\
&\quad - C_{\omega 2} f_2 \omega^2 + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + \nu_t / \sigma_\omega) r \frac{\partial \omega}{\partial r} \right] + \frac{1}{r} \frac{\partial}{\partial \theta} \left[(v + \nu_t / \sigma_\omega) \frac{1}{r} \frac{\partial \omega}{\partial \theta} \right] \\
&\quad + \frac{\partial}{\partial z} \left[(v + \nu_t / \sigma_\omega) \frac{\partial \omega}{\partial z} \right]
\end{aligned} \tag{B.71}$$

Applying the simplifications for fully developed flow in a pipe, these equations can be written as

$$\begin{aligned}
\bar{V}_r \frac{dk}{dr} &= 2\nu_t \left[\left(\frac{d\bar{V}_r}{dr} \right)^2 + \left(\frac{\bar{V}_r}{r} \right)^2 + \frac{1}{2} \left(\frac{d\bar{V}_z}{dr} \right)^2 \right] - C_\mu f_k k \omega + \frac{1}{r} \frac{d}{dr} \left[(v + \nu_t / \sigma_k) r \frac{dk}{dr} \right] \\
\bar{V}_r \frac{d\omega}{dr} &= 2C_{\omega 1} f_1 \nu_t \frac{\omega}{k} \left[\left(\frac{d\bar{V}_r}{dr} \right)^2 + \left(\frac{\bar{V}_r}{r} \right)^2 + \frac{1}{2} \left(\frac{d\bar{V}_z}{dr} \right)^2 \right] - C_{\omega 2} f_2 \omega^2 + \frac{1}{r} \frac{d}{dr} \left[(v + \nu_t / \sigma_\omega) r \frac{d\omega}{dr} \right]
\end{aligned} \tag{B.72}$$

Applying Eq. (C.59) gives

$$\begin{aligned}
\frac{1}{r} \frac{d}{dr} \left[-(\nu + \nu_t / \sigma_k) r \frac{dk}{dr} \right] &= \nu_t \left(\frac{d\bar{V}_z}{dr} \right)^2 - C_\mu f_k k \omega \\
\frac{1}{r} \frac{d}{dr} \left[-(\nu + \nu_t / \sigma_\omega) r \frac{d\omega}{dr} \right] &= C_{\omega 1} f_1 \nu_t \frac{\omega}{k} \left(\frac{d\bar{V}_z}{dr} \right)^2 - C_{\omega 2} f_2 \omega^2
\end{aligned} \tag{B.73}$$

The k - ω formulation with the remaining boundary conditions can be written as

$$\begin{aligned}
\frac{d\bar{V}_z}{dr} &= -\frac{u_\tau^2}{(\nu + \nu_t)} \frac{r}{R} \\
\nu_t &= f_\mu k / \omega \\
\frac{1}{r} \frac{d}{dr} \left[-(\nu + \nu_t / \sigma_k) r \frac{dk}{dr} \right] &= \nu_t \left(\frac{d\bar{V}_z}{dr} \right)^2 - C_\mu f_k k \omega \\
\frac{1}{r} \frac{d}{dr} \left[-(\nu + \nu_t / \sigma_\omega) r \frac{d\omega}{dr} \right] &= C_{\omega 1} f_1 \nu_t \frac{\omega}{k} \left(\frac{d\bar{V}_z}{dr} \right)^2 - C_{\omega 2} f_2 \omega^2 \\
\bar{V}_z(R) = 0, \quad k(R) = 0, \quad \frac{dk}{dr}(R) = 0, \quad \frac{dk}{dr}(0) = 0, \quad \frac{d\omega}{dr}(0) = 0
\end{aligned} \tag{B.74}$$

Applying the nondimensional parameter definitions

$$k^+ \equiv \frac{k}{u_\tau^2}, \quad \omega^+ \equiv \frac{\omega \nu}{u_\tau^2} \tag{B.75}$$

with those given in Eq. (C.68), the formulation can be written as

$$\begin{aligned}
\frac{du^+}{d\hat{r}} &= -\frac{R_\tau \hat{r}}{(1 + \nu^+)} \\
\nu^+ &= f_\mu k^+ / \omega^+ \\
\frac{1}{\hat{r}} \frac{d}{d\hat{r}} \left[-(1 + \nu^+ / \sigma_k) \hat{r} \frac{dk^+}{d\hat{r}} \right] &= \nu^+ \left(\frac{du^+}{d\hat{r}} \right)^2 - C_\mu f_k R_\tau^2 k^+ \omega^+ \\
\frac{1}{\hat{r}} \frac{d}{d\hat{r}} \left[-(1 + \nu^+ / \sigma_\omega) \hat{r} \frac{d\omega^+}{d\hat{r}} \right] &= C_{\omega 1} f_1 f_\mu \left(\frac{du^+}{d\hat{r}} \right)^2 - C_{\omega 2} f_2 R_\tau^2 \omega^{+2} \\
u^+(1) = 0, \quad k^+(1) = 0, \quad \frac{dk^+}{d\hat{r}}(1) = 0, \quad \frac{dk^+}{d\hat{r}}(0) = 0, \quad \frac{d\omega^+}{d\hat{r}}(0) = 0
\end{aligned} \tag{B.76}$$

D. Sample Models

1. The Traditional k - ω Model

The traditional k - ω turbulence model is a special case of Eq. (B.59) where

$$\begin{aligned}
 f_\mu = 1.0, \quad f_k = 1.0, \quad f_1 = 1.0, \quad f_2 = 1.0, \\
 C_\mu = 0.09, \quad C_{\omega 1} = 0.52, \quad C_{\omega 2} = 0.072, \quad \sigma_k = 2.0, \quad \sigma_\omega = 2.0
 \end{aligned}
 \tag{B.77}$$

This formulation can be written in terms of the k and ω transport equations given in Table F.1 and is a special case where

$$\begin{aligned}
 f_{k1} = 2\nu_t, \quad f_{k2} = C_\mu k\omega, \quad f_{\omega 1} = 2C_{\omega 1}, \quad f_{\omega 2} = C_{\omega 2}\omega^2, \quad \nu_t = \frac{k}{\omega}, \\
 C_\mu = 0.09, \quad C_{\omega 1} = 0.52, \quad C_{\omega 2} = 0.072, \quad \sigma_k = 2.0, \quad \sigma_\omega = 2.0
 \end{aligned}
 \tag{B.78}$$

In the limit as $y \rightarrow 0$, ω becomes singular. The near-wall asymptotic behavior for ω can be written

$$f_{\omega 0} = \frac{6\nu}{0.072y^2}
 \tag{B.79}$$

where y is the coordinate measured normal to the wall. Fluent uses this near-wall approximation for the first cell off of a smooth wall.

The traditional k - ω model can be written for two-dimensional fully developed channel flow as

$$\begin{aligned}
 \frac{d}{dy} \left[(\nu + \nu_t) \left(\frac{du}{dy} \right) \right] &= \frac{1}{\rho} \frac{d\hat{p}}{dx} \\
 -\frac{d}{dy} \left[(\nu + \nu_t / \sigma_k) \frac{dk}{dy} \right] &= \nu_t \left(\frac{du}{dy} \right)^2 - C_\mu k\omega \\
 -\frac{d}{dy} \left[(\nu + \nu_t / \sigma_\omega) \frac{d\omega}{dy} \right] &= C_{\omega 1} \nu_t \frac{\omega}{k} \left(\frac{du}{dy} \right)^2 - C_{\omega 2} \omega^2
 \end{aligned}
 \tag{B.80}$$

where

$$\nu_t = \frac{k}{\omega}
 \tag{B.81}$$

and the closure constants are

$$C_\mu = 0.09, \quad C_{\omega 1} = 0.52, \quad C_{\omega 2} = 0.072, \quad \sigma_k = 2.0, \quad \sigma_\omega = 2.0
 \tag{B.82}$$

Using the change of variables, $h = \omega$, Eq. (B.80) is a special case of Eq. (E.1) where

$$\begin{aligned}
 f_v = \frac{k}{h}, \quad f_u = -\frac{1}{\rho} \frac{d\hat{p}}{dx}, \quad f_{k1} = f_v, \quad f_{k2} = 0.09kh, \quad f_{k3} = 0 \\
 f_{h1} = 0.52, \quad f_{h2} = 0.072h^2, \quad f_{h3} = 0
 \end{aligned}
 \tag{B.83}$$

Because h is singular at the wall, the near-wall solution must be used for the value of h near the wall. In the limit as $y \rightarrow 0$, the near-wall asymptotic behavior can be written for ω in the form of Eq. (E.34) as

$$f_{h0} = \frac{6\nu}{0.072y_j^2} \quad (\text{B.84})$$

Fluent uses Eq. (B.84) for the first node off of a smooth wall. This is equivalent to setting $m_h = 1$. Figures B.7–B.9 show the nondimensional results of a grid-resolution study for the model as a function of y^+ where $\mu = .001$, $\rho = .1$, $l = 0.5$, and $d\hat{p}/dx = -7.2$.

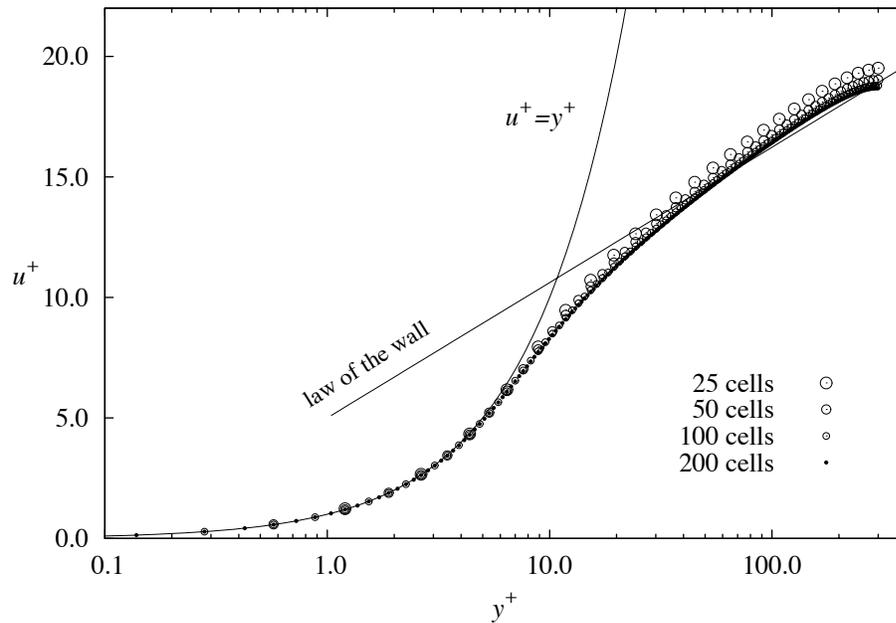


Fig. B.7 Nondimensional mean velocity results from the traditional k - ω model.

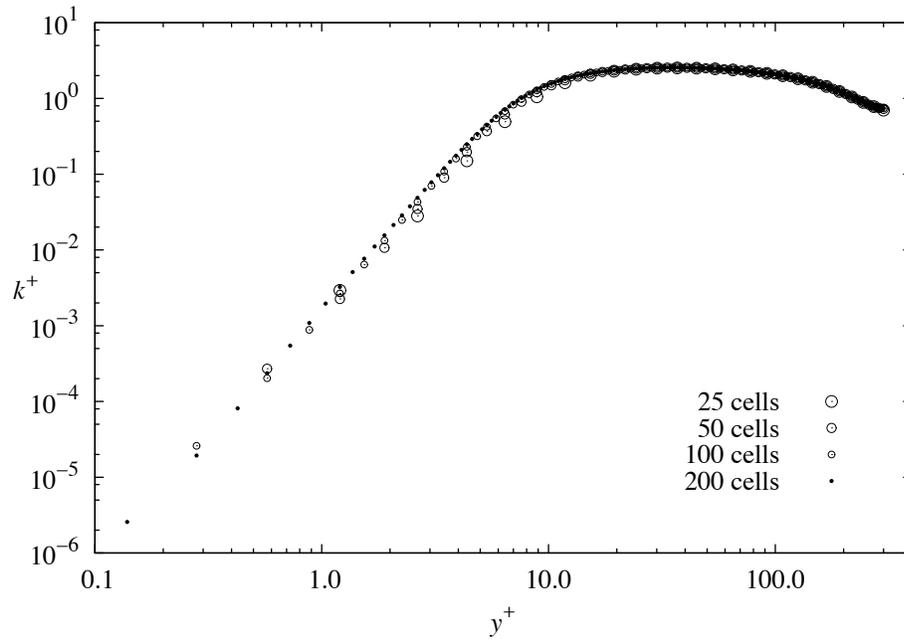


Fig. B.8 Nondimensional turbulent kinetic energy results from the traditional $k-\omega$ model.

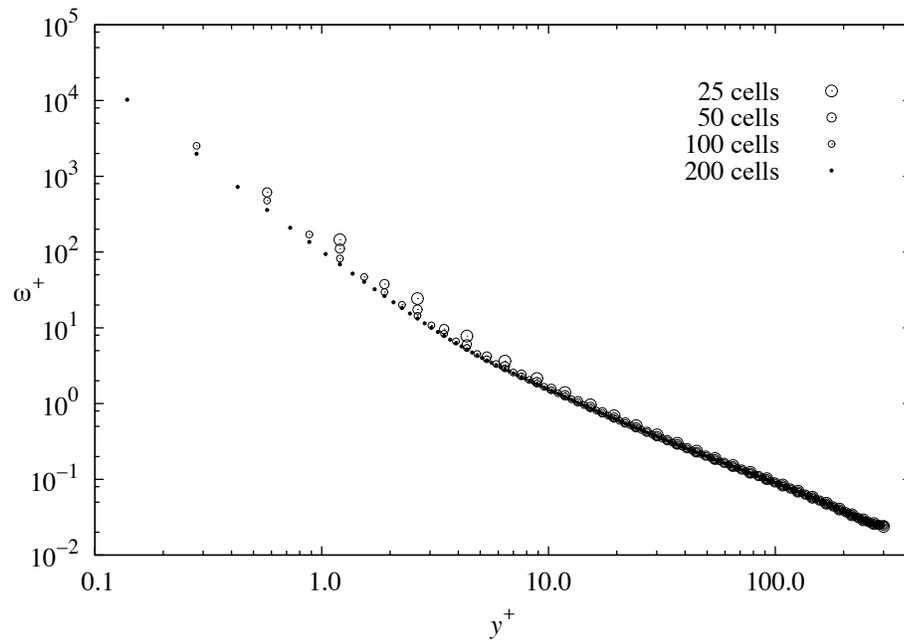


Fig. B.9 Nondimensional dissipation frequency results from the traditional $k-\omega$ model.

2. The Wilcox 1998 k - ω Model

The Wilcox 1998 k - ω model [45] including shear-flow corrections is a special case of Eq. (B.59)

where

$$\begin{aligned}
 f_\mu &= \frac{0.024 + R_t/6}{1 + R_t/6}, \quad f_k = \frac{4/15 + (R_t/8)^4}{1 + (R_t/8)^4} \begin{cases} 1, & \psi_k \leq 0 \\ \frac{1 + 680\psi_k^2}{1 + 400\psi_k^2}, & \psi_k > 0 \end{cases}, \\
 f_1 &= \frac{1/9 + R_t/2.95}{f_\mu(1 + R_t/2.95)}, \quad f_2 = 1.0, \quad \psi_k = \frac{\nabla k \cdot \nabla \omega}{\omega^3}, \quad R_t = \frac{k}{\nu \omega} \\
 C_\mu &= 0.09, \quad C_{\omega 1} = 0.52, \quad C_{\omega 2} = 0.072, \quad \sigma_k = 2.0, \quad \sigma_\omega = 2.0
 \end{aligned} \tag{B.85}$$

This formulation can be written in terms of the k and ω transport equations given in Table F.1 and is a special case where

$$\begin{aligned}
 f_{k1} &= 2\nu_t, \quad f_{k2} = C_\mu k \omega \frac{4/15 + (R_t/8)^4}{1 + (R_t/8)^4} \begin{cases} 1, & \psi_k \leq 0 \\ \frac{1 + 680\psi_k^2}{1 + 400\psi_k^2}, & \psi_k > 0 \end{cases}, \\
 f_{\omega 1} &= 2C_{\omega 1} \frac{1/9 + R_t/2.95}{1 + R_t/2.95}, \quad f_{\omega 2} = C_{\omega 2} \omega^2, \quad \nu_t = \frac{k}{\omega} \left(\frac{0.024 + R_t/6}{1 + R_t/6} \right), \\
 \psi_k &= \frac{\nabla k \cdot \nabla \omega}{\omega^3}, \quad R_t = \frac{k}{\nu \omega}, \\
 C_\mu &= 0.09, \quad C_{\omega 1} = 0.52, \quad C_{\omega 2} = 0.072, \quad \sigma_k = 2.0, \quad \sigma_\omega = 2.0
 \end{aligned} \tag{B.86}$$

In the limit as $y \rightarrow 0$, ω becomes singular. The near-wall asymptotic behavior for ω can be written

$$f_{\omega 0} = \frac{6\nu}{0.072y^2} \tag{B.87}$$

where y is the coordinate measured normal to the wall. Fluent uses this near-wall approximation for the first cell off of a smooth wall. Wilcox [86] suggests using this leading-order solution for the first 7 to 10 grid points off the surface to avoid numerical errors that can develop from using the central differencing scheme near a wall boundary. Wilcox states that this approximation is only good for grid points where $y^+ < 2.5$. Therefore, it is recommended that the grid be fine enough to ensure that at least 7 grid points are within this constraint.

As an alternative, Wilcox suggests using the slightly rough wall boundary condition

$$\omega(x,0) = \frac{u_t^2 2500}{\nu k_s^{+2}}, \quad k_s^+ < 25 \quad (\text{B.88})$$

where

$$k_s^+ \equiv \frac{u_t k_s}{\nu} \quad (\text{B.89})$$

is the scaled surface roughness height. The condition $k_s^+ < 5$ gives results for a hydraulically smooth surface.

Fluent 6.3 [85] implements the boundary conditions a similar way. Fluent sets the first node off of the wall, y_1 , to

$$\omega(x, y_1) = \min \left(\frac{u_t^2 2500}{\nu k_s^{+2}}, \frac{6\nu}{C_{\omega 2} y^2} \right), \quad k_s^+ < 25 \quad (\text{B.90})$$

where

$$k_s^+ = \max \left(1.0, \frac{u_t k_s}{\nu} \right) \quad (\text{B.91})$$

The current research is not concerned about rough surfaces. Therefore, the boundary conditions for rough walls are not included here.

The shear-flow corrections included in this model are dependent on the quantity

$$\psi_k = \frac{\nabla k \cdot \nabla \omega}{\omega^3} \quad (\text{B.92})$$

This scalar quantity can be written in Cartesian coordinates as

$$\psi_k = \frac{1}{\omega^3} \begin{Bmatrix} \frac{\partial k}{\partial x} \\ \frac{\partial k}{\partial y} \end{Bmatrix} \cdot \begin{Bmatrix} \frac{\partial \omega}{\partial x} \\ \frac{\partial \omega}{\partial y} \end{Bmatrix} = \frac{1}{\omega^3} \left[\frac{\partial k}{\partial x} \frac{\partial \omega}{\partial x} + \frac{\partial k}{\partial y} \frac{\partial \omega}{\partial y} \right] \quad (\text{B.93})$$

Applying Eqs. (F.10) and (F.11) gives

$$\begin{aligned} \psi_k &= \frac{1}{\omega^3} \left[J_{y,\eta} \frac{\partial k}{\partial \xi} J_{y,\eta} \frac{\partial \omega}{\partial \xi} + J_{x,\xi} \frac{\partial k}{\partial \eta} J_{x,\xi} \frac{\partial \omega}{\partial \eta} \right] \\ &= \frac{J^2}{\omega^3} \left[y_{,\eta}^2 \frac{\partial k}{\partial \xi} \frac{\partial \omega}{\partial \xi} + x_{,\xi}^2 \frac{\partial k}{\partial \eta} \frac{\partial \omega}{\partial \eta} \right] \end{aligned} \quad (\text{B.94})$$

This is discretized using the second-order differencing schemes explained in Appendix H.

The Wilcox 1998 k - ω model with wall damping functions without shear flow corrections can be written for two-dimensional fully developed channel flow as

$$\begin{aligned} \frac{d}{dy} \left[(v + v_t) \left(\frac{du}{dy} \right) \right] &= \frac{1}{\rho} \frac{d\hat{p}}{dx} \\ -\frac{d}{dy} \left[(v + v_t / \sigma_k) \frac{dk}{dy} \right] &= v_t \left(\frac{du}{dy} \right)^2 - C_\mu f_k k \omega \\ -\frac{d}{dy} \left[(v + v_t / \sigma_\omega) \frac{d\omega}{dy} \right] &= C_{\omega 1} f_1 v_t \frac{\omega}{k} \left(\frac{du}{dy} \right)^2 - C_{\omega 2} f_2 \omega^2 \end{aligned} \quad (\text{B.95})$$

where

$$\begin{aligned} f_\mu &= \frac{0.024 + R_t/6}{1 + R_t/6}, \quad f_1 = \frac{1/9 + R_t/2.95}{f_\mu(1 + R_t/2.95)}, \quad f_2 = 1.0, \\ f_k &= \frac{4/15 + (R_t/8)^4}{1 + (R_t/8)^4}, \quad v_t = f_\mu \frac{k}{\omega}, \quad R_t = \frac{k}{v\omega} \end{aligned} \quad (\text{B.96})$$

and the closure constants are

$$C_\mu = 0.09, \quad C_{\omega 1} = 0.52, \quad C_{\omega 2} = 0.072, \quad \sigma_k = 2.0, \quad \sigma_\omega = 2.0 \quad (\text{B.97})$$

Using the change of variables, $h = \omega$, Eq. (B.95) is a special case of Eq. (E.1) where

$$\begin{aligned} f_v &= \frac{k}{h} \left(\frac{0.024 + R_t/6}{1 + R_t/6} \right), \quad f_u = -\frac{1}{\rho} \frac{d\hat{p}}{dx}, \quad f_{k1} = f_v, \quad f_{k2} = 0.09 \frac{4/15 + (R_t/8)^4}{1 + (R_t/8)^4} kh, \quad f_{k3} = 0 \\ f_{h1} &= 0.52 \frac{1/9 + R_t/2.95}{(1 + R_t/2.95)}, \quad f_{h2} = 0.072h^2, \quad f_{h3} = 0, \quad R_t = \frac{k}{vh} \end{aligned} \quad (\text{B.98})$$

Because h is singular at the wall, the near-wall solution must be used for the value of h near the wall. In the limit as $y \rightarrow 0$, the near-wall asymptotic behavior can be written for ω in the form of Eq. (E.34) as

$$f_{h0} = \frac{6v}{0.072y_j^2} \quad (\text{B.99})$$

Fluent uses Eq. (B.99) for the first node off of a smooth wall. This is equivalent to setting $m_h = 1$. Figures B.10–B.12 show the nondimensional results of a grid-resolution study for the model as a function of y^+ where $\mu = .001$, $\rho = .1$, $l = 0.5$, and $d\hat{p}/dx = -7.2$.

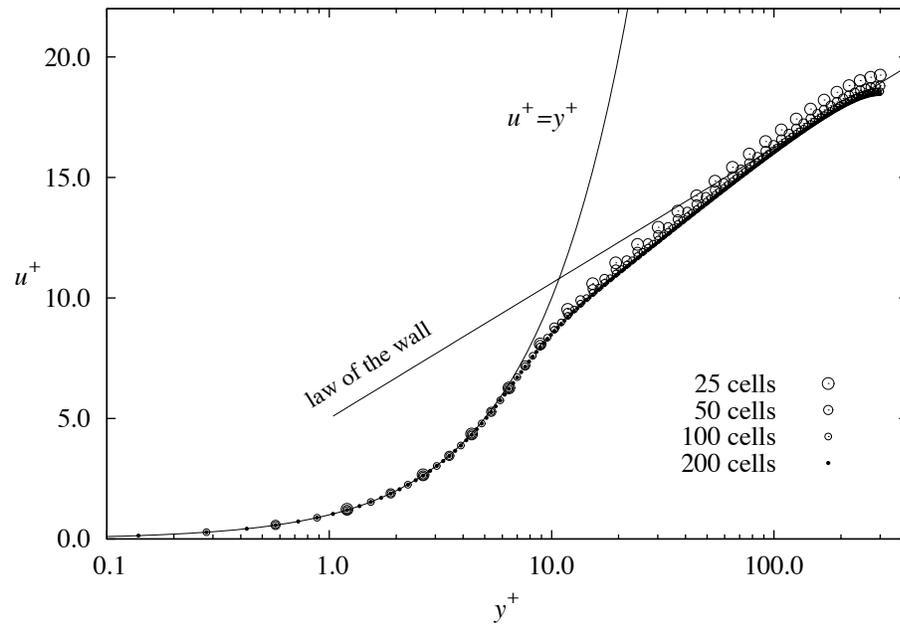


Fig. B.10 Nondimensional mean velocity results from the Wilcox 1998 model.

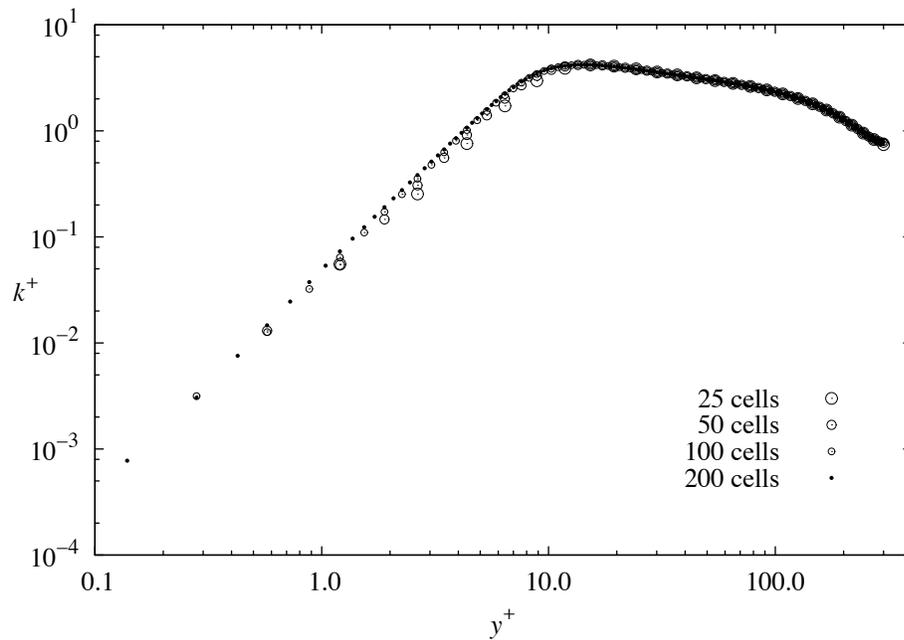


Fig. B.11 Nondimensional turbulent kinetic energy results from the Wilcox 1998 model.

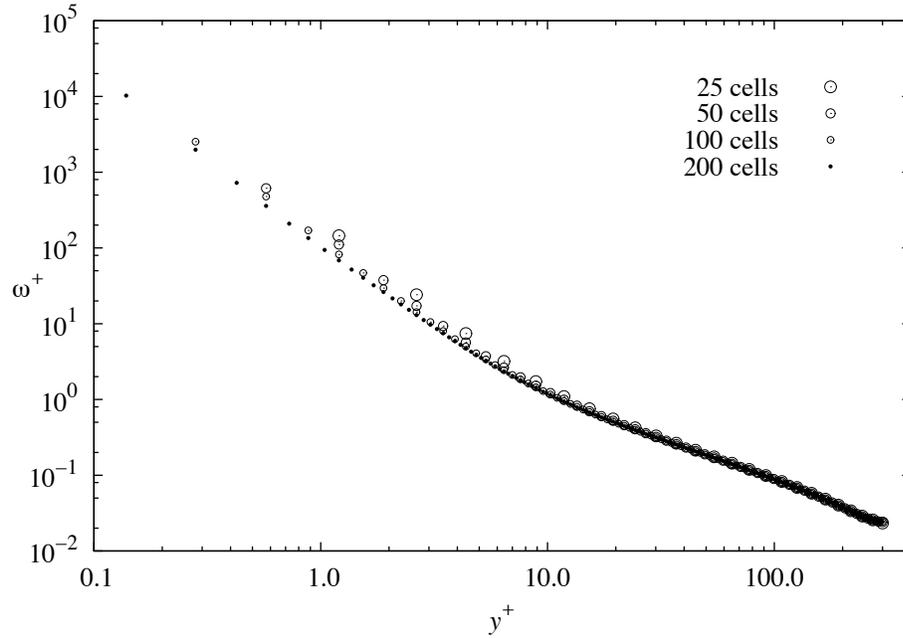


Fig. B.12 Nondimensional dissipation frequency results from the Wilcox 1998 model.

The Wilcox 1998 k - ω model is a special case of Eq. (B.76) where

$$\begin{aligned}
 R_t &\equiv \frac{k}{\nu\omega}, \quad \psi_k \equiv \frac{\nabla k \cdot \nabla \omega}{\omega^3}, \quad f_\mu = \frac{0.024 + R_t/6}{1 + R_t/6} \\
 f_k &= \frac{4/15 + (R_t/8)^4}{1 + (R_t/8)^4} \begin{cases} 1, & \psi_k \leq 0 \\ \frac{1 + 680\psi_k^2}{1 + 400\psi_k^2}, & \psi_k > 0 \end{cases}, \quad f_1 = \frac{1/9 + R_t/2.95}{f_\mu(1 + R_t/2.95)}, \quad f_2 = 1 \\
 C_\mu &= 0.09, \quad C_{\omega 1} = 0.52, \quad C_{\omega 2} = 0.072, \quad \sigma_k = 2.0, \quad \sigma_\omega = 2.0
 \end{aligned} \tag{B.100}$$

The shear-flow correction term in the damping equations can be written in cylindrical coordinates as

$$\psi_k \equiv \frac{\nabla k \cdot \nabla \omega}{\omega^3} = \frac{1}{\omega^3} \left(\frac{\partial k}{\partial r} \frac{\partial \omega}{\partial r} + \frac{1}{r^2} \frac{\partial k}{\partial \theta} \frac{\partial \omega}{\partial \theta} + \frac{\partial k}{\partial z} \frac{\partial \omega}{\partial z} \right) \tag{B.101}$$

For fully developed pipe flow, derivatives with respect to θ and z are zero. Applying these simplifications to Eq. (B.101) and applying the nondimensional parameters gives

$$\psi_k \equiv \frac{\nabla k \cdot \nabla \omega}{\omega^3} = \frac{1}{\omega^3} \frac{\partial k}{\partial r} \frac{\partial \omega}{\partial r} = \frac{1}{\omega^+{}^3 R_r^2} \frac{\partial k^+}{\partial \hat{r}} \frac{\partial \omega^+}{\partial \hat{r}} \tag{B.102}$$

This model is a special case of Eq. (D.1) where

$$\begin{aligned}
h^+ &= \omega^+ \\
f_v &= f_\mu k^+ / \omega^+ \\
S_k &= v^+ \left(\frac{du^+}{d\hat{r}} \right)^2 - C_\mu f_k R_\tau^2 k^+ \omega^+ \\
S_h &= C_{\omega 1} f_1 f_\mu \left(\frac{du^+}{d\hat{r}} \right)^2 - C_{\omega 2} f_2 R_\tau^2 \omega^{+2} \\
R_t &\equiv \frac{k^+}{\omega^+}, \quad \psi_k = \frac{1}{\omega^{+3} R_\tau^2} \frac{\partial k^+}{\partial \hat{r}} \frac{\partial \omega^+}{\partial \hat{r}}, \quad f_\mu = \frac{0.024 + R_t/6}{1 + R_t/6} \\
f_k &= \frac{4/15 + (R_t/8)^4}{1 + (R_t/8)^4} \left\{ \begin{array}{l} 1, \quad \psi_k \leq 0 \\ 1 + 680\psi_k^2, \quad \psi_k > 0 \end{array} \right\}, \quad f_1 = \frac{1/9 + R_t/2.95}{f_\mu(1 + R_t/2.95)}, \quad f_2 = 1 \\
C_\mu &= 0.09, \quad C_{\omega 1} = 0.52, \quad C_{\omega 2} = 0.072, \quad \sigma_k = 2.0, \quad \sigma_\omega = 2.0
\end{aligned} \tag{B.103}$$

The algorithms given in Eqs. (D.18)–(D.20) can be used to solve this model where the functions are defined as

$$\begin{aligned}
R_t &\equiv \frac{k_j^+}{\omega_j^+}, \quad \psi_k = \frac{k_j^{+'} \omega_j^{+'}}{\omega_j^{+3} R_\tau^2} \\
f_v &= \frac{k_j^+}{\omega_j^+} \frac{0.024 + R_t/6}{1 + R_t/6} \\
S_k &= v_j^+ u_j^{+'2} - 0.09 R_\tau^2 k_j^+ \omega_j^+ \frac{4/15 + (R_t/8)^4}{1 + (R_t/8)^4} \left\{ \begin{array}{l} 1, \quad \psi_k \leq 0 \\ 1 + 680\psi_k^2, \quad \psi_k > 0 \end{array} \right\} \\
S_h &= 0.52 \frac{1/9 + R_t/2.95}{1 + R_t/2.95} u_j^{+'2} - 0.072 R_\tau^2 \omega_j^{+2} \\
f_{h0} &= \frac{6}{0.072 y_j^{+2}}
\end{aligned} \tag{B.104}$$

and $m_h = 7$ can be used because ω^+ is singular at the wall.

APPENDIX C

COMMON CASE STUDIES

I. Boundary Layer Flow over an Infinite Flat Plate**A. Case Description**

Boundary layer flow is often one of the first scenarios considered for evaluating a turbulence model. The test case is rather simple consisting of an infinite flat plate placed in-line with the velocity vector of a uniform flow. As the flow advances along the plate, the momentum deficit from the skin drag on the plate causes a boundary layer to develop. The profile of this boundary layer has been studied in great detail and much experimental data exists for this case. Experimental results show that the boundary layer profile is self-similar and a function of the skin friction along the plate.

Because of the behavior of the boundary layer profile, certain approximations can be applied to the RANS equations and the resulting simplified equations can be written in a similarity form. The nature of the boundary layer equations allows a solution to be obtained by directly integrating the equations in space. This solution method allows for the effect of turbulence model closure coefficients to be quickly assessed because any one solution requires only a matter of seconds to compute with modern computers. However, this method suffers from the fact that approximations about the flowfield must be assumed in order to develop the boundary layer equations.

The boundary layer flow case in a two-dimensional CFD solver can be constructed on a rectangular domain as shown in Fig. C.1. On the west side of the domain a velocity inlet boundary condition is specified. At this boundary condition, all freestream properties of the flow must be specified. In this case, uniform properties of the flow are specified across the inlet boundary. Along the south side of the domain, a no-slip wall boundary condition is specified. Along the north and east sides of the domain, pressure outlet boundary conditions are specified. Once the solution has reached a converged state, there should be no flow entering the domain from the north and east boundaries. However, during the solution process, freestream

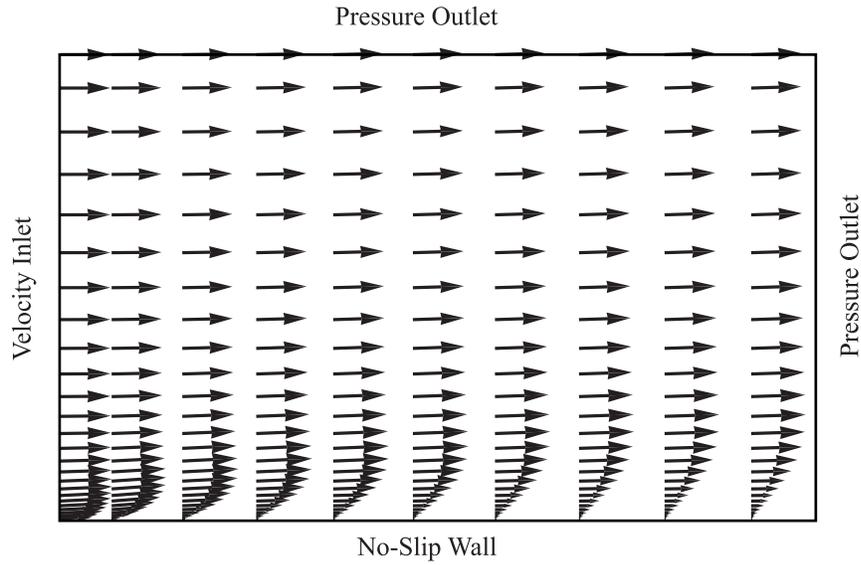


Fig. C.1 Boundary layer case description.

conditions must be specified along these boundaries in the case that during any given iteration, backflow develops along one of these boundaries. Figure C.1 shows the setup for this case.

Two of the most significant parameters of interest in flat-plate boundary layer flow are the prediction of the skin friction along the plate, and the prediction of the nondimensional velocity profile. According to convention, R_x is defined as the Reynolds number at any x -position along the plate where the x -position is measured from the leading edge of the plate along the direction of flow. This Reynolds number can be expressed as

$$R_x \equiv \bar{V}_\infty x / \nu \quad (\text{C.1})$$

Here we define R_L as the Reynolds number based on the length of the plate, L , which can be expressed as

$$R_L \equiv \bar{V}_\infty L / \nu \quad (\text{C.2})$$

The friction coefficient is defined as

$$C_f \equiv 2\tau_w / (\rho \bar{V}_\infty^2) \quad (\text{C.3})$$

Plots of the nondimensional velocity profile, u^+ , as a function of y^+ where $u^+ \equiv V_x / u_\tau$ and $y^+ \equiv y u_\tau / \nu$ in comparison with experimental data are often included. There is a wealth of experimental data for boundary

layer flows. For example, results of boundary layer flow can be compared to experimental data by Klebanoff [53], Bradshaw [54], Marusic and Hutchins [55], and Guala et al. [56].

B. Laminar Flow

The laminar case for boundary layers was studied in detail by Blasius [104]. Applying the boundary layer equations, Blasius developed an expression for the friction coefficient in terms of the Reynolds number along the plate

$$C_f \approx \frac{0.664}{\sqrt{R_x}} \quad (\text{C.4})$$

A laminar case was run in both ICESS and Fluent using the following dimensional properties:

$$\begin{aligned} \text{domain width} &= 1.0 \text{ m}, & \text{domain height} &= 0.2 \text{ m}, \\ V_\infty &= 100.0 \text{ m/s}, & \rho &= 0.1 \text{ kg/m}^3, & \mu &= 0.001 \text{ N}\cdot\text{s/m}^2, \\ L &= 1.0 \text{ m}, & R_L &= 1e4 \end{aligned} \quad (\text{C.5})$$

Figures C.2 and C.3 show grid resolution results from ICESS and Fluent for the case and include grid resolutions of 25×25 , 50×50 , and 100×100 . The figures show the friction coefficient as a function of Reynolds number along the plate and include the Blasius solution. Figure C.4 compares the grid-resolved results from ICESS to those from Fluent, both on grids of 100×100 cells.

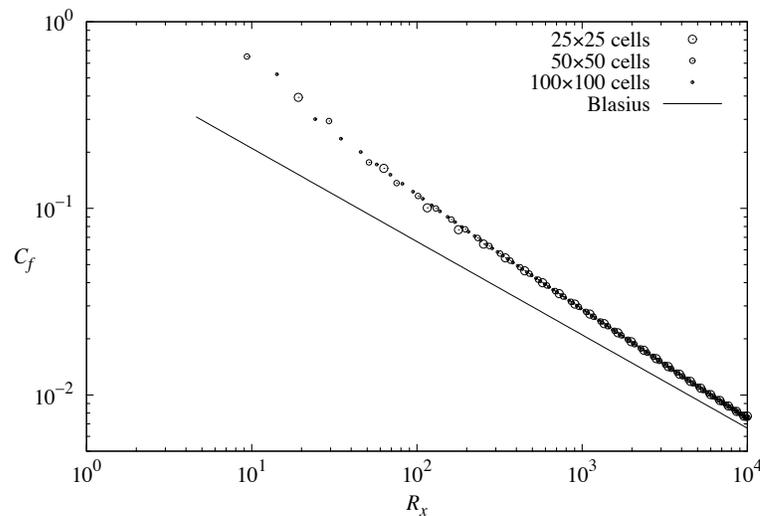


Fig. C.2 Grid resolution results from ICESS for the laminar boundary layer.

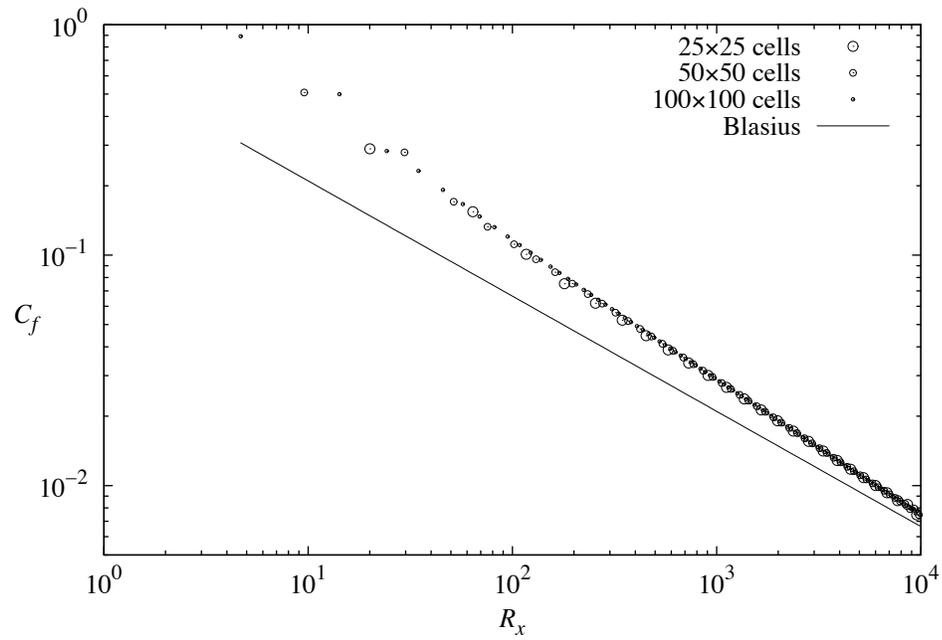


Fig. C.3 Grid resolution results from Fluent for the laminar boundary layer.

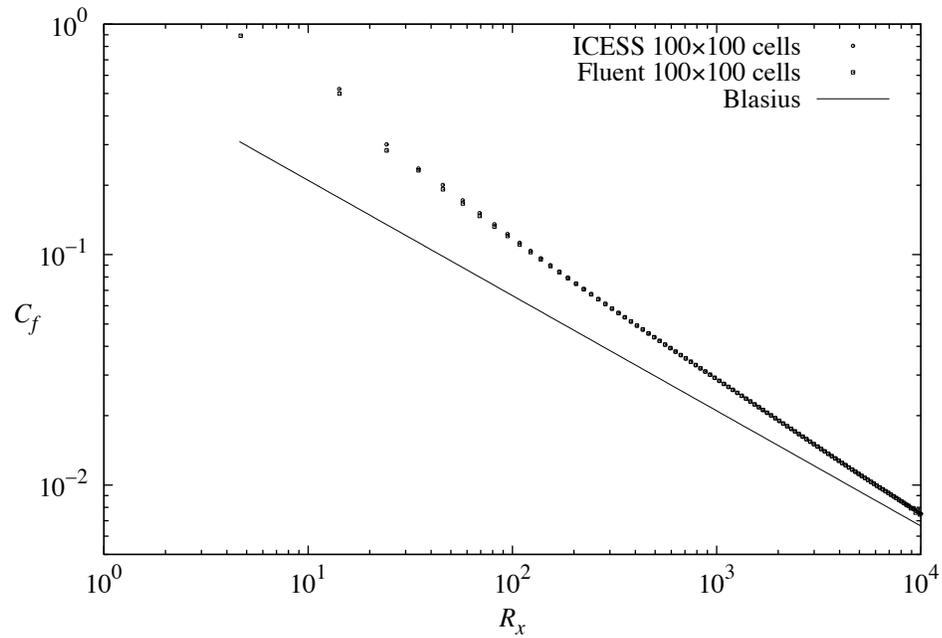


Fig. C.4 Grid-resolved results from ICESS and Fluent for the laminar boundary layer.

C. Turbulent Flow

The turbulent boundary layer case has been studied in detail including both analytical studies and experimental studies. Several approximations for the friction coefficient exist. For example, White [105] develops the relation

$$C_f \approx \frac{0.455}{\ln^2(0.06R_x)} \quad (\text{C.6})$$

Kays and Crawford [106] develop the relation

$$C_f \approx \frac{0.0574}{R_x^{1/5}} \quad (\text{C.7})$$

Schultz-Grunow [107] obtains the relation

$$C_f \approx \frac{0.37}{[\log(R_x)]^{2.584}} \quad (\text{C.8})$$

It is well understood that the nondimensional velocity profile of a turbulent boundary layer is a function of the nondimensional wall coordinate. In the laminar sublayer, this profile follows the expression

$$u^+ = y^+, \quad y^+ < \sim 5 \quad (\text{C.9})$$

In the log layer, the velocity profile follows what is known as the law of the wall

$$u^+ = \frac{1}{\kappa} \ln(y^+) + C, \quad 30 < y^+ < \sim 500 \quad (\text{C.10})$$

Based on modern experimental data, it is generally accepted that the best values for the constants in the law-of-the-wall equation are $\kappa = 0.41$ and $C = 5.0$.

II. Fully Developed Channel Flow

A. Case Description

Fully developed flow in a channel is a common case for evaluating a turbulence model. This case is a two-dimensional flow which is identical to the three-dimensional flow between two infinite flat plates. As flow enters a channel, a boundary layer develops on each channel wall. As the flow moves downstream, the two boundary layers eventually meet, and the flow reaches the fully developed state.

This flow case in two dimensions can be constructed on a rectangular domain as shown in Fig. C.5. Along the south side of the domain, a no-slip wall boundary condition is applied. Because this case is symmetrical along the centerline of the channel, only half of the channel is modeled, and a symmetry boundary condition is applied along the north side of the domain. The east and west sides of the domain are then specified as periodic boundary conditions. This boundary condition forces the properties of the flow exiting one side of the domain to be equal to those entering the other side of the domain. The use of periodic boundary conditions requires either a pressure drop across the domain to be specified, or a mass flux across the periodic boundary to be specified. Figure C.5 shows the setup for this case in Cartesian coordinates.

For fully developed channel flow, gradients in the flow properties with respect to the flow direction disappear, and the profiles of flow properties become dependent only on the coordinate normal to the wall. Therefore, the governing equations can be simplified to a one-dimensional problem and the solution can be obtained numerically very quickly on modern computers. The formulation can be discretized into a banded system of equations and solved quite rapidly where the full two-dimensional problem may take a significantly longer amount of time. Therefore, this one-dimensional test case is ideal for the optimization process of closure coefficients.

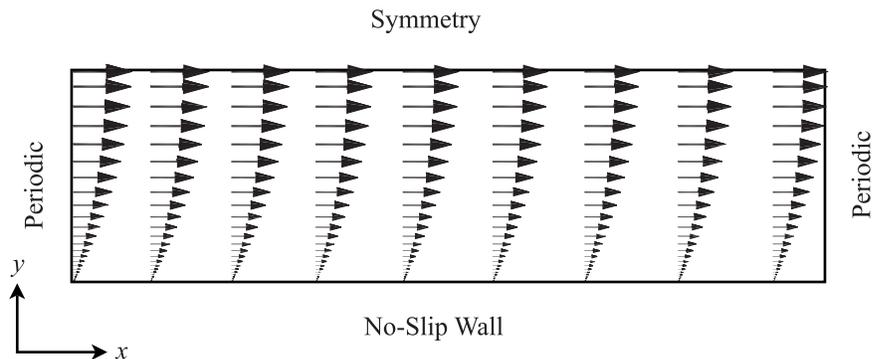


Fig. C.5 Fully developed channel flow case description.

B. The Continuity and RANS Equations

The steady-state, incompressible continuity and Boussinesq-RANS equations can be written in vector format as

$$\nabla \cdot \bar{\mathbf{V}} = 0 \quad (\text{C.11})$$

$$(\bar{\mathbf{V}} \cdot \nabla) \bar{\mathbf{V}} = -\nabla \hat{p} / \rho + \nabla \cdot [2(\nu + \nu_t) \bar{\mathbf{S}}(\bar{\mathbf{V}})] \quad (\text{C.12})$$

These can be written for two-dimensional flow in Cartesian coordinates as

$$\frac{\partial \bar{V}_x}{\partial x} + \frac{\partial \bar{V}_y}{\partial y} = 0 \quad (\text{C.13})$$

$$\frac{\partial(\bar{V}_x \bar{V}_x)}{\partial x} + \frac{\partial(\bar{V}_y \bar{V}_x)}{\partial y} = -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial x} + \frac{\partial}{\partial x} \left[2(\nu + \nu_t) \frac{\partial \bar{V}_x}{\partial x} \right] + \frac{\partial}{\partial y} \left[(\nu + \nu_t) \left(\frac{\partial \bar{V}_x}{\partial y} + \frac{\partial \bar{V}_y}{\partial x} \right) \right] \quad (\text{C.14})$$

$$\frac{\partial(\bar{V}_x \bar{V}_y)}{\partial x} + \frac{\partial(\bar{V}_y \bar{V}_y)}{\partial y} = -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial y} + \frac{\partial}{\partial y} \left[2(\nu + \nu_t) \frac{\partial \bar{V}_y}{\partial y} \right] + \frac{\partial}{\partial x} \left[(\nu + \nu_t) \left(\frac{\partial \bar{V}_x}{\partial y} + \frac{\partial \bar{V}_y}{\partial x} \right) \right] \quad (\text{C.15})$$

where the y -coordinate is the normal coordinate measured outward from the wall of the channel and the x -coordinate is the coordinate along the channel. For fully developed flow in a pipe, the gradients of transport properties with respect to x are zero, and these equations can be simplified to

$$\frac{\partial \bar{V}_y}{\partial y} = 0 \quad (\text{C.16})$$

$$\frac{\partial(\bar{V}_y \bar{V}_x)}{\partial y} = -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial x} + \frac{\partial}{\partial y} \left[(\nu + \nu_t) \left(\frac{\partial \bar{V}_x}{\partial y} \right) \right] \quad (\text{C.17})$$

$$\frac{\partial(\bar{V}_y \bar{V}_y)}{\partial y} = -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial y} + \frac{\partial}{\partial y} \left[2(\nu + \nu_t) \frac{\partial \bar{V}_y}{\partial y} \right] \quad (\text{C.18})$$

The no-slip boundary conditions at the wall of the pipe, $y = 0$, are

$$\begin{aligned} \bar{V}_x(0) &= 0 \\ \bar{V}_y(0) &= 0 \end{aligned} \quad (\text{C.19})$$

Applying the \bar{V}_y boundary condition to the integral of Eq. (C.16) gives

$$\bar{V}_y = 0 \quad (\text{C.20})$$

Applying this result to Eq. (C.18) gives

$$\frac{\partial \hat{p}}{\partial y} = 0 \quad (\text{C.21})$$

This shows that $\hat{p} = \hat{p}(x)$. Using Eqs. (C.20) and (C.21) in Eq. (C.17) gives

$$\frac{d}{dy} \left[(v + v_t) \left(\frac{d\bar{V}_x}{dy} \right) \right] = \frac{1}{\rho} \frac{d\hat{p}}{dx} \quad (\text{C.22})$$

Integrating Eq. (C.22) from the wall to some arbitrary distance, y , gives

$$\int_{y=0}^y \frac{d}{dy} \left[(v + v_t) \frac{d\bar{V}_x}{dy} \right] dy = \int_{y=0}^y \frac{1}{\rho} \frac{d\hat{p}}{dx} dx \quad (\text{C.23})$$

$$\left[(v + v_t) \frac{d\bar{V}_x}{dy} \right]_{y=y} - \left[(v + v_t) \frac{d\bar{V}_x}{dy} \right]_{y=0} = \frac{1}{\rho} \frac{d\hat{p}}{dx} y$$

Because the turbulent eddy viscosity goes to zero at a no-slip wall, the second term on the left-hand side of Eq. (C.23) evaluated at the wall can be written in terms of the friction velocity or the wall shear stress

$$\left[(v + v_t) \frac{d\bar{V}_x}{dy} \right]_{y=0} = \frac{\tau_w}{\rho} = u_\tau^2 \quad (\text{C.24})$$

Using this in Eq. (C.23) gives the relation

$$(v + v_t) \frac{d\bar{V}_x}{dy} = \frac{1}{\rho} \frac{d\hat{p}}{dx} y + u_\tau^2 \quad (\text{C.25})$$

The velocity boundary condition at the center of the channel, $y = L$, is

$$\left. \frac{d\bar{V}_x}{dy} \right|_{y=L} = 0 \quad (\text{C.26})$$

Applying this to Eq. (C.25) gives

$$\frac{1}{\rho} \frac{d\hat{p}}{dx} = -\frac{u_\tau^2}{L} \quad (\text{C.27})$$

Finally, applying this relation to Eq. (C.25) gives

$$\frac{d\bar{V}_x}{dy} = \frac{u_\tau^2(1 - y/L)}{(v + v_t)} \quad (\text{C.28})$$

The friction velocity is commonly used to nondimensionalize the parameters of a wall-bounded flow.

Defining the nondimensional parameters

$$u^+ \equiv \frac{\bar{V}_x}{u_\tau}, \quad y^+ \equiv \frac{yu_\tau}{\nu}, \quad R_\tau \equiv \frac{Lu_\tau}{\nu}, \quad \nu^+ \equiv \frac{\nu}{\nu}, \quad p^+ \equiv \frac{\nu}{\rho u_\tau^3} \frac{d\hat{p}}{dx} \quad (\text{C.29})$$

Using these definitions, Eqs. (C.27) and (C.28) can be rewritten as

$$p^+ = -1/R_\tau \quad (\text{C.30})$$

$$\frac{du^+}{dy^+} = \frac{1-y^+/R_\tau}{1+\nu^+} \quad (\text{C.31})$$

The most significant parameters of interest for the channel flow case are the ability of a model to predict the correct relation between flow Reynolds number and the friction coefficient, and the ability of the model to predict the correct nondimensional velocity profile. The channel Reynolds number is defined as

$$R_e \equiv \bar{V}_{\text{bulk}} D_h / \nu \quad (\text{C.32})$$

where

$$\bar{V}_{\text{bulk}} \equiv \frac{1}{L} \int_{y=0}^L \bar{V}_x dy \quad (\text{C.33})$$

is the bulk velocity, and

$$D_h = 4L \quad (\text{C.34})$$

is the hydraulic diameter based on the channel half-width, L . The Fanning friction factor is defined as

$$C_f \equiv \frac{2\tau_w}{\rho \bar{V}_{\text{bulk}}^2} = \frac{2u_\tau^2}{\bar{V}_{\text{bulk}}^2} \quad (\text{C.35})$$

The Darcy friction factor is defined as

$$f_D \equiv \frac{D_h}{\frac{1}{2} \rho \bar{V}_{\text{bulk}}^2} \left(-\frac{d\hat{p}}{dx} \right) \quad (\text{C.36})$$

Using Eqs. (C.27) and (C.34) in this definition gives an alternate form for channel flow

$$f_D = \frac{8u_\tau^2}{\bar{V}_{\text{bulk}}^2} = 4C_f \quad (\text{C.37})$$

Using the nondimensional parameters given in Eq. (C.29) as well as the definition

$$u_{\text{bulk}}^+ \equiv \frac{\bar{V}_{\text{bulk}}}{u_\tau} = \frac{1}{R_\tau} \int_{y^+=0}^{R_\tau} u^+ dy^+ \quad (\text{C.38})$$

in Eqs. (C.32), (C.35), and (C.37) gives the expressions

$$R_e = 4u_{\text{bulk}}^+ R_\tau \quad (\text{C.39})$$

$$C_f = \frac{2}{u_{\text{bulk}}^{+2}} \quad (\text{C.40})$$

$$f_D = \frac{8}{u_{\text{bulk}}^{+2}} = 4C_f \quad (\text{C.41})$$

C. Laminar Flow

For laminar flow, the turbulent eddy viscosity is zero throughout the flow. Integrating Eq. (C.28) for laminar flow and applying Eq. (C.27) gives the laminar fully developed channel flow solution

$$\begin{aligned} \bar{V}_x &= \frac{u_\tau^2}{\nu} \left(y - \frac{y^2}{2L} \right) = \frac{1}{\mu} \frac{d\hat{p}}{dx} \left(\frac{y^2}{2} - Ly \right) \\ u^+ &= y^+ - \frac{y^{+2}}{2R_\tau} \end{aligned} \quad (\text{C.42})$$

Evaluating Eq. (C.42) at $y = L$ gives the centerline velocity

$$\begin{aligned} \bar{V}_c &= \frac{u_\tau^2 L}{2\nu} = -\frac{L^2}{2\mu} \frac{d\hat{p}}{dx} \\ u_c^+ &= \frac{R_\tau}{2} \end{aligned} \quad (\text{C.43})$$

Dividing Eq. (C.42) by Eq. (C.43) gives a normalized expression for the velocity profile

$$\begin{aligned} \frac{\bar{V}_x}{\bar{V}_c} &= \frac{y}{L} \left(2 - \frac{y}{L} \right) \\ \frac{u^+}{u_c^+} &= \frac{y^+}{R_\tau} \left(2 - \frac{y^+}{R_\tau} \right) \end{aligned} \quad (\text{C.44})$$

The bulk velocity can be evaluated by integrating Eq. (C.42)

$$\bar{V}_{\text{bulk}} = \frac{1}{L} \int_{y=0}^L \frac{u_{\tau}^2}{\nu} \left(y - \frac{y^2}{2L} \right) dy = \frac{u_{\tau}^2 L}{3\nu} = -\frac{L^3}{3\mu} \frac{d\hat{p}}{dx}$$

$$u_{\text{bulk}}^+ = \frac{1}{R_{\tau}} \int_{y^+=0}^{R_{\tau}} \left(y^+ - \frac{y^{+2}}{2R_{\tau}} \right) dy^+ = \frac{R_{\tau}}{3}$$
(C.45)

Using Eq. (C.45) along with Eq. (C.32) in Eq. (C.35) gives

$$C_f = \frac{24}{R_e}$$
(C.46)

The fully developed channel flow case was run in ICESST using the following properties

$$\mu = 0.001 \text{ kg}/(\text{m} \cdot \text{s}), \quad \rho = 0.1 \text{ kg}/\text{m}^3, \quad L = 0.5 \text{ m}, \quad \frac{d\hat{p}}{dx} = -7.2 \text{ Pa}/\text{m}$$

$$\tau_w = 3.6 \text{ Pa}/\text{m}, \quad u_{\tau} = 6.0 \text{ m}/\text{s}, \quad R_{\tau} = 300.0$$
(C.47)

Figure C.6 shows the nondimensional velocity profile results for a grid with 50 cells using logarithmic clustering near the wall.

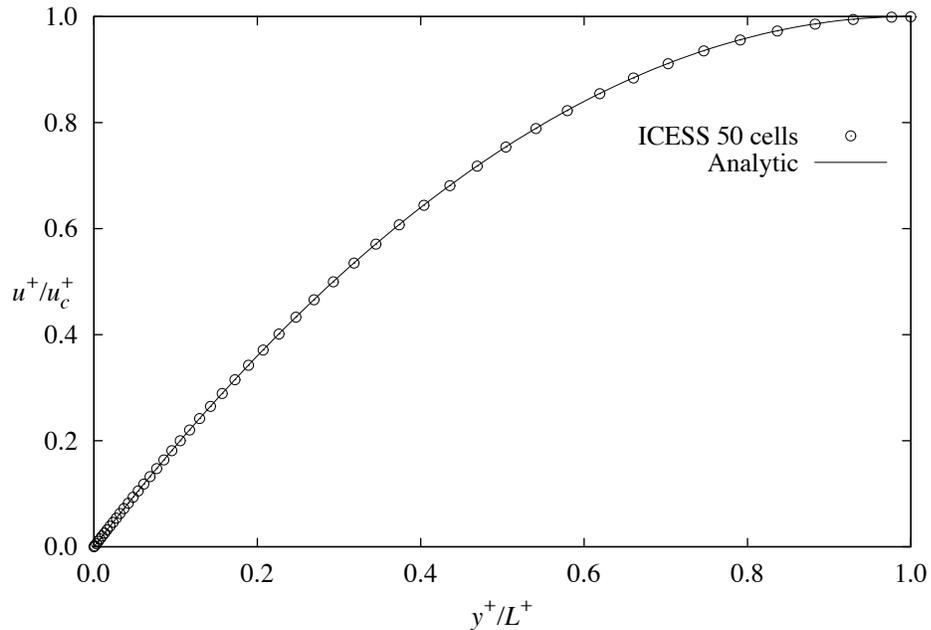


Fig. C.6 Normalized velocity profile for fully developed channel flow.

D. Turbulent Flow

Two-equation RANS-based turbulence models can be coupled with Eq. (C.31). A general form for the complete system of equations including boundary conditions can often be written in the form

$$\begin{aligned}
\frac{du^+}{dy^+} &= \frac{1-y^+/R_\tau}{1+v^+} \\
v^+ &= f_v \\
\frac{d}{dy^+} \left[-(1+v^+/\sigma_k) \frac{dk^+}{dy^+} \right] &= S_k \\
\frac{d}{dy^+} \left[-(1+v^+/\sigma_h) \frac{dh^+}{dy^+} \right] &= S_h \\
u^+(0) = 0, \quad k^+(0) = 0, \quad \frac{dk^+}{dy^+}(0) = 0, \quad \frac{dk^+}{dy^+}(R_\tau) = 0, \quad \frac{dh^+}{dy^+}(R_\tau) = 0
\end{aligned} \tag{C.48}$$

where h^+ represents the second turbulence variable and f_v , S_k , and S_h are model-dependent functions. For sample results, see Appendix B.

III. Fully Developed Pipe Flow

A. Case Description

The fully developed pipe flow case is very closely related to the fully developed channel flow case. As flow enters a pipe, a boundary layer develops along the pipe wall. As the flow moves downstream, the boundary layer eventually fills the entire pipe, and the flow reaches the fully developed state. In this state, there are no gradients in the azimuthal direction of the pipe, and the case can be simplified to a two-dimensional flow in cylindrical coordinates.

This flow case in two dimensions can be constructed on a rectangular domain as shown in Fig. C.7. Along the north side of the domain, a no-slip wall boundary condition is applied. Because this case is symmetrical along the centerline of the pipe, only half of the pipe is modeled, and a symmetry boundary condition is applied along the south side of the domain. This side of the domain is also the axis of rotation of the pipe. The east and west sides of the domain are then specified as periodic boundary conditions where either a pressure drop across the domain or a mass flux across the periodic boundary must be specified. Figure C.7 shows the setup for this case in cylindrical coordinates.

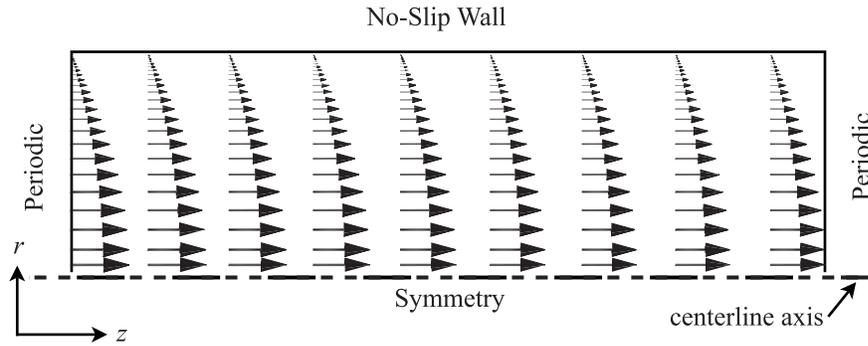


Fig. C.7 Fully developed pipe flow case description.

For fully developed pipe flow, gradients in the flow properties with respect to the flow direction disappear, and the profiles of flow properties become dependent only on the coordinate normal to the axis of symmetry. Therefore, the governing equations can be simplified to a one-dimensional problem and the solution can be obtained numerically very quickly on modern computers. Because solutions to this case can be quickly obtained, this test case is ideal for evaluating closure coefficients.

B. The Continuity and RANS Equations

The governing steady-state, incompressible continuity and Boussinesq-RANS equations can be written in vector form as

$$\nabla \cdot \bar{\mathbf{V}} = 0 \quad (\text{C.49})$$

$$(\bar{\mathbf{V}} \cdot \nabla) \bar{\mathbf{V}} = -\nabla \hat{p} / \rho + \nabla \cdot [2(\nu + \nu_t) \bar{\mathbf{S}}(\bar{\mathbf{V}})] \quad (\text{C.50})$$

These can be written for flow in a pipe in cylindrical coordinates including the continuity equation

$$\frac{1}{r} \frac{\partial(r\bar{V}_r)}{\partial r} + \frac{1}{r} \frac{\partial\bar{V}_\theta}{\partial \theta} + \frac{\partial\bar{V}_z}{\partial z} = 0 \quad (\text{C.51})$$

and the three components of the Boussinesq-RANS equations

$$\begin{aligned}
\bar{V}_r \frac{\partial \bar{V}_r}{\partial r} + \frac{\bar{V}_\theta}{r} \frac{\partial \bar{V}_r}{\partial \theta} + \bar{V}_z \frac{\partial \bar{V}_r}{\partial z} - \frac{\bar{V}_\theta^2}{r} = & -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial r} + \frac{1}{r} \frac{\partial}{\partial r} \left[2(v + v_t) r \frac{\partial \bar{V}_r}{\partial r} \right] \\
& + \frac{1}{r^2} \frac{\partial}{\partial \theta} \left[(v + v_t) \left(r^2 \frac{\partial (\bar{V}_\theta/r)}{\partial r} + \frac{\partial \bar{V}_r}{\partial \theta} \right) \right] \\
& + \frac{\partial}{\partial z} \left[(v + v_t) \left(\frac{\partial \bar{V}_z}{\partial r} + \frac{\partial \bar{V}_r}{\partial z} \right) \right] - \frac{2(v + v_t)}{r^2} \left(\frac{\partial \bar{V}_\theta}{\partial \theta} + \bar{V}_r \right)
\end{aligned} \tag{C.52}$$

$$\begin{aligned}
\bar{V}_r \frac{\partial \bar{V}_\theta}{\partial r} + \frac{\bar{V}_\theta}{r} \frac{\partial \bar{V}_\theta}{\partial \theta} + \bar{V}_z \frac{\partial \bar{V}_\theta}{\partial z} + \frac{\bar{V}_r \bar{V}_\theta}{r} = & -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial \theta} + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t) r \left(\frac{1}{r} \frac{\partial \bar{V}_r}{\partial \theta} + r \frac{\partial (\bar{V}_\theta/r)}{\partial r} \right) \right] \\
& + \frac{1}{r} \frac{\partial}{\partial \theta} \left[2(v + v_t) \left(\frac{1}{r} \frac{\partial \bar{V}_\theta}{\partial \theta} + \frac{\bar{V}_r}{r} \right) \right] \\
& + \frac{\partial}{\partial z} \left[(v + v_t) \left(\frac{1}{r} \frac{\partial \bar{V}_z}{\partial \theta} + \frac{\partial \bar{V}_\theta}{\partial z} \right) \right] + \frac{(v + v_t)}{r^2} \left(\frac{\partial \bar{V}_r}{\partial \theta} + r^2 \frac{\partial (\bar{V}_\theta/r)}{\partial r} \right)
\end{aligned} \tag{C.53}$$

$$\begin{aligned}
\bar{V}_r \frac{\partial \bar{V}_z}{\partial r} + \frac{\bar{V}_\theta}{r} \frac{\partial \bar{V}_z}{\partial \theta} + \bar{V}_z \frac{\partial \bar{V}_z}{\partial z} = & -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial z} + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t) r \left(\frac{\partial \bar{V}_r}{\partial z} + \frac{\partial \bar{V}_z}{\partial r} \right) \right] \\
& + \frac{1}{r} \frac{\partial}{\partial \theta} \left[(v + v_t) \left(\frac{\partial \bar{V}_\theta}{\partial z} + \frac{1}{r} \frac{\partial \bar{V}_z}{\partial \theta} \right) \right] + \frac{\partial}{\partial z} \left[2(v + v_t) \left(\frac{\partial \bar{V}_z}{\partial z} \right) \right]
\end{aligned} \tag{C.54}$$

For fully developed flow in a pipe, the gradients of transport properties with respect to z are zero. Additionally, all gradients with respect to θ are zero and $\bar{V}_\theta = 0$. Therefore, Eqs. (C.51)–(C.54) can be simplified to

$$\frac{d(r\bar{V}_r)}{dr} = 0 \tag{C.55}$$

$$\bar{V}_r \frac{d\bar{V}_r}{dr} = -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial r} + \frac{1}{r} \frac{d}{dr} \left[2(v + v_t) r \frac{d\bar{V}_r}{dr} \right] - \frac{2(v + v_t)}{r^2} \bar{V}_r \tag{C.56}$$

$$\bar{V}_r \frac{d\bar{V}_z}{dr} = -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial z} + \frac{1}{r} \frac{d}{dr} \left[(v + v_t) r \frac{d\bar{V}_z}{dr} \right] \tag{C.57}$$

The no-slip boundary conditions at the wall of the pipe, $r=R$, are

$$\begin{aligned}
\bar{V}_r(R) &= 0 \\
\bar{V}_z(R) &= 0
\end{aligned} \tag{C.58}$$

Applying the \bar{V}_r boundary condition to the integral of Eq. (C.55) gives $\bar{V}_r = 0$ and the governing equations can be written as

$$\bar{V}_r = 0 \quad (\text{C.59})$$

$$\frac{\partial \hat{p}}{\partial r} = 0 \quad (\text{C.60})$$

$$\frac{1}{r} \frac{d}{dr} \left[(v + v_t) r \frac{d\bar{V}_z}{dr} \right] = \frac{1}{\rho} \frac{d\hat{p}}{dz} \quad (\text{C.61})$$

The results of Eq. (C.60) show that $\hat{p} = \hat{p}(z)$ which has been used in Eq. (C.61). The coordinate r is the normal coordinate measured outward from the center of the pipe. The velocity boundary condition at the center of the pipe, $r=0$, is

$$\frac{d\bar{V}_z}{dr}(0) = 0 \quad (\text{C.62})$$

Integrating Eq. (C.61) from the centerline to some arbitrary distance, r , gives

$$\int_{r=0}^r \frac{d}{dr} \left[(v + v_t) r \frac{d\bar{V}_z}{dr} \right] = \int_{r=0}^r \frac{r}{\rho} \frac{d\hat{p}}{dz} \quad (\text{C.63})$$

$$(v + v_t) r \frac{d\bar{V}_z}{dr} = \frac{1}{\rho} \frac{d\hat{p}}{dz} \left(\frac{r^2}{2} + C \right)$$

where C is an arbitrary constant. Applying the boundary condition given in Eq. (C.62) gives $C = 0$ and Eq. (C.63) can be written

$$(v + v_t) \frac{d\bar{V}_z}{dr} = \frac{1}{2\rho} \frac{d\hat{p}}{dz} r \quad (\text{C.64})$$

The left-hand side of Eq. (C.64) evaluated at the wall is related to the shear stress at the wall which can also be written in terms of the friction velocity

$$(v + v_t) \frac{d\bar{V}_z}{dr}(R) = -\frac{\tau_w}{\rho} = -u_\tau^2 \quad (\text{C.65})$$

Using this in Eq. (C.64) evaluated at $r=R$ gives the relationship

$$\frac{1}{2\rho} \frac{d\hat{p}}{dz} = -\frac{u_\tau^2}{R} \quad (\text{C.66})$$

Using this in Eq. (C.64) gives the formulation and the remaining boundary condition

$$\frac{d\bar{V}_z}{dr} = -\frac{u_\tau^2}{(v + v_t)} \frac{r}{R}, \quad \bar{V}_z(R) = 0 \quad (\text{C.67})$$

This can be written in nondimensional form using the following parameters

$$\begin{aligned}\hat{r} &\equiv \frac{r}{R}, \quad R_\tau \equiv \frac{u_\tau R}{\nu}, \quad u^+ \equiv \frac{\bar{V}_z}{u_\tau}, \quad v^+ \equiv \frac{V_t}{\nu}, \\ y^+ &\equiv \frac{u_\tau(R-r)}{\nu} = \frac{u_\tau R(1-r/R)}{\nu} = R_\tau(1-\hat{r}), \\ p^+ &\equiv \frac{\hat{d}\hat{p}}{dz} \frac{\nu}{\rho u_\tau^3} = -\frac{2\nu}{u_\tau^3} \frac{u_\tau^2}{R} = -\frac{2\nu}{u_\tau R} = -\frac{2}{R_\tau}\end{aligned}\quad (\text{C.68})$$

Using these nondimensional parameters, Eq. (C.67) can be written as

$$\frac{du^+}{d\hat{r}} = -\frac{R_\tau \hat{r}}{(1+v^+)}, \quad u^+(1) = 0 \quad (\text{C.69})$$

The most significant parameters of interest for the pipe flow case are the ability of a model to predict the correct relation between flow Reynolds number and the friction coefficient, and the ability of the model to predict the correct nondimensional velocity distribution. The pipe Reynolds number is defined as

$$R_e \equiv \bar{V}_{\text{bulk}} D_h / \nu \quad (\text{C.70})$$

where

$$\bar{V}_{\text{bulk}} \equiv \frac{2}{R^2} \int_{r=0}^R \bar{V}_z r dr \quad (\text{C.71})$$

is the bulk velocity, and

$$D_h = 2R \quad (\text{C.72})$$

is the hydraulic diameter based on the pipe radius, R . Again, the Fanning friction factor is defined as

$$C_f \equiv \frac{2\tau_w}{\rho \bar{V}_{\text{bulk}}^2} = \frac{2u_\tau^2}{\bar{V}_{\text{bulk}}^2} \quad (\text{C.73})$$

The Darcy friction factor is defined as

$$f_D \equiv \frac{D_h}{\frac{1}{2} \rho \bar{V}_{\text{bulk}}^2} \left(-\frac{d\hat{p}}{dz} \right) \quad (\text{C.74})$$

Using Eqs. (C.66) and (C.72) in this definition gives an alternate form for pipe flow

$$f_D = \frac{8u_\tau^2}{\bar{V}_{\text{bulk}}^2} = 4C_f \quad (\text{C.75})$$

Using the nondimensional parameters given in Eq. (C.68) as well as the definition

$$u_{\text{bulk}}^+ \equiv \frac{\bar{V}_{\text{bulk}}}{u_\tau} = 2 \int_{\hat{r}=0}^1 u^+ \hat{r} d\hat{r} \quad (\text{C.76})$$

in Eqs. (C.70), (C.73), and (C.75) gives the expressions

$$R_e = 2u_{\text{bulk}}^+ R_\tau \quad (\text{C.77})$$

$$C_f = \frac{2}{u_{\text{bulk}}^{+2}} \quad (\text{C.78})$$

$$f_D = \frac{8}{u_{\text{bulk}}^{+2}} = 4C_f \quad (\text{C.79})$$

C. Laminar Flow

For laminar flow, the turbulent eddy viscosity is zero throughout the flow. Integrating Eq. (C.67) for laminar flow gives

$$\bar{V}_z = -\frac{u_\tau^2 r^2}{2\nu R} + C \quad (\text{C.80})$$

Applying the boundary condition $\bar{V}_z(R) = 0$ gives $C = u_\tau^2 R/2\nu$ and the laminar fully developed pipe flow solution can be expressed as

$$\begin{aligned} \bar{V}_z &= \frac{u_\tau^2}{2\nu R} (R^2 - r^2) = \frac{1}{4\mu} \frac{d\hat{p}}{dz} (r^2 - R^2) \\ u^+ &= \frac{R_\tau}{2} (1 - \hat{r}^2) \end{aligned} \quad (\text{C.81})$$

Evaluating Eq. (C.81) at $r = 0$ gives the centerline velocity

$$\begin{aligned} \bar{V}_c &= \frac{u_\tau^2 R}{2\nu} = -\frac{R^2}{4\mu} \frac{d\hat{p}}{dz} \\ u_c^+ &= \frac{R_\tau}{2} \end{aligned} \quad (\text{C.82})$$

Dividing Eq. (C.81) by Eq. (C.82) gives a normalized expression for the velocity profile

$$\frac{\bar{V}_z}{\bar{V}_c} = 1 - \frac{r^2}{R^2}$$

$$\frac{u^+}{u_c^+} = 1 - \hat{r}^2$$
(C.83)

The bulk velocity can be evaluated by integrating Eq. (C.81)

$$\bar{V}_{\text{bulk}} = \frac{2}{R^2} \int_{r=0}^R \frac{u_\tau^2}{2\nu R} (R^2 - r^2) r dr = \frac{R u_\tau^2}{4\nu} = -\frac{R^2}{8\mu} \frac{d\hat{p}}{dz}$$

$$u_{\text{bulk}}^+ = R_\tau \int_{\hat{r}=0}^1 (1 - \hat{r}^2) \hat{r} d\hat{r} = \frac{R_\tau}{4}$$
(C.84)

Using Eq. (C.84) along with Eq. (C.70) in Eq. (C.73) gives

$$C_f = \frac{16}{R_e}$$
(C.85)

A laminar fully developed pipe flow case was run in ICESS using the following properties

$$\mu = 0.001 \text{ kg/(m}\cdot\text{s)}, \quad \rho = 0.1 \text{ kg/m}^3, \quad R = 0.5 \text{ m}, \quad \frac{d\hat{p}}{dz} = -14.4 \text{ Pa/m}$$

$$\tau_w = 3.6 \text{ Pa/m}, \quad u_\tau = 6.0 \text{ m/s}, \quad R_\tau = 300.0$$
(C.86)

Figure C.8 shows the nondimensional velocity profile results for a grid with 50 cells using logarithmic clustering near the wall.

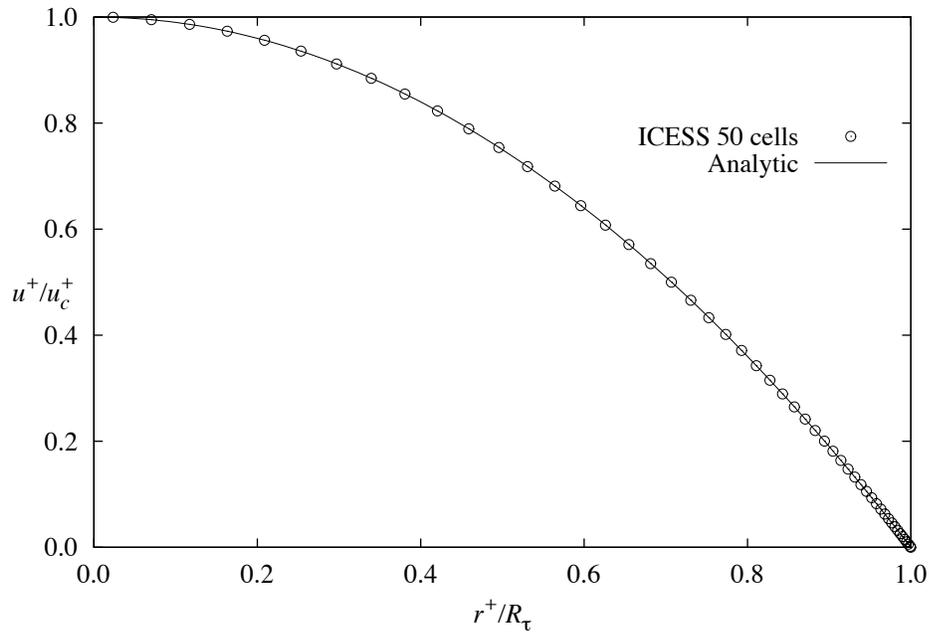


Fig. C.8 Normalized velocity profile for fully developed pipe flow.

D. Turbulent Flow

Two-equation RANS-based turbulence models can be coupled with Eq. (C.69). A general form for the complete system of equations including boundary conditions can often be written in the form

$$\begin{aligned}
 \frac{du^+}{d\hat{r}} &= -\frac{R_\tau \hat{r}}{(1+\nu^+)} \\
 \nu^+ &= f_\nu \\
 \frac{1}{\hat{r}} \frac{d}{d\hat{r}} \left[-(1+\nu^+/\sigma_k) \hat{r} \frac{dk^+}{d\hat{r}} \right] &= S_k \\
 \frac{1}{\hat{r}} \frac{d}{d\hat{r}} \left[-(1+\nu^+/\sigma_h) \hat{r} \frac{dh^+}{d\hat{r}} \right] &= S_h \\
 u^+(1) = 0, \quad k^+(1) = 0, \quad \frac{dk^+}{d\hat{r}}(1) = 0, \quad \frac{dk^+}{d\hat{r}}(0) = 0, \quad \frac{dh^+}{d\hat{r}}(0) = 0
 \end{aligned} \tag{C.87}$$

where h^+ represents the second turbulence variable and f_ν , S_k , and S_h are model-dependent functions. For sample models, see Appendix B.

IV. Plane Jet Flow

A. Case Description

The plane jet flow case is a valuable case for testing the ability of the model to predict shear flows. The case consists of a two-dimensional jet of fluid entering a quiescent fluid. As the jet of fluid advances into the quiescent fluid, the momentum of the jet is diffused outward normal to the jet axis. The jet centerline velocity decreases as the flow moves downstream, and the width of the jet grows. Eventually, the core of the jet profile becomes self-similar. The boundary layer equations are often applied to this case in order to develop a similarity solution for the jet profile. However, these equations are based on the assumption that the fluid velocity normal to the jet axis is much smaller than the fluid velocity in the direction of the jet axis. This assumption holds near the center of the jet, but is obviously not correct far from the jet centerline. In fact, in the regions far from the jet centerline, the fluid velocity normal to the axis of the jet is much greater than the fluid velocity in the direction of the jet axis. This is caused from the fluid entrainment surrounding the jet. The fact that the boundary layer equations don't hold outside of the core of

the jet is seldom mentioned in the literature, and the similarity solution for the jet is often used to evaluate the performance of a turbulence model for the plane jet case. The similarity solutions provide a system of equations that can be quickly solved to give the self-similar jet profile. However, in this work, a full two-dimensional RANS solver will be used to evaluate the performance of the new turbulence model. This solver will give more accurate results for the spreading rate of the jet because it is not based on the similarity solution.

This flow case in two dimensions can be constructed on a rectangular domain by assigning a separate boundary condition to each side of the domain. For our purposes here, we assume the jet is moving in the x -direction and the y -coordinate is measured normal to the centerline of the jet. Along the west side of the domain, an initial profile for the jet is specified along with inlet conditions for any flow parameters. This initial profile can be taken from the similarity solution for the jet or from experimental data. Along the south side of the domain, a symmetry boundary condition is applied. The east and north sides of the domain are specified as pressure boundary conditions. Figure C.9 shows the setup for this case in Cartesian coordinates.

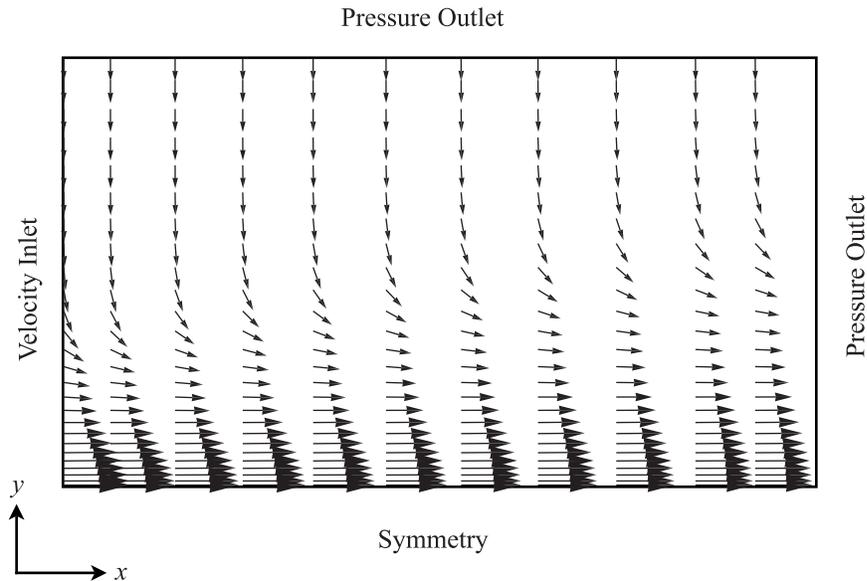


Fig. C.9 Plane jet flow case description.

Some of the most significant parameters of interest in plane jet flow are the prediction of the jet velocity along the centerline of the jet, the prediction of the jet spread rate, and the prediction of the jet velocity profile. Defining w as the span of the jet in the third dimension, the specific x -momentum flux per unit span is defined as

$$K \equiv \frac{M/w}{\rho} = \int_{-\infty}^{\infty} \bar{V}_x^2 dy \quad (\text{C.88})$$

where M is the total momentum flux. From conservation of momentum, the momentum flux across each x cross section must remain constant. The mass flux across each x cross section is a function of the distance downstream of the jet because as the jet moves downstream, it entrains flow. The volume flow rate is defined as

$$Q = w \int_{-\infty}^{\infty} \bar{V}_x dy \quad (\text{C.89})$$

Because of fluid entrainment, the volume flow rate increases as the fluid moves downstream. The volume flow rate is commonly written in nondimensional form to give a local Reynolds number of the jet flow

$$R_x \equiv \frac{Q/w}{\nu} \quad (\text{C.90})$$

In order to define a spread rate for the jet, the width of the jet must be defined. Here we define the jet width, y_h , as the y -coordinate at which the velocity in the direction of the jet centerline is equal to half the velocity along the jet centerline

$$\bar{V}_x(x, y_h) = 0.5\bar{V}_x(x, 0) \quad (\text{C.91})$$

The spread rate can then be evaluated by plotting the jet width as a function of distance along the jet centerline. The jet velocity profile is commonly reported in nondimensional form where the velocity has been nondimensionalized by the centerline velocity, and the distance from the jet centerline is nondimensionalized by the jet width. Results for the plane jet can be compared to data by Bradbury [62] and Heskestad [63].

B. Laminar Flow

An approximation to the laminar plane jet solution can be found by using boundary layer theory and applying a similarity analysis. Schlichting [108] gives a solution where the velocity field can be written as

$$\begin{aligned}\bar{V}_x &= \left(\frac{3K^2}{32\nu x} \right)^{1/3} \operatorname{sech}^2(\gamma) \\ \bar{V}_y &= \left(\frac{K\nu}{6x^2} \right)^{1/3} [2\gamma \operatorname{sech}^2(\gamma) - \tanh(\gamma)]\end{aligned}\tag{C.92}$$

where

$$\gamma \equiv \left(\frac{K}{48\nu^2} \right)^{1/3} \frac{y}{x^{2/3}}\tag{C.93}$$

Using this solution in Eq. (C.89), the volume flow rate per unit span of the jet is

$$Q/w = \int_{-\infty}^{\infty} \bar{V}_x dy = (36K\nu x)^{1/3}\tag{C.94}$$

and the local Reynolds number of the jet flow is

$$R_x \equiv \frac{Q/w}{\nu} = \left(\frac{36Kx}{\nu^2} \right)^{1/3}\tag{C.95}$$

The right-hand side of this expression reveals an important length scale for the plane jet case. Because both K and ν are constants related to the flow field, they can be used to give a scale by which length parameters can be nondimensionalized. The nondimensional form of a length parameter where this characteristic length has been used will be given the symbol H . For example, the nondimensional position, x , can be expressed as

$$H_x \equiv \frac{Kx}{\nu^2}\tag{C.96}$$

Using Eq. (C.95) in Eqs. (C.92) and (C.93), the velocity field can be rewritten in terms of Reynolds number

$$\begin{aligned}\bar{V}_x &= \frac{3}{2} \frac{K}{\nu R_x} \operatorname{sech}^2(\gamma) \\ \bar{V}_y &= \frac{R_x \nu}{6x} [2\gamma \operatorname{sech}^2(\gamma) - \tanh(\gamma)]\end{aligned}\tag{C.97}$$

where

$$\gamma \equiv \frac{R_x y}{12 x} \quad (\text{C.98})$$

At the centerline, $\gamma = 0$ which can be used in Eq. (C.97) to give the centerline velocity of the jet.

$$\bar{V}_c \equiv \bar{V}_x(x,0) = \frac{3 K}{2 \nu R_x} \quad (\text{C.99})$$

The x -velocity profile can be nondimensionalized by dividing the profile by the centerline velocity yielding

$$\frac{\bar{V}_x}{\bar{V}_c} = \frac{\frac{3 K}{2 \nu R_x} \operatorname{sech}^2(\gamma)}{\frac{3 K}{2 \nu R_x}} = \operatorname{sech}^2(\gamma) \quad (\text{C.100})$$

Here we define the width of the jet to be the point at which the x -velocity is 50% of the centerline velocity

$$\frac{\bar{V}_x}{\bar{V}_c} = \operatorname{sech}^2(\gamma) = 0.5 \Rightarrow \gamma = \operatorname{sech}^{-1}(\sqrt{0.5}) \cong 0.88 \quad (\text{C.101})$$

Using this in Eq. (C.98) gives the y -coordinate at which the x -velocity of the jet is only 50% of the centerline velocity

$$y_h = \operatorname{sech}^{-1}(\sqrt{0.5}) \frac{12x}{R_x} \cong 10.58 \frac{x}{R_x} \quad (\text{C.102})$$

As a side note, it is sometimes useful to know the point at which the x -velocity of the jet is only 1% of the centerline velocity. Following the procedure above, we obtain

$$\begin{aligned} \frac{\bar{V}_x}{\bar{V}_c} = \operatorname{sech}^2(\gamma) = 0.01 \Rightarrow \gamma = \operatorname{sech}^{-1}(\sqrt{0.01}) \cong 2.99 \\ y_{1\%} = \operatorname{sech}^{-1}(\sqrt{0.01}) \frac{12x}{R_x} \cong 35.92 \frac{x}{R_x} \end{aligned} \quad (\text{C.103})$$

The most significant parameters of this flowfield can be plotted in nondimensional format. Once the width of the jet has been found, the nondimensional velocity profile at any x -coordinate can be found

$$\frac{\bar{V}_x}{\bar{V}_c} = \operatorname{sech}^2 \left[\frac{R_x y}{12x} \left(\frac{10.58x/R_x}{y_h} \right) \right] = \operatorname{sech}^2 \left(0.88 \frac{y}{y_h} \right) \quad (\text{C.104})$$

The velocity at the centerline as a function of x can be written in nondimensional form by rearranging Eq. (C.99)

$$\frac{\bar{V}_c V}{K} = \frac{3}{2R_x} = \left(\frac{3}{32}\right)^{1/3} \left(\frac{Kx}{\nu^2}\right)^{-1/3} = \left(\frac{3}{32H_x}\right)^{1/3} \quad (\text{C.105})$$

Finally, the jet width as a function of x can be written in nondimensional form by dividing Eq. (C.102) by the length scale given in Eq. (C.96)

$$\frac{Ky_h}{\nu^2} = \text{sech}^{-1}(\sqrt{0.5}) \frac{12x}{R_x} \frac{K}{\nu^2} \cong 0.88 \frac{12H_x}{R_x} = 0.88 \frac{12H_x}{(36H_x)^{1/3}} = 0.88(48H_x^2)^{1/3} \quad (\text{C.106})$$

A laminar plane jet case was run in ICESS using the following properties

$$\mu = 0.001 \text{ kg}/(\text{m}\cdot\text{s}), \quad \rho = 0.1 \text{ kg}/\text{m}^3, \quad K = 10 \text{ m}^3/\text{s}^2 \quad (\text{C.107})$$

The case was run on a grid of 100x100 cells and a grid of 200x200 cells using logarithmic clustering near the wall. Figure C.10 shows the nondimensional velocity profile solution of the fine grid at several locations along the jet. The analytical solution is that given in Eq. (C.104). Note that in the core of the jet, the results match the analytical solution almost perfectly. However, far from the jet, the numerical solution deviates somewhat from the analytical solution. This deviation is due to the assumptions made in order to develop the analytical solution. Figure C.11 shows the nondimensional centerline velocity as a function of distance along the jet centerline where the analytical solution is that given in Eq. (C.105). Figure C.12 shows the nondimensional spread rate of the jet as a function of distance along the jet where the analytical solution is given in Eq. (C.106).

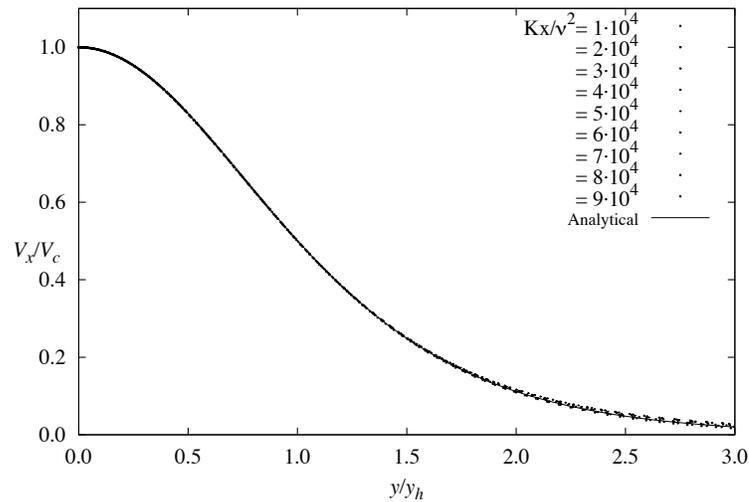


Fig. C.10 Numerical results for the nondimensional x -velocity profile for the plane laminar jet.

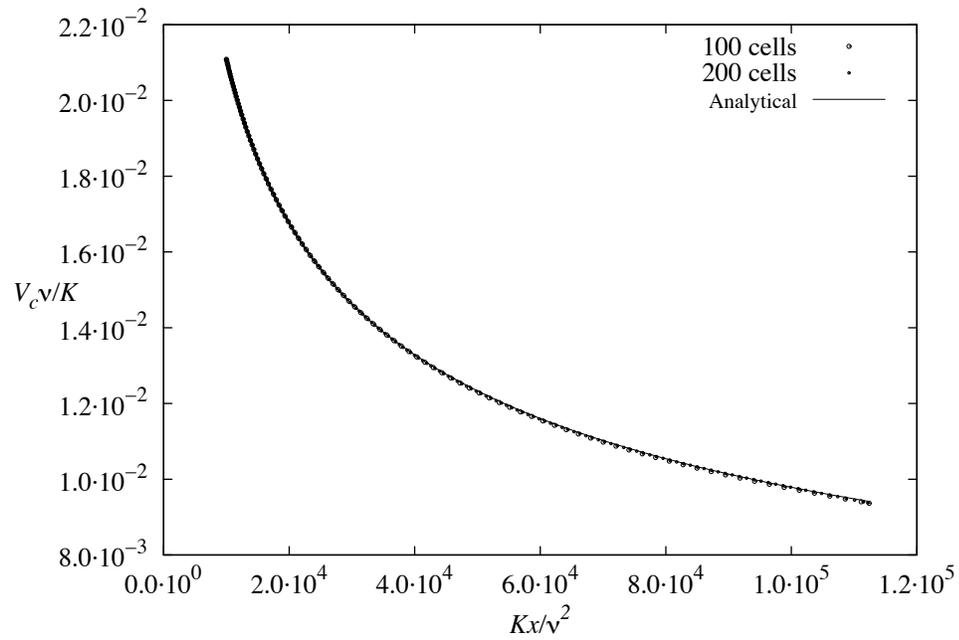


Fig. C.11 Numerical results for the nondimensional plane laminar jet centerline velocity.

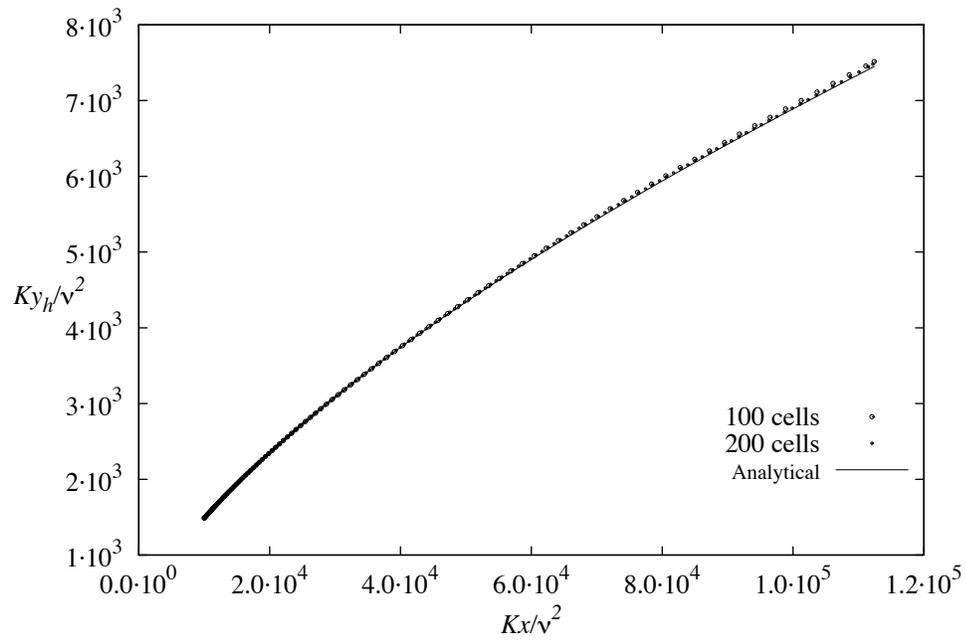


Fig. C.12 Numerical results for the nondimensional plane laminar jet spread rate.

V. Round Jet Flow

A. Case Description

The round jet flow case is very similar to the plane jet, but examines an axisymmetric jet rather than a two-dimensional jet. Just like the plane jet, as the jet of fluid advances into the quiescent fluid, the momentum of the jet is diffused outward normal to the jet axis. The jet centerline velocity decreases as the flow moves downstream, and the radius of the jet grows. Eventually, the core of the jet profile becomes self-similar. Because the jet is axisymmetric, the case can be simplified to a two-dimensional problem in cylindrical coordinates. The boundary layer equations are often applied to this case in order to develop a similarity solution for the jet profile. However, just like the case of the plane jet, these equations do not hold far from the jet centerline. The similarity solutions provide a system of equations that can be quickly solved to give the self-similar jet profile. However, in this work, an axisymmetric two-dimensional RANS solver will be used to evaluate the performance of the new turbulence model. This solver will give more accurate results for the spreading rate of the jet because it is not based on the similarity solution.

This flow case in two dimensions can be constructed on a rectangular domain by assigning a separate boundary condition to each side of the domain. We assume the jet is moving in the z -direction and the r -coordinate is measured normal to the centerline of the jet. Along the west side of the domain, an initial profile for the jet is specified along with inlet conditions for any flow parameters. This initial profile can be taken from the similarity solution for the jet or from experimental data. Along the south side of the domain, a symmetry boundary condition is applied. The south side of the domain is the axis of rotation for the case. The east and north sides of the domain are specified as pressure boundary conditions. Figure C.13 shows the setup for this case in cylindrical coordinates.

Some of the most significant parameters of interest in plane jet flow are the prediction of the jet velocity along the centerline of the jet, the prediction of the jet spread rate, and the prediction of the jet velocity profile. The specific z -momentum flux is defined as

$$K \equiv \frac{M}{\rho} = \int_0^{\infty} \bar{V}_z^2 2\pi r dr \quad (\text{C.108})$$

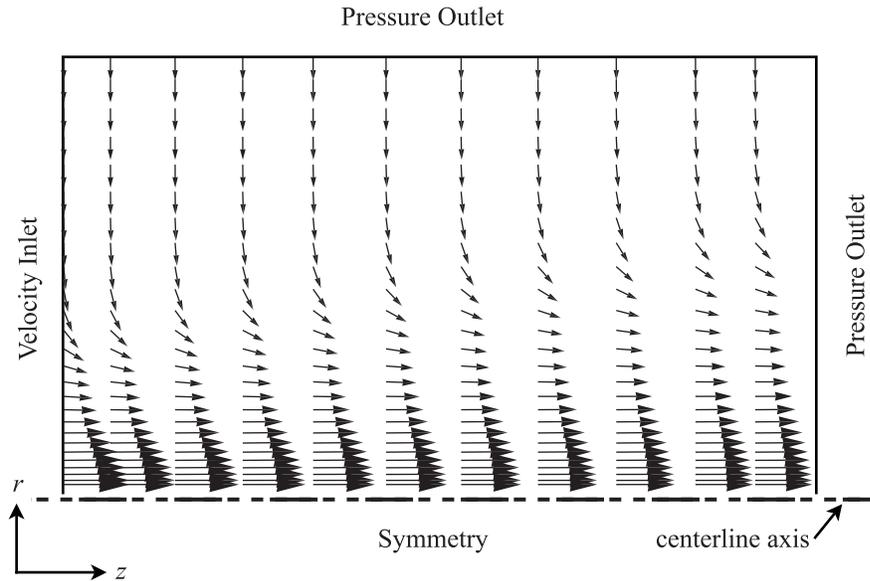


Fig. C.13 Round jet flow case description.

where M is the momentum flux. From conservation of momentum, the momentum flux across each z cross section must remain constant. The mass flux across each z cross section is a function of the distance downstream of the jet because as the jet moves downstream, it entrains flow. The volume flow rate is defined as

$$Q = \int_0^{\infty} \bar{V}_z 2\pi r dr \quad (\text{C.109})$$

Results for the round jet can be compared to data by Wagnanski and Fiedler [64] and Rodi [65].

B. Laminar Flow

An approximation for the laminar round jet can be found by using boundary layer theory and applying a similarity analysis. Schlichting [108] gives a solution where the velocity field can be written as

$$\begin{aligned} \bar{V}_z &= \frac{3K}{8\pi\nu z} \frac{1}{(1+\gamma^2)^2} \\ \bar{V}_r &= \frac{1}{2z} \left(\frac{3K}{\pi} \right)^{1/2} \frac{\gamma(1-\gamma^2)}{(1+\gamma^2)^2} \end{aligned} \quad (\text{C.110})$$

where

$$\gamma \equiv \left(\frac{3K}{64\pi\nu^2} \right)^{1/2} \frac{r}{z} \quad (\text{C.111})$$

Using Eq. (C.110) in Eq. (C.109) gives the volume flow rate

$$Q = \int_0^{\infty} \bar{V}_z 2\pi r dr = 8\pi\nu z \quad (\text{C.112})$$

This shows that the volume flow rate is directly proportional to the distance along the centerline from the virtual origin. In general, the position of the virtual origin is not known a priori. However, the initial volume flow rate at the physical origin of the jet, z_o , can be defined as

$$Q_o = 8\pi\nu z_o \quad (\text{C.113})$$

Any downstream position, \hat{z} , measured relative to the physical origin of the jet can now be expressed relative to the virtual origin, $z = 0$, as

$$z = \hat{z} + \frac{Q_o}{8\pi\nu} \quad (\text{C.114})$$

At the centerline, $\gamma = 0$ which can be used in Eq. (C.110) to give the centerline velocity of the jet.

$$\bar{V}_c \equiv \bar{V}_z(z,0) = \frac{3K}{8\pi\nu z} \quad (\text{C.115})$$

This can be rewritten in terms of the distance downstream of the physical origin by applying Eq. (C.114)

$$\bar{V}_c = \frac{3K}{8\pi\nu} \frac{1}{\left(\hat{z} + \frac{Q_o}{8\pi\nu} \right)} = \frac{3}{8\pi} \left(\frac{K^2}{Q_o\nu^2} \right) \frac{1}{\left(\frac{K\hat{z}}{Q_o\nu} + \frac{K}{8\pi\nu^2} \right)} \quad (\text{C.116})$$

This can be rearranged to give the nondimensional form of the centerline velocity as a function of z .

$$\frac{Q_o\nu^2\bar{V}_c}{K^2} = \frac{3}{8\pi \left(\frac{K\hat{z}}{Q_o\nu} + \frac{K}{8\pi\nu^2} \right)} \quad (\text{C.117})$$

The z -velocity profile can be written in nondimensional form by dividing the profile by the centerline velocity. This gives

$$\frac{\bar{V}_z}{\bar{V}_c} = \frac{\frac{3K}{8\pi\nu z} \frac{1}{(1+\gamma^2)^2}}{\frac{3K}{8\pi\nu z}} = \frac{1}{(1+\gamma^2)^2} \quad (\text{C.118})$$

Using Eq. (C.112) in Eq. (C.111) gives the relation

$$\gamma \equiv (3\pi)^{1/2} \frac{K^{1/2}r}{Q} \quad (\text{C.119})$$

Applying this to Eq. (C.118) gives the nondimensional form of the jet velocity profile

$$\frac{\bar{V}_z}{\bar{V}_c} = \frac{1}{\left[1 + 3\pi \left(\frac{K^{1/2}r}{Q}\right)^2\right]^2} \quad (\text{C.120})$$

Here we define the width of the jet to be the point at which the z -velocity is 50% of the centerline velocity

$$\frac{\bar{V}_z}{\bar{V}_c} = \frac{1}{(1 + \gamma^2)^2} = 0.5 \Rightarrow \gamma = (2^{1/2} - 1)^{1/2} \cong 0.64 \quad (\text{C.121})$$

Using this in Eq. (C.113) gives the r -coordinate at which the z -velocity of the jet is only 50% of the centerline velocity

$$r_h = z \left[\frac{64\pi v^2 (2^{1/2} - 1)}{3K} \right]^{1/2} = 2.97z \left(\frac{\pi v^2}{K} \right)^{1/2} \quad (\text{C.122})$$

Using Eq. (C.114) in Eq. (C.122) and rearranging gives a nondimensional form of the spread rate

$$\frac{Kr_h}{Q_o v} = 2.97 \left(\frac{\pi v^2}{K} \right)^{1/2} \left(\frac{Kz}{Q_o v} + \frac{K}{8\pi v^2} \right) \quad (\text{C.123})$$

As a side note, it is sometimes useful to know the point at which the z -velocity of the jet is only 1% of the centerline velocity. Following the procedure above, we obtain

$$\begin{aligned} \frac{\bar{V}_z}{\bar{V}_c} &= \frac{1}{(1 + \gamma^2)^2} = 0.01 \Rightarrow \gamma = 3 \\ r_{1\%} &= 3z \left(\frac{64\pi v^2}{3K} \right)^{1/2} \end{aligned} \quad (\text{C.124})$$

A laminar round jet case was run in ICESSE using the following properties

$$\mu = 0.001 \text{ kg}/(\text{m} \cdot \text{s}), \quad \rho = 0.1 \text{ kg}/\text{m}^3, \quad K = 10 \text{ m}^4/\text{s}^2, \quad Q_o = 0.25 \text{ m}^3/\text{s} \quad (\text{C.125})$$

The case was run on a grid of 100x100 cells and a grid of 200x200 cells using logarithmic clustering near the jet center. Figure C.14 shows the nondimensional velocity profile solution of the fine grid at several

locations along the jet. The analytical solution is that given in Eq. (C.120). Figure C.15 shows the nondimensional centerline velocity as a function of distance along the jet centerline where the analytical solution is that given in Eq. (C.117). Figure C.16 shows the nondimensional spread rate of the jet as a function of distance along the jet where the analytical solution is given in Eq. (C.123).

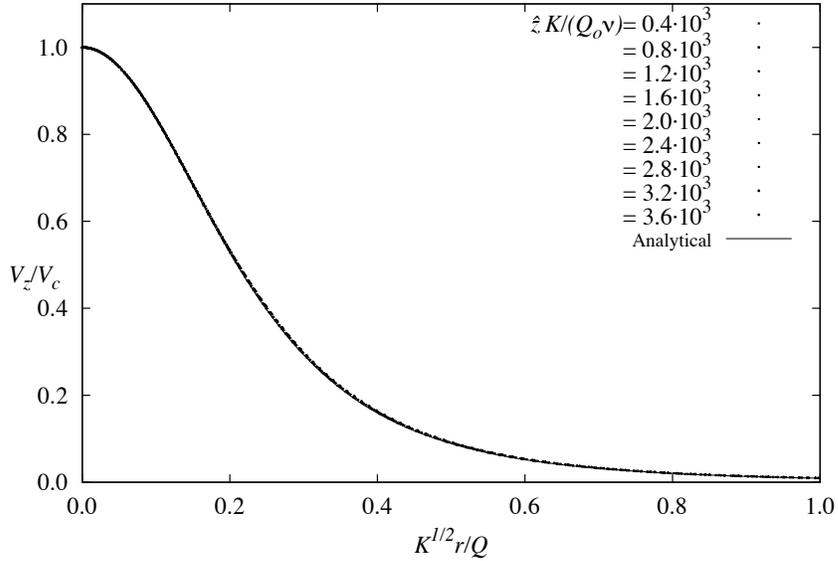


Fig. C.14 Numerical results for the nondimensional z -velocity profile for the round laminar jet.

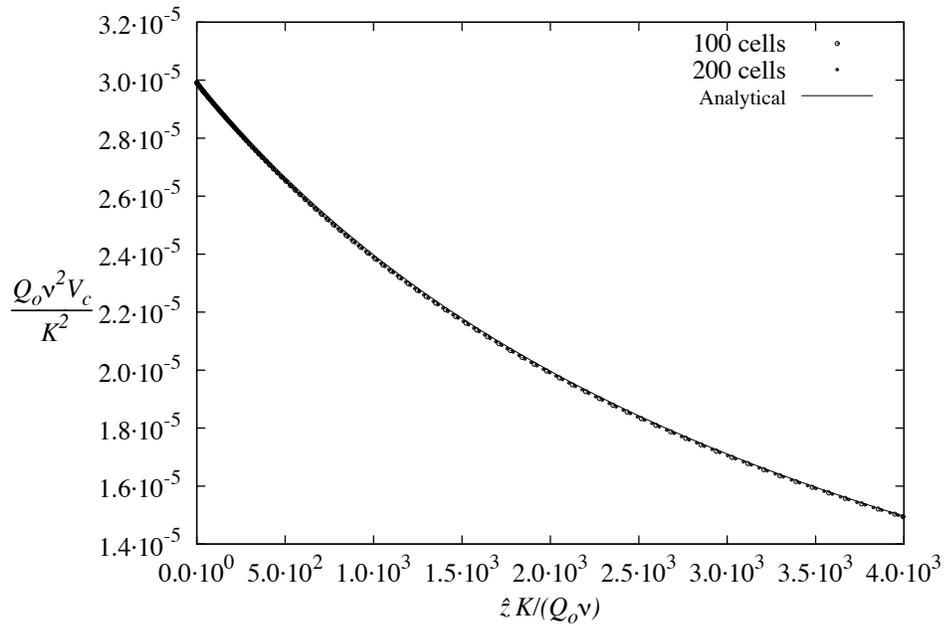


Fig. C.15 Numerical results for the nondimensional round laminar jet centerline velocity.

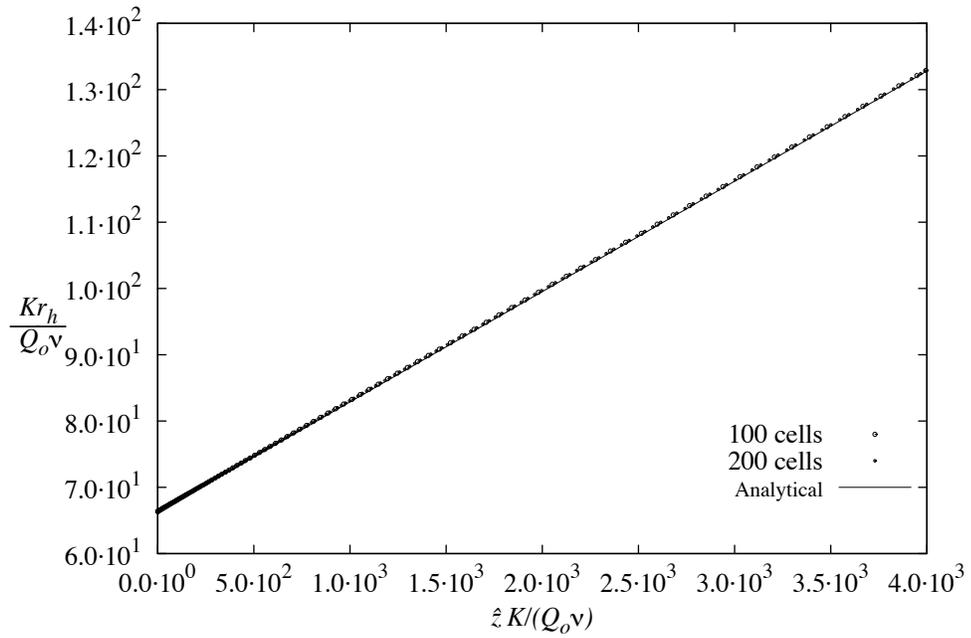


Fig. C.16 Numerical results for the nondimensional round laminar jet spread rate.

APPENDIX D

ONE-DIMENSIONAL FINITE-DIFFERENCE FORMULATIONS

I. A Second-Order Formulation

A general formulation for fully developed pipe and channel flow can be written to allow implementation of several two-equation turbulence models. This formulation takes the form

$$\begin{aligned}
y^+ &= R_\tau(1-\hat{r}), \quad v^+ = f_v, \quad u^{+'} = -\frac{R_\tau \hat{r}}{(1+v^+)} \\
-\frac{v^{+'}}{\sigma_k} k^{+'} - \frac{1}{\hat{r}} \left(1 + \frac{v^+}{\sigma_k}\right) k^{+'} - \left(1 + \frac{v^+}{\sigma_k}\right) k^{+''} &= S_k \\
-\frac{v^{+'}}{\sigma_\omega} h^{+'} - \frac{1}{\hat{r}} \left(1 + \frac{v^+}{\sigma_h}\right) h^{+'} - \left(1 + \frac{v^+}{\sigma_h}\right) h^{+''} &= S_h \\
u^+(1) = 0, \quad k^+(1) = 0, \quad k^{+'}(1) = 0, \quad k^{+'}(0) = 0, \quad h^{+'}(0) = 0
\end{aligned} \tag{D.1}$$

where h^+ represents the second turbulence variable, the primes represent derivatives with respect to \hat{r} , and f_v , S_k , and S_h are model-dependent functions. Equation (D.1) can be solved numerically using the finite-difference method on a non-uniform grid. Here we define a one-dimensional domain discretized by m nodes. Node 1 is located at $\hat{r} = 0$, and node m is located at $\hat{r} = 1$. Applying second-order forward difference approximations, the first and second derivatives of any variable, ϕ , can be approximated at node 1 as

$$\begin{aligned}
\delta \hat{r}_a &\equiv \hat{r}_2 - \hat{r}_1, \quad \delta \hat{r}_b \equiv \hat{r}_3 - \hat{r}_1, \quad \delta \hat{r}_c \equiv \hat{r}_4 - \hat{r}_1 \\
\phi_1' &= \frac{\delta \hat{r}_b^2 - \delta \hat{r}_a^2}{\delta \hat{r}_a^2 \delta \hat{r}_b - \delta \hat{r}_b^2 \delta \hat{r}_a} \phi_1 + \frac{-\delta \hat{r}_b^2}{\delta \hat{r}_a^2 \delta \hat{r}_b - \delta \hat{r}_b^2 \delta \hat{r}_a} \phi_2 + \frac{\delta \hat{r}_a^2}{\delta \hat{r}_a^2 \delta \hat{r}_b - \delta \hat{r}_b^2 \delta \hat{r}_a} \phi_3 \\
\phi_1'' &= \frac{2(\delta \hat{r}_a + \delta \hat{r}_b + \delta \hat{r}_c)}{\delta \hat{r}_a \delta \hat{r}_b \delta \hat{r}_c} \phi_1 + \frac{2(\delta \hat{r}_b + \delta \hat{r}_c)}{\delta \hat{r}_a (\delta \hat{r}_a - \delta \hat{r}_b) (\delta \hat{r}_c - \delta \hat{r}_a)} \phi_2 \\
&\quad + \frac{2(\delta \hat{r}_c + \delta \hat{r}_a)}{\delta \hat{r}_b (\delta \hat{r}_b - \delta \hat{r}_c) (\delta \hat{r}_a - \delta \hat{r}_b)} \phi_3 + \frac{2(\delta \hat{r}_a + \delta \hat{r}_b)}{\delta \hat{r}_c (\delta \hat{r}_b - \delta \hat{r}_c) (\delta \hat{r}_c - \delta \hat{r}_a)} \phi_4
\end{aligned} \tag{D.2}$$

Applying second-order forward difference approximations, the first and second derivatives of any variable, ϕ , can be approximated at node 2 as

$$\begin{aligned}
\delta\hat{r}_a &\equiv \hat{r}_1 - \hat{r}_2, \quad \delta\hat{r}_b \equiv \hat{r}_3 - \hat{r}_2, \quad \delta\hat{r}_c \equiv \hat{r}_4 - \hat{r}_2 \\
\phi_2' &= \frac{-\delta\hat{r}_b^2}{\delta\hat{r}_a^2\delta\hat{r}_b - \delta\hat{r}_b^2\delta\hat{r}_a} \phi_1 + \frac{\delta\hat{r}_b^2 - \delta\hat{r}_a^2}{\delta\hat{r}_a^2\delta\hat{r}_b - \delta\hat{r}_b^2\delta\hat{r}_a} \phi_2 + \frac{\delta\hat{r}_a^2}{\delta\hat{r}_a^2\delta\hat{r}_b - \delta\hat{r}_b^2\delta\hat{r}_a} \phi_3 \\
\phi_2'' &= \frac{2(\delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a(\delta\hat{r}_a - \delta\hat{r}_b)(\delta\hat{r}_c - \delta\hat{r}_a)} \phi_1 + \frac{2(\delta\hat{r}_a + \delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a\delta\hat{r}_b\delta\hat{r}_c} \phi_2 \\
&\quad + \frac{2(\delta\hat{r}_c + \delta\hat{r}_a)}{\delta\hat{r}_b(\delta\hat{r}_b - \delta\hat{r}_c)(\delta\hat{r}_a - \delta\hat{r}_b)} \phi_3 + \frac{2(\delta\hat{r}_a + \delta\hat{r}_b)}{\delta\hat{r}_c(\delta\hat{r}_b - \delta\hat{r}_c)(\delta\hat{r}_c - \delta\hat{r}_a)} \phi_4
\end{aligned} \tag{D.3}$$

Applying second-order finite difference approximations, the first and second derivatives of any variable, ϕ , can be approximated at an interior node, j , as

$$\begin{aligned}
&\text{for } 3 \leq j \leq m-1 \\
\delta\hat{r}_a &\equiv \hat{r}_{j-1} - \hat{r}_j, \quad \delta\hat{r}_b \equiv \hat{r}_{j+1} - \hat{r}_j, \quad \delta\hat{r}_c \equiv \hat{r}_{j-2} - \hat{r}_j \\
\phi_j' &= \frac{-\delta\hat{r}_b^2}{\delta\hat{r}_a^2\delta\hat{r}_b - \delta\hat{r}_b^2\delta\hat{r}_a} \phi_{j-1} + \frac{\delta\hat{r}_b^2 - \delta\hat{r}_a^2}{\delta\hat{r}_a^2\delta\hat{r}_b - \delta\hat{r}_b^2\delta\hat{r}_a} \phi_j + \frac{\delta\hat{r}_a^2}{\delta\hat{r}_a^2\delta\hat{r}_b - \delta\hat{r}_b^2\delta\hat{r}_a} \phi_{j+1} \\
\phi_j'' &= \frac{2(\delta\hat{r}_a + \delta\hat{r}_b)}{\delta\hat{r}_c(\delta\hat{r}_b - \delta\hat{r}_c)(\delta\hat{r}_c - \delta\hat{r}_a)} \phi_{j-2} + \frac{2(\delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a(\delta\hat{r}_a - \delta\hat{r}_b)(\delta\hat{r}_c - \delta\hat{r}_a)} \phi_{j-1} \\
&\quad + \frac{2(\delta\hat{r}_a + \delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a\delta\hat{r}_b\delta\hat{r}_c} \phi_j + \frac{2(\delta\hat{r}_c + \delta\hat{r}_a)}{\delta\hat{r}_b(\delta\hat{r}_b - \delta\hat{r}_c)(\delta\hat{r}_a - \delta\hat{r}_b)} \phi_{j+1}
\end{aligned} \tag{D.4}$$

Applying second-order backward difference approximations, the first and second derivatives of any variable, ϕ , can be approximated at node m as

$$\begin{aligned}
\delta\hat{r}_a &\equiv \hat{r}_{m-1} - \hat{r}_m, \quad \delta\hat{r}_b \equiv \hat{r}_{m-2} - \hat{r}_m, \quad \delta\hat{r}_c \equiv \hat{r}_{m-3} - \hat{r}_m \\
\phi_m' &= \frac{\delta\hat{r}_a^2}{\delta\hat{r}_a^2\delta\hat{r}_b - \delta\hat{r}_b^2\delta\hat{r}_a} \phi_{m-2} + \frac{-\delta\hat{r}_b^2}{\delta\hat{r}_a^2\delta\hat{r}_b - \delta\hat{r}_b^2\delta\hat{r}_a} \phi_{m-1} + \frac{\delta\hat{r}_b^2 - \delta\hat{r}_a^2}{\delta\hat{r}_a^2\delta\hat{r}_b - \delta\hat{r}_b^2\delta\hat{r}_a} \phi_m \\
\phi_m'' &= \frac{2(\delta\hat{r}_a + \delta\hat{r}_b)}{\delta\hat{r}_c(\delta\hat{r}_b - \delta\hat{r}_c)(\delta\hat{r}_c - \delta\hat{r}_a)} \phi_{m-3} + \frac{2(\delta\hat{r}_c + \delta\hat{r}_a)}{\delta\hat{r}_b(\delta\hat{r}_b - \delta\hat{r}_c)(\delta\hat{r}_a - \delta\hat{r}_b)} \phi_{m-2} \\
&\quad + \frac{2(\delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a(\delta\hat{r}_a - \delta\hat{r}_b)(\delta\hat{r}_c - \delta\hat{r}_a)} \phi_{m-1} + \frac{2(\delta\hat{r}_a + \delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a\delta\hat{r}_b\delta\hat{r}_c} \phi_m
\end{aligned} \tag{D.5}$$

Any number of algorithms can be used for grid generation. An algorithm that works well for the present study clusters the nodes near the wall and can be written as

$$\begin{aligned}
&\text{for } 1 \leq j \leq m \\
\eta_j &= \frac{j-1}{m-1}, \quad \hat{r}_{m+1-j} = 1 - \frac{(\beta+1) - (\beta-1)[(\beta+1)/(\beta-1)]^{1-\eta_j}}{1 + [(\beta+1)/(\beta-1)]^{1-\eta_j}}, \quad y_{m+1-j}^+ = R_c(1 - \hat{r}_{m+1-j})
\end{aligned} \tag{D.6}$$

where β is a clustering parameter.

Once the \hat{r} array has been filled out, the following coefficient arrays can also be filled out

$$\begin{aligned}
\delta\hat{r}_a &\equiv \hat{r}_2 - \hat{r}_1, \quad \delta\hat{r}_b \equiv \hat{r}_3 - \hat{r}_1, \quad \delta\hat{r}_c \equiv \hat{r}_4 - \hat{r}_1 \\
C_{L2,1} &\equiv 0, \quad C_{L1,1} \equiv 0, \quad C_{D,1} \equiv \frac{(\delta\hat{r}_a + \delta\hat{r}_b)}{\delta\hat{r}_a \delta\hat{r}_b}, \\
C_{U1,1} &\equiv \frac{-\delta\hat{r}_b}{\delta\hat{r}_a (\delta\hat{r}_a - \delta\hat{r}_b)}, \quad C_{U2,1} \equiv \frac{\delta\hat{r}_a}{\delta\hat{r}_b (\delta\hat{r}_a - \delta\hat{r}_b)} \\
A_{L3,1} &\equiv 0, \quad A_{L2,1} \equiv 0, \quad A_{L1,1} \equiv 0, \quad A_{D,1} \equiv \frac{2(\delta\hat{r}_a + \delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a \delta\hat{r}_b \delta\hat{r}_c}, \quad A_{U1,1} \equiv \frac{2(\delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a (\delta\hat{r}_a - \delta\hat{r}_b) (\delta\hat{r}_c - \delta\hat{r}_a)} \\
A_{U2,1} &\equiv \frac{2(\delta\hat{r}_c + \delta\hat{r}_a)}{\delta\hat{r}_b (\delta\hat{r}_b - \delta\hat{r}_c) (\delta\hat{r}_a - \delta\hat{r}_b)}, \quad A_{U3,1} \equiv \frac{2(\delta\hat{r}_a + \delta\hat{r}_b)}{\delta\hat{r}_c (\delta\hat{r}_b - \delta\hat{r}_c) (\delta\hat{r}_c - \delta\hat{r}_a)}
\end{aligned} \tag{D.7}$$

$$\begin{aligned}
\delta\hat{r}_a &\equiv \hat{r}_1 - \hat{r}_2, \quad \delta\hat{r}_b \equiv \hat{r}_3 - \hat{r}_2, \quad \delta\hat{r}_c \equiv \hat{r}_4 - \hat{r}_2 \\
C_{L2,2} &\equiv 0, \quad C_{L1,2} \equiv \frac{-\delta\hat{r}_b}{\delta\hat{r}_a (\delta\hat{r}_a - \delta\hat{r}_b)}, \quad C_{D,2} \equiv \frac{-(\delta\hat{r}_a + \delta\hat{r}_b)}{\delta\hat{r}_a \delta\hat{r}_b}, \\
C_{U1,2} &\equiv \frac{\delta\hat{r}_a}{\delta\hat{r}_b (\delta\hat{r}_a - \delta\hat{r}_b)}, \quad C_{U2,2} \equiv 0 \\
A_{L3,2} &\equiv 0, \quad A_{L2,2} \equiv 0, \quad A_{L1,2} \equiv \frac{2(\delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a (\delta\hat{r}_a - \delta\hat{r}_b) (\delta\hat{r}_c - \delta\hat{r}_a)}, \quad A_{D,2} \equiv \frac{2(\delta\hat{r}_a + \delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a \delta\hat{r}_b \delta\hat{r}_c}, \\
A_{U1,2} &\equiv \frac{2(\delta\hat{r}_c + \delta\hat{r}_a)}{\delta\hat{r}_b (\delta\hat{r}_b - \delta\hat{r}_c) (\delta\hat{r}_a - \delta\hat{r}_b)}, \quad A_{U2,2} \equiv \frac{2(\delta\hat{r}_a + \delta\hat{r}_b)}{\delta\hat{r}_c (\delta\hat{r}_b - \delta\hat{r}_c) (\delta\hat{r}_c - \delta\hat{r}_a)}, \quad A_{U3,2} \equiv 0
\end{aligned} \tag{D.8}$$

for $3 \leq j \leq m-1$

$$\begin{aligned}
\delta\hat{r}_a &\equiv \hat{r}_{j-1} - \hat{r}_j, \quad \delta\hat{r}_b \equiv \hat{r}_{j+1} - \hat{r}_j, \quad \delta\hat{r}_c \equiv \hat{r}_{j-2} - \hat{r}_j \\
C_{L2,j} &\equiv 0, \quad C_{L1,j} \equiv \frac{-\delta\hat{r}_b}{\delta\hat{r}_a (\delta\hat{r}_a - \delta\hat{r}_b)}, \quad C_{D,j} \equiv \frac{-(\delta\hat{r}_a + \delta\hat{r}_b)}{\delta\hat{r}_a \delta\hat{r}_b}, \\
C_{U1,j} &\equiv \frac{\delta\hat{r}_a}{\delta\hat{r}_b (\delta\hat{r}_a - \delta\hat{r}_b)}, \quad C_{U2,j} \equiv 0 \\
A_{L3,j} &\equiv 0, \quad A_{L2,j} \equiv \frac{2(\delta\hat{r}_a + \delta\hat{r}_b)}{\delta\hat{r}_c (\delta\hat{r}_b - \delta\hat{r}_c) (\delta\hat{r}_c - \delta\hat{r}_a)}, \quad A_{L1,j} \equiv \frac{2(\delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a (\delta\hat{r}_a - \delta\hat{r}_b) (\delta\hat{r}_c - \delta\hat{r}_a)}, \\
A_{D,j} &\equiv \frac{2(\delta\hat{r}_a + \delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a \delta\hat{r}_b \delta\hat{r}_c}, \quad A_{U1,j} \equiv \frac{2(\delta\hat{r}_c + \delta\hat{r}_a)}{\delta\hat{r}_b (\delta\hat{r}_b - \delta\hat{r}_c) (\delta\hat{r}_a - \delta\hat{r}_b)}, \quad A_{U2,j} \equiv 0, \quad A_{U3,j} \equiv 0
\end{aligned} \tag{D.9}$$

$$\begin{aligned}
\delta\hat{r}_a &\equiv \hat{r}_{m-1} - \hat{r}_m, \quad \delta\hat{r}_b \equiv \hat{r}_{m-2} - \hat{r}_m, \quad \delta\hat{r}_c \equiv \hat{r}_{m-3} - \hat{r}_m \\
C_{L2,m} &\equiv \frac{\delta\hat{r}_a}{\delta\hat{r}_b (\delta\hat{r}_a - \delta\hat{r}_b)}, \quad C_{L1,m} \equiv \frac{-\delta\hat{r}_b}{\delta\hat{r}_a (\delta\hat{r}_a - \delta\hat{r}_b)}, \quad C_{D,m} \equiv \frac{-(\delta\hat{r}_a + \delta\hat{r}_b)}{\delta\hat{r}_a \delta\hat{r}_b}, \\
C_{U1,m} &\equiv 0, \quad C_{U2,m} \equiv 0 \\
A_{L3,m} &\equiv \frac{2(\delta\hat{r}_a + \delta\hat{r}_b)}{\delta\hat{r}_c (\delta\hat{r}_b - \delta\hat{r}_c) (\delta\hat{r}_c - \delta\hat{r}_a)}, \quad A_{L2,m} \equiv \frac{2(\delta\hat{r}_c + \delta\hat{r}_a)}{\delta\hat{r}_b (\delta\hat{r}_b - \delta\hat{r}_c) (\delta\hat{r}_a - \delta\hat{r}_b)}, \\
A_{L1,m} &\equiv \frac{2(\delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a (\delta\hat{r}_a - \delta\hat{r}_b) (\delta\hat{r}_c - \delta\hat{r}_a)}, \quad A_{D,m} \equiv \frac{2(\delta\hat{r}_a + \delta\hat{r}_b + \delta\hat{r}_c)}{\delta\hat{r}_a \delta\hat{r}_b \delta\hat{r}_c}, \\
A_{U1,m} &\equiv 0, \quad A_{U2,m} \equiv 0, \quad A_{U3,m} \equiv 0
\end{aligned} \tag{D.10}$$

Using Eqs. (D.7)–(D.10) in Eqs. (D.2)–(D.5) gives second-order approximations for the first and second derivatives of any variable, ϕ

$$\begin{aligned}
\phi_1' &= C_{D,1}\phi_1 + C_{U,1}\phi_2 + C_{U,2}\phi_3 \\
\phi_1'' &= A_{D,1}\phi_1 + A_{U,1}\phi_2 + A_{U,2}\phi_3 + A_{U,3}\phi_4 \\
\phi_2' &= C_{L,2}\phi_1 + C_{D,2}\phi_2 + C_{U,2}\phi_3 \\
\phi_2'' &= A_{L,2}\phi_1 + A_{D,2}\phi_2 + A_{U,2}\phi_3 + A_{U,2}\phi_4 \\
&\text{for } 3 \leq j \leq m-1 \\
\phi_j' &= C_{L,j}\phi_{j-1} + C_{D,j}\phi_j + C_{U,j}\phi_{j+1} \\
\phi_j'' &= A_{L,j}\phi_{j-2} + A_{L,j}\phi_{j-1} + A_{D,j}\phi_j + A_{U,j}\phi_{j+1} \\
\phi_m' &= C_{L,m}\phi_{m-2} + C_{L,m}\phi_{m-1} + C_{D,m}\phi_m \\
\phi_m'' &= A_{L,m}\phi_{m-3} + A_{L,m}\phi_{m-2} + A_{L,m}\phi_{m-1} + A_{D,m}\phi_m
\end{aligned} \tag{D.11}$$

Beginning with initial estimates for k^+ and h^+ along with the known arrays of \hat{r} and y^+ , initial estimates for v^+ and u^+ can be found from

$$\begin{aligned}
&\text{for } 1 \leq j \leq m \\
v_j^+ &= f_v, \quad u_j^+ = -\frac{R_\tau \hat{r}_j}{(1 + v_j^+)}
\end{aligned} \tag{D.12}$$

If the model requires the second derivative of u^+ to be calculated, it can be estimated using Eq. (D.11)

$$\begin{aligned}
u_1^{+''} &= C_{D,1}u_1^{+'} + C_{U,1}u_2^{+'} + C_{U,2}u_3^{+'} \\
&\text{for } 2 \leq j \leq m-1 \\
u_j^{+''} &= C_{L,j}u_{j-1}^{+'} + C_{D,j}u_j^{+'} + C_{U,j}u_{j+1}^{+'} \\
u_m^{+''} &= C_{L,m}u_{m-2}^{+'} + C_{L,m}u_{m-1}^{+'} + C_{D,m}u_m^{+'}
\end{aligned} \tag{D.13}$$

Likewise, in light of the boundary conditions, the first derivatives for k^+ , h^+ , and v^+ can be estimated

$$\begin{aligned}
k_1^{+'} &= 0, \quad h_1^{+'} = 0, \quad v_1^{+'} = 0 \\
\text{for } 2 \leq j \leq m-1 \\
k_j^{+'} &= C_{L1,j}k_{j-1}^+ + C_{D,j}k_j^+ + C_{U1,j}k_{j+1}^+ \\
h_j^{+'} &= C_{L1,j}h_{j-1}^+ + C_{D,j}h_j^+ + C_{U1,j}h_{j+1}^+ \\
v_j^{+'} &= C_{L1,j}v_{j-1}^+ + C_{D,j}v_j^+ + C_{U1,j}v_{j+1}^+ \\
k_m^{+'} &= 0 \\
h_m^{+'} &= C_{L2,m}h_{m-2}^+ + C_{L1,m}h_{m-1}^+ + C_{D,m}h_m^+ \\
v_m^{+'} &= C_{L2,m}v_{m-2}^+ + C_{L1,m}v_{m-1}^+ + C_{D,m}v_m^+
\end{aligned} \tag{D.14}$$

The k -transport equation can now be written in terms of the second-order finite-difference approximations expressed above including the boundary conditions on k

$$\begin{aligned}
&C_{D,1}k_1^+ + C_{U1,1}k_2^+ + C_{U2,1}k_3^+ = 0 \\
&-\frac{v_2^{+'}}{\sigma_k}(C_{L1,2}k_1^+ + C_{D,2}k_2^+ + C_{U1,2}k_3^+) - \frac{1}{\hat{r}_2} \left(1 + \frac{v_2^+}{\sigma_k} \right) (C_{L1,2}k_1^+ + C_{D,2}k_2^+ + C_{U1,2}k_3^+) \\
&-\left(1 + \frac{v_2^+}{\sigma_k} \right) (A_{L1,2}k_1^+ + A_{D,2}k_2^+ + A_{U1,2}k_3^+ + A_{U2,2}k_4^+) = S_k \\
&\text{for } 3 \leq j \leq m-2 \\
&-\frac{v_j^{+'}}{\sigma_k}(C_{L1,j}k_{j-1}^+ + C_{D,j}k_j^+ + C_{U1,j}k_{j+1}^+) - \frac{1}{\hat{r}_j} \left(1 + \frac{v_j^+}{\sigma_k} \right) (C_{L1,j}k_{j-1}^+ + C_{D,j}k_j^+ + C_{U1,j}k_{j+1}^+) \\
&-\left(1 + \frac{v_j^+}{\sigma_k} \right) (A_{L2,j}k_{j-2}^+ + A_{L1,j}k_{j-1}^+ + A_{D,j}k_j^+ + A_{U1,j}k_{j+1}^+) = S_k \\
&C_{L2,m}k_{m-2}^+ + C_{L1,m}k_{m-1}^+ + C_{D,m}k_m^+ = 0 \\
&k_m^+ = 0
\end{aligned} \tag{D.15}$$

The h -transport equation can also be written in terms of the second-order finite-difference approximations expressed above including the boundary conditions on h

$$\begin{aligned}
& C_{D,1}h_1^+ + C_{U,1}h_2^+ + C_{U,2}h_3^+ = 0 \\
& -\frac{v_2^+}{\sigma_h}(C_{L1,2}h_1^+ + C_{D,2}h_2^+ + C_{U,2}h_3^+) - \frac{1}{\hat{r}_2}\left(1 + \frac{v_2^+}{\sigma_h}\right)(C_{L1,2}h_1^+ + C_{D,2}h_2^+ + C_{U,2}h_3^+) \\
& -\left(1 + \frac{v_2^+}{\sigma_h}\right)(A_{L1,2}h_1^+ + A_{D,2}h_2^+ + A_{U,2}h_3^+ + A_{U,2,2}h_4^+) = S_h \\
& \text{for } 3 \leq j \leq m - m_h \\
& -\frac{v_j^+}{\sigma_h}(C_{L1,j}h_{j-1}^+ + C_{D,j}h_j^+ + C_{U,j}h_{j+1}^+) - \frac{1}{\hat{r}_j}\left(1 + \frac{v_j^+}{\sigma_h}\right)(C_{L1,j}h_{j-1}^+ + C_{D,j}h_j^+ + C_{U,j}h_{j+1}^+) \\
& -\left(1 + \frac{v_j^+}{\sigma_h}\right)(A_{L2,j}h_{j-2}^+ + A_{L1,j}h_{j-1}^+ + A_{D,j}h_j^+ + A_{U,j}h_{j+1}^+) = S_h \\
& \text{for } m - m_h + 1 \leq j \leq m \\
& h_j^+ = f_{h0}
\end{aligned} \tag{D.16}$$

where f_{h0} is the near-wall asymptotic solution for h^+ at node j , and m_h is an integer that determines how many nodes are calculated from the near-wall solution. The integer $m_h \geq 7$ should be used if the asymptotic solution of h^+ is singular. Otherwise, $m_h = 1$ can generally be used.

The formulations given in Eqs. (D.15) and (D.16) are for fully developed flow in a pipe. This formulation can be used to calculate fully developed flow in a channel by making minor modifications. Here we define \mathbf{P} as an array that can be conditionally evaluated and used for the terms in the pipe formulation that are different than those in the channel formulation. For example, in the k -equation P_j can be evaluated as follows

$$\begin{aligned}
& \text{for } 1 \leq j \leq m \\
& \text{if pipe flow; } P_j = \frac{1}{\hat{r}_j}\left(1 + \frac{v_j^+}{\sigma_k}\right) \\
& \text{if channel flow; } P_j = 0
\end{aligned} \tag{D.17}$$

The systems of equations given in Eqs. (D.15) and (D.16) can be solved through an iterative process by lagging certain terms. The systems reduce to tridiagonal systems if any terms other than the tridiagonal terms are lagged (moved to the right-hand side of the system) along with the source terms. The solution process becomes more stable if part of the off-diagonals of the resulting tridiagonal system are also lagged. In the following algorithms, Ω is a relaxation factor, and Γ is a blending factor. Given initial estimates for \mathbf{k}^+ , \mathbf{h}^+ , \mathbf{u}^+ , $\mathbf{u}^{+''}$, \mathbf{k}^+ , and \mathbf{h}^+ , the following arrays can be calculated

$$\begin{aligned}
& \text{for } 1 \leq j \leq m; \quad v_j^+ = f_v; \quad u_j^{+'} = -R_r \hat{r}_j / (1 + v_j^+) \\
& u_1^{+''} = C_{D,1} u_1^{+'} + C_{U,1} u_2^{+'} + C_{U,2,1} u_3^{+'}; \quad k_1^{+'} = 0; \quad h_1^{+'} = 0; \quad v_1^{+'} = 0 \\
& \text{for } 2 \leq j \leq m-1; \quad u_j^{+''} = C_{L,1,j} u_{j-1}^{+'} + C_{D,j} u_j^{+'} + C_{U,1,j} u_{j+1}^{+'}; \quad k_j^{+'} = C_{L,1,j} k_{j-1}^{+'} + C_{D,j} k_j^{+'} + C_{U,1,j} k_{j+1}^{+'} \\
& \quad h_j^{+'} = C_{L,1,j} h_{j-1}^{+'} + C_{D,j} h_j^{+'} + C_{U,1,j} h_{j+1}^{+'}; \quad v_j^{+'} = C_{L,1,j} v_{j-1}^{+'} + C_{D,j} v_j^{+'} + C_{U,1,j} v_{j+1}^{+'} \\
& u_m^{+''} = C_{L,2,m} u_{m-2}^{+'} + C_{L,1,m} u_{m-1}^{+'} + C_{D,m} u_m^{+'}; \quad k_m^{+'} = 0 \\
& h_m^{+'} = C_{L,2,m} h_{m-2}^{+'} + C_{L,1,m} h_{m-1}^{+'} + C_{D,m} h_m^{+'}; \quad v_m^{+'} = C_{L,2,m} v_{m-2}^{+'} + C_{L,1,m} v_{m-1}^{+'} + C_{D,m} v_m^{+'}
\end{aligned} \tag{D.18}$$

Using the notation $T_{Lj} k_{j-1}^+ + T_{Dj} k_j^+ + T_{Uj} k_{j+1}^+ = B_j$ as row j in the tridiagonal system, the following algorithm can be used to obtain an improved estimate for \mathbf{k}^+

$$\begin{aligned}
& \text{for } 1 \leq j \leq m; \quad \text{if pipe flow; } P_j = (1 + v_j^+ / \sigma_k) / \hat{r}_j; \quad \text{else; } P_j = 0 \\
& T_{L1} = 0; \quad T_{D1} = C_{D,1}; \quad T_{U1} = C_{U,1}; \quad B_1 = -C_{U,2,1} k_3^+ \\
& T_{L2} = -\left(\frac{v_2^{+'}}{\sigma_k} + P_2 \right) C_{L,1,2} - \left(1 + \frac{v_2^+}{\sigma_k} \right) A_{L,1,2}; \quad T_{D2} = -\left(\frac{v_2^{+'}}{\sigma_k} + P_2 \right) C_{D,2} - \left(1 + \frac{v_2^+}{\sigma_k} \right) A_{D,2}; \\
& T_{U2} = -\left(\frac{v_2^{+'}}{\sigma_k} + P_2 \right) C_{U,1,2} - \left(1 + \frac{v_2^+}{\sigma_k} \right) A_{U,1,2} \\
& B_2 = S_k + \left(1 + \frac{v_2^+}{\sigma_k} \right) A_{U,2,2} k_4^+ \\
& \text{for } 3 \leq j \leq m-2 \\
& T_{Lj} = -\left(\frac{v_j^{+'}}{\sigma_k} + P_j \right) C_{L,1,j} - \left(1 + \frac{v_j^+}{\sigma_k} \right) A_{L,1,j}; \quad T_{Dj} = -\left(\frac{v_j^{+'}}{\sigma_k} + P_j \right) C_{D,j} - \left(1 + \frac{v_j^+}{\sigma_k} \right) A_{D,j}; \\
& T_{Uj} = -\left(\frac{v_j^{+'}}{\sigma_k} + P_j \right) C_{U,1,j} - \left(1 + \frac{v_j^+}{\sigma_k} \right) A_{U,1,j}; \\
& B_j = S_k + \left(1 + \frac{v_j^+}{\sigma_k} \right) A_{L,2,j} k_{j-2}^+ \\
& T_{Lm-1} = C_{L,2,m}; \quad T_{Dm-1} = C_{L,1,m}; \quad T_{Um-1} = C_{D,m}; \quad B_{m-1} = 0 \\
& T_{Lm} = 0; \quad T_{Dm} = 1; \quad T_{Um} = 0; \quad B_m = 0 \\
& \text{for } 2 \leq j \leq m-1; \quad T_{Lj} = \Gamma_k T_{Lj}; \quad T_{Uj} = \Gamma_k T_{Uj}; \quad B_j = B_j - (1 - \Gamma_k)(T_{Lj} k_{j-1}^+ + T_{Uj} k_{j+1}^+) \\
& \mathbf{k}_{\text{old}} = \mathbf{k}; \quad \mathbf{k} = [\mathbf{T}]^{-1} \mathbf{B}; \quad \mathbf{k} = \mathbf{k}_{\text{old}} + \Omega_k (\mathbf{k} - \mathbf{k}_{\text{old}})
\end{aligned} \tag{D.19}$$

The new estimate for \mathbf{k}^+ is used in Eq. (D.18) to update the arrays in that equation. A similar tridiagonal algorithm can then be used to evaluate an improved estimate for \mathbf{h}^+

$$\begin{aligned}
& \text{for } 1 \leq j \leq m; \text{ if pipe flow; } P_j = (1 + v_j^+ / \sigma_h) / \hat{r}_j; \text{ else; } P_j = 0 \\
& T_{L1} = 0; \quad T_{D1} = C_{D,1}; \quad T_{U1} = C_{U1,1}; \quad B_1 = -C_{U2,1} h_3^+ \\
& T_{L2} = -\left(\frac{v_2^{+'}}{\sigma_h} + P_2 \right) C_{L1,2} - \left(1 + \frac{v_2^+}{\sigma_h} \right) A_{L1,2}; \quad T_{D2} = -\left(\frac{v_2^{+'}}{\sigma_h} + P_2 \right) C_{D,2} - \left(1 + \frac{v_2^+}{\sigma_h} \right) A_{D,2}; \\
& T_{U2} = -\left(\frac{v_2^{+'}}{\sigma_h} + P_2 \right) C_{U1,2} - \left(1 + \frac{v_2^+}{\sigma_h} \right) A_{U1,2} \\
& B_2 = S_h + \left(1 + \frac{v_2^+}{\sigma_h} \right) A_{U2,2} h_4^+ \\
& \text{for } 3 \leq j \leq m - m_h \\
& T_{Lj} = -\left(\frac{v_j^{+'}}{\sigma_h} + P_j \right) C_{L1,j} - \left(1 + \frac{v_j^+}{\sigma_h} \right) A_{L1,j}; \quad T_{Dj} = -\left(\frac{v_j^{+'}}{\sigma_h} + P_j \right) C_{D,j} - \left(1 + \frac{v_j^+}{\sigma_h} \right) A_{D,j}; \\
& T_{Uj} = -\left(\frac{v_j^{+'}}{\sigma_h} + P_j \right) C_{U1,j} - \left(1 + \frac{v_j^+}{\sigma_h} \right) A_{U1,j}; \\
& B_j = S_h + \left(1 + \frac{v_j^+}{\sigma_h} \right) A_{L2,j} h_{j-2}^+ \\
& \text{for } m - m_h + 1 \leq j \leq m; \\
& T_{Lj} = 0; \quad T_{Dj} = 1; \quad T_{Uj} = 0; \quad B_j = f_{h0} \tag{D.20} \\
& \text{for } 2 \leq j \leq m - m_h; \quad T_{Lj} = \Gamma_h T_{Lj}; \quad T_{Uj} = \Gamma_h T_{Uj}; \quad B_j = B_j - (1 - \Gamma_h)(T_{Lj} h_{j-1}^+ + T_{Uj} h_{j+1}^+) \\
& \mathbf{h}_{\text{old}} = \mathbf{h}; \quad \mathbf{h} = [\mathbf{T}]^{-1} \mathbf{B}; \quad \mathbf{h} = \mathbf{h}_{\text{old}} + \Omega_h (\mathbf{h} - \mathbf{h}_{\text{old}})
\end{aligned}$$

The algorithms given in Eqs. (D.18)–(D.20) are repeated until a converged solution is obtained. Upon completion, \mathbf{u}^+ can be found by direct numerical integration using the trapezoidal rule

$$\begin{aligned}
& u_m^+ = 0 \\
& \text{for } m-1 \geq j \geq 1; \quad u_j^+ = u_{j+1}^+ + \frac{1}{2} (\hat{r}_j - \hat{r}_{j+1}) (u_j^{+'} + u_{j+1}^{+'}) \tag{D.21}
\end{aligned}$$

II. Higher-Order Formulations

A. Motivation

Second-order approximations are generally used in numerical methods to solve computational fluid dynamics problems. However, near a wall, second-order approximations are limited in their ability to capture higher-order phenomena. Many turbulence models are based on modeled turbulence parameters

that approach the wall as powers of y^2 or less. Therefore, second-order methods are often sufficient for capturing the model behavior. However, the model presented in this research includes higher-order phenomena near a wall. Therefore, higher-order methods must be used in this region. The following sections present the method used here for developing higher-order finite-differencing methods.

B. An n^{th} -Order First Derivative Approximation

The second-order approximation for the first derivative at a point can be obtained by retaining the first three terms in the Taylor series at two points near the point of interest, y_j . This gives

$$\begin{aligned}\phi_1 &\equiv \phi(y_j + \Delta y_1) = \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_1 + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_1^2}{2!} + O(\Delta y_1^3) + \dots \\ \phi_2 &\equiv \phi(y_j + \Delta y_2) = \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_2 + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_2^2}{2!} + O(\Delta y_2^3) + \dots\end{aligned}\quad (\text{D.22})$$

This system of equations can be rearranged to yield

$$\begin{Bmatrix} \left(\frac{d\phi}{dy}\right)_j \\ \left(\frac{d^2\phi}{dy^2}\right)_j \end{Bmatrix} = \begin{bmatrix} \Delta y_1 & \frac{\Delta y_1^2}{2} \\ \Delta y_2 & \frac{\Delta y_2^2}{2} \end{bmatrix}^{-1} \begin{Bmatrix} \phi_1 - \phi_j \\ \phi_2 - \phi_j \end{Bmatrix} = \frac{1}{\Delta y_1 \Delta y_2^2 - \Delta y_1^2 \Delta y_2} \begin{bmatrix} \Delta y_2^2 & -\Delta y_1^2 \\ -2\Delta y_2 & 2\Delta y_1 \end{bmatrix} \begin{Bmatrix} \phi_1 - \phi_j \\ \phi_2 - \phi_j \end{Bmatrix}\quad (\text{D.23})$$

This gives

$$\left(\frac{d\phi}{dy}\right)_j = \frac{\Delta y_2^2 \phi_1 + \phi_j (\Delta y_1^2 - \Delta y_2^2) - \Delta y_1^2 \phi_2}{\Delta y_1 \Delta y_2^2 - \Delta y_1^2 \Delta y_2}\quad (\text{D.24})$$

This can be used to obtain the second-order central difference formula for the first derivative on a uniform grid. Setting the y -distances in Eq. (D.24) as unit distances from the central node, gives $\Delta y_1 = -1\Delta y$ and $\Delta y_2 = 1\Delta y$. The function values are $\phi_1 = \phi_{j-1}$ and $\phi_2 = \phi_{j+1}$. Using this in Eq. (D.24) gives

$$\left(\frac{d\phi}{dy}\right)_j = \frac{\phi_{j+1} - \phi_{j-1}}{2\Delta y}\quad (\text{D.25})$$

This method can be used to obtain forward-difference and backward-difference formulas. For example, the forward-difference formula can be obtained from Eq. (D.24) by using $\Delta y_1 = 1\Delta y$ and $\Delta y_2 = 2\Delta y$ where the function values are $\phi_1 = \phi_{j+1}$, $\phi_2 = \phi_{j+2}$.

In general, the n^{th} -order approximation for the first derivative can be developed by retaining $n+1$ terms in the Taylor series at n points about the point of interest

$$\begin{aligned}
\phi_1 &\equiv \phi(y_j + \Delta y_1) = \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_1 + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_1^2}{2!} + \dots + \left(\frac{d^n\phi}{dy^n}\right)_j \frac{\Delta y_1^n}{n!} + O(\Delta y_1^{n+1}) + \dots \\
\phi_2 &\equiv \phi(y_j + \Delta y_2) = \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_2 + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_2^2}{2!} + \dots + \left(\frac{d^n\phi}{dy^n}\right)_j \frac{\Delta y_2^n}{n!} + O(\Delta y_2^{n+1}) + \dots \\
&\vdots \\
\phi_{n-1} &\equiv \phi(y_j + \Delta y_{n-1}) = \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_{n-1} + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_{n-1}^2}{2!} + \dots + \left(\frac{d^n\phi}{dy^n}\right)_j \frac{\Delta y_{n-1}^n}{n!} + O(\Delta y_{n-1}^{n+1}) + \dots \\
\phi_n &\equiv \phi(y_j + \Delta y_n) = \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_n + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_n^2}{2!} + \dots + \left(\frac{d^n\phi}{dy^n}\right)_j \frac{\Delta y_n^n}{n!} + O(\Delta y_n^{n+1}) + \dots
\end{aligned} \tag{D.26}$$

This system of equations can be rewritten as

$$\begin{pmatrix} \left(\frac{d\phi}{dy}\right)_j \\ \left(\frac{d^2\phi}{dy^2}\right)_j \\ \vdots \\ \left(\frac{d^{n-1}\phi}{dy^{n-1}}\right)_j \\ \left(\frac{d^n\phi}{dy^n}\right)_j \end{pmatrix} = \begin{bmatrix} \Delta y_1 & \frac{\Delta y_1^2}{2!} & \dots & \frac{\Delta y_1^{n-1}}{(n-1)!} & \frac{\Delta y_1^n}{n!} \\ \Delta y_2 & \frac{\Delta y_2^2}{2!} & \dots & \frac{\Delta y_2^{n-1}}{(n-1)!} & \frac{\Delta y_2^n}{n!} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \Delta y_{n-1} & \frac{\Delta y_{n-1}^2}{2!} & \dots & \frac{\Delta y_{n-1}^{n-1}}{(n-1)!} & \frac{\Delta y_{n-1}^n}{n!} \\ \Delta y_n & \frac{\Delta y_n^2}{2!} & \dots & \frac{\Delta y_n^{n-1}}{(n-1)!} & \frac{\Delta y_n^n}{n!} \end{bmatrix}^{-1} \begin{pmatrix} \phi_1 - \phi_j \\ \phi_2 - \phi_j \\ \vdots \\ \phi_{n-1} - \phi_j \\ \phi_n - \phi_j \end{pmatrix} \tag{D.27}$$

The solution to this system of equations gives the n^{th} -order approximation for the first derivative at a point. It should be noted that this method is not constrained to uniform grids. However, uniform grids are commonly used, and results are included in a subsequent section for 2nd-, 4th-, 6th-, and 8th-order approximations for uniform grid spacing.

C. An n^{th} -Order Second Derivative Approximation

The second-order finite-difference approximation for the second derivative at a point can be developed by considering the Taylor series expansion of the function about the point y_j . Given the location of three

discrete points at distances Δy_1 , Δy_2 , and Δy_3 , along with the values of the function at those points, ϕ_1 , ϕ_2 , and ϕ_3 , the system of equations can be written by retaining the first four terms in the Taylor series

$$\begin{aligned}\phi_1 &\equiv \phi(y_j + \Delta y_1) = \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_1 + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_1^2}{2!} + \left(\frac{d^3\phi}{dy^3}\right)_j \frac{\Delta y_1^3}{3!} + O(\Delta y_1^4) + \dots \\ \phi_2 &\equiv \phi(y_j + \Delta y_2) = \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_2 + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_2^2}{2!} + \left(\frac{d^3\phi}{dy^3}\right)_j \frac{\Delta y_2^3}{3!} + O(\Delta y_2^4) + \dots \\ \phi_3 &\equiv \phi(y_j + \Delta y_3) = \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_3 + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_3^2}{2!} + \left(\frac{d^3\phi}{dy^3}\right)_j \frac{\Delta y_3^3}{3!} + O(\Delta y_3^4) + \dots\end{aligned}\quad (\text{D.28})$$

This can be rearranged to yield

$$\begin{Bmatrix} \left(\frac{d\phi}{dy}\right)_j \\ \left(\frac{d^2\phi}{dy^2}\right)_j \\ \left(\frac{d^3\phi}{dy^3}\right)_j \end{Bmatrix} = \begin{bmatrix} \Delta y_1 & \frac{\Delta y_1^2}{2!} & \frac{\Delta y_1^3}{3!} \\ \Delta y_2 & \frac{\Delta y_2^2}{2!} & \frac{\Delta y_2^3}{3!} \\ \Delta y_3 & \frac{\Delta y_3^2}{2!} & \frac{\Delta y_3^3}{3!} \end{bmatrix}^{-1} \begin{Bmatrix} \phi_1 - \phi_j \\ \phi_2 - \phi_j \\ \phi_3 - \phi_j \end{Bmatrix}\quad (\text{D.29})$$

The solution of this system of equations gives the second-order finite difference approximations for the first and second derivatives of the function.

$$\begin{Bmatrix} \left(\frac{d\phi}{dy}\right)_j \\ \left(\frac{d^2\phi}{dy^2}\right)_j \\ \left(\frac{d^3\phi}{dy^3}\right)_j \end{Bmatrix} = \frac{\begin{bmatrix} \Delta y_2^2 \Delta y_3^2 (\Delta y_3 - \Delta y_2) & -\Delta y_1^2 \Delta y_3^2 (\Delta y_3 - \Delta y_1) & \Delta y_1^2 \Delta y_2^2 (\Delta y_2 - \Delta y_1) \\ -2\Delta y_2 \Delta y_3 (\Delta y_3^2 - \Delta y_2^2) & 2\Delta y_1 \Delta y_3 (\Delta y_3^2 - \Delta y_1^2) & -2\Delta y_1 \Delta y_2 (\Delta y_2^2 - \Delta y_1^2) \\ 6\Delta y_2 \Delta y_3 (\Delta y_3 - \Delta y_2) & -6\Delta y_1 \Delta y_3 (\Delta y_3 - \Delta y_1) & 6\Delta y_1 \Delta y_2 (\Delta y_2 - \Delta y_1) \end{bmatrix}}{\Delta y_1 \Delta y_2^2 \Delta y_3^2 (\Delta y_3 - \Delta y_2) - \Delta y_2 \Delta y_1^2 \Delta y_3^2 (\Delta y_3 - \Delta y_1) + \Delta y_3 \Delta y_1^2 \Delta y_2^2 (\Delta y_2 - \Delta y_1)} \begin{Bmatrix} \phi_1 - \phi_j \\ \phi_2 - \phi_j \\ \phi_3 - \phi_j \end{Bmatrix}\quad (\text{D.30})$$

This can be used to obtain the second-order central difference formula for the second derivative on a uniform grid. Setting the y -distances in Eq. (D.30) as distances from the central node, gives $\Delta y_1 = -1\Delta y$, $\Delta y_2 = 1\Delta y$, and $\Delta y_3 = 2\Delta y$, where the function values are $\phi_1 = \phi_{j-1}$, $\phi_2 = \phi_{j+1}$, and $\phi_3 = \phi_{j+2}$. Using this in Eq. (D.30) gives

$$\begin{Bmatrix} \left(\frac{d\phi}{dy}\right)_j \\ \left(\frac{d^2\phi}{dy^2}\right)_j \\ \left(\frac{d^3\phi}{dy^3}\right)_j \end{Bmatrix} = \begin{bmatrix} -1 & 1 & -1 \\ 3\Delta y & \Delta y & 6\Delta y \\ \Delta y^2 & \Delta y^2 & 0 \\ -1 & -3 & 1 \\ \Delta y^3 & \Delta y^3 & \Delta y^3 \end{bmatrix} \begin{Bmatrix} \phi_{j-1} - \phi_j \\ \phi_{j+1} - \phi_j \\ \phi_{j+2} - \phi_j \end{Bmatrix} \quad (\text{D.31})$$

Or, after simplifying,

$$\left(\frac{d^2\phi}{dy^2}\right)_j = \frac{\phi_{j-1} - 2\phi_j + \phi_{j+1}}{\Delta y^2} \quad (\text{D.32})$$

This method can be used to obtain forward-difference and backward-difference formulas. For example, the forward-difference formula can be obtained from Eq. (D.30) by using $\Delta y_1 = 1\Delta y$, $\Delta y_2 = 2\Delta y$, and $\Delta y_3 = 3\Delta y$, where the function values are $\phi_1 = \phi_{j+1}$, $\phi_2 = \phi_{j+2}$, and $\phi_3 = \phi_{j+3}$.

In general, the n^{th} -order approximation for the second derivative can be developed by retaining $n+2$ terms in the Taylor series at $n+1$ points.

$$\begin{aligned} \phi_1 \equiv \phi(y_j + \Delta y_1) &= \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_1 + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_1^2}{2!} + \dots + \left(\frac{d^{n+1}\phi}{dy^{n+1}}\right)_j \frac{\Delta y_1^{n+1}}{(n+1)!} + O(\Delta y_1^{n+2}) + \dots \\ \phi_2 \equiv \phi(y_j + \Delta y_2) &= \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_2 + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_2^2}{2!} + \dots + \left(\frac{d^{n+1}\phi}{dy^{n+1}}\right)_j \frac{\Delta y_2^{n+1}}{(n+1)!} + O(\Delta y_2^{n+2}) + \dots \\ &\vdots \\ \phi_n \equiv \phi(y_j + \Delta y_n) &= \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_n + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_n^2}{2!} + \dots + \left(\frac{d^{n+1}\phi}{dy^{n+1}}\right)_j \frac{\Delta y_n^{n+1}}{(n+1)!} + O(\Delta y_n^{n+2}) + \dots \\ \phi_{n+1} \equiv \phi(y_j + \Delta y_{n+1}) &= \phi_j + \left(\frac{d\phi}{dy}\right)_j \Delta y_{n+1} + \left(\frac{d^2\phi}{dy^2}\right)_j \frac{\Delta y_{n+1}^2}{2!} + \dots + \left(\frac{d^{n+1}\phi}{dy^{n+1}}\right)_j \frac{\Delta y_{n+1}^{n+1}}{(n+1)!} + O(\Delta y_{n+1}^{n+2}) + \dots \end{aligned} \quad (\text{D.33})$$

This system of equations can be rewritten as

$$\begin{pmatrix} \left(\frac{d\phi}{dy}\right)_j \\ \left(\frac{d^2\phi}{dy^2}\right)_j \\ \vdots \\ \left(\frac{d^n\phi}{dy^n}\right)_j \\ \left(\frac{d^{n+1}\phi}{dy^{n+1}}\right)_j \end{pmatrix} = \begin{bmatrix} \Delta y_1 & \frac{\Delta y_1^2}{2!} & \dots & \frac{\Delta y_1^n}{n!} & \frac{\Delta y_1^{n+1}}{(n+1)!} \\ \Delta y_2 & \frac{\Delta y_2^2}{2!} & \dots & \frac{\Delta y_2^n}{n!} & \frac{\Delta y_2^{n+1}}{(n+1)!} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \Delta y_n & \frac{\Delta y_n^2}{2!} & \dots & \frac{\Delta y_n^n}{n!} & \frac{\Delta y_n^{n+1}}{(n+1)!} \\ \Delta y_{n+1} & \frac{\Delta y_{n+1}^2}{2!} & \dots & \frac{\Delta y_{n+1}^n}{n!} & \frac{\Delta y_{n+1}^{n+1}}{(n+1)!} \end{bmatrix}^{-1} \begin{pmatrix} \phi_1 - \phi_j \\ \phi_2 - \phi_j \\ \vdots \\ \phi_n - \phi_j \\ \phi_{n+1} - \phi_j \end{pmatrix} \tag{D.34}$$

The solution to this system of equations gives the nth-order approximation for the second derivative at a point. Again, this method is not constrained to uniform grids. However, tabulated solutions for 2nd-, 4th-, 6th-, and 8th-order approximations are included in the next section.

D. Uniform-Grid Tabulated Approximations

Table D.1 Second-order approximation for the first derivative

	Forward Difference	Central Difference	Backward Difference
ϕ_{j+2}	-1		
ϕ_{j+1}	4	1	
ϕ_j	-3	0	3
ϕ_{j-1}		-1	-4
ϕ_{j-2}			1
Denominator	2Δy		

Table D.2 Second-order approximation for the second derivative

	Forward Difference	Central Difference	Backward Difference
ϕ_{j+3}	-1		
ϕ_{j+2}	4		
ϕ_{j+1}	-5	1	
ϕ_j	2	-2	2
ϕ_{j-1}		1	-5
ϕ_{j-2}			4
ϕ_{j-3}			-1
Denominator	Δy ²		

Table D.3 Fourth-order approximation for the first derivative

	Forward Difference		Central Difference	Backward Difference	
ϕ_{j+4}	-3				
ϕ_{j+3}	16	1			
ϕ_{j+2}	-36	-6	-1		
ϕ_{j+1}	48	18	8	3	
ϕ_j	-25	-10	0	10	25
ϕ_{j-1}		-3	-8	-18	-48
ϕ_{j-2}			1	6	36
ϕ_{j-3}				-1	-16
ϕ_{j-4}					3
Denominator	12 Δy				

Table D.4 Fourth-order approximation for the second derivative

	Forward Difference		Central Difference	Backward Difference	
ϕ_{j+5}	-10				
ϕ_{j+4}	61	1			
ϕ_{j+3}	-156	-6			
ϕ_{j+2}	214	14	-1		
ϕ_{j+1}	-154	-4	16	10	
ϕ_j	45	-15	-30	-15	45
ϕ_{j-1}		10	16	-4	-154
ϕ_{j-2}			-1	14	214
ϕ_{j-3}				-6	-156
ϕ_{j-4}				1	61
ϕ_{j-5}					-10
Denominator	12 Δy^2				

Table D.5 Sixth-order approximation for the first derivative

	Forward Difference			Central Difference	Backward Difference		
ϕ_{j+6}	-10						
ϕ_{j+5}	72	2					
ϕ_{j+4}	-225	-15	-1				
ϕ_{j+3}	400	50	8	1			
ϕ_{j+2}	-450	-100	-30	-9	-2		
ϕ_{j+1}	360	150	80	45	24	10	
ϕ_j	-147	-77	-35	0	35	77	147
ϕ_{j-1}		-10	-24	-45	-80	-150	-360
ϕ_{j-2}			2	9	30	100	450
ϕ_{j-3}				-1	-8	-50	-400
ϕ_{j-4}					1	15	225
ϕ_{j-5}						-2	-72
ϕ_{j-6}							10
Denominator	60 Δy						

Table D.6 Sixth-order approximation for the second derivative

	Forward Difference			Central Difference	Backward Difference		
ϕ_{j+7}	-126						
ϕ_{j+6}	1019	11					
ϕ_{j+5}	-3618	-90	-2				
ϕ_{j+4}	7380	324	16				
ϕ_{j+3}	-9490	-670	-54	2			
ϕ_{j+2}	7911	855	85	-27	-11		
ϕ_{j+1}	-4014	-486	130	270	214	126	
ϕ_j	938	-70	-378	-490	-378	-70	938
ϕ_{j-1}		126	214	270	130	-486	-4014
ϕ_{j-2}			-11	-27	85	855	7911
ϕ_{j-3}				2	-54	-670	-9490
ϕ_{j-4}					16	324	7380
ϕ_{j-5}					-2	-90	-3618
ϕ_{j-6}						11	1019
ϕ_{j-7}							-126
Denominator	180 Δy^2						

Table D.7 Eighth-order approximation for the first derivative

	Forward Difference				Central Difference	Backward Difference			
ϕ_{j+8}	-105								
ϕ_{j+7}	960	15							
ϕ_{j+6}	-3920	-140	-5						
ϕ_{j+5}	9408	588	48	3					
ϕ_{j+4}	-14700	-1470	-210	-30	-3				
ϕ_{j+3}	15680	2450	560	140	32	5			
ϕ_{j+2}	-11760	-2940	-1050	-420	-168	-60	-15		
ϕ_{j+1}	6720	2940	1680	1050	672	420	240	105	
ϕ_j	-2283	-1338	-798	-378	0	378	798	1338	2283
ϕ_{j-1}		-105	-240	-420	-672	-1050	-1680	-2940	-6720
ϕ_{j-2}			15	60	168	420	1050	2940	11760
ϕ_{j-3}				-5	-32	-140	-560	-2450	-15680
ϕ_{j-4}					3	30	210	1470	14700
ϕ_{j-5}						-3	-48	-588	-9408
ϕ_{j-6}							5	140	3920
ϕ_{j-7}								-15	-960
ϕ_{j-8}									105
Denominator	840 Δy								

Table D.8 Eighth-order approximation for the second derivative

	Forward Difference				Central Difference	Backward Difference			
ϕ_{j+9}	-3044								
ϕ_{j+8}	30663	223							
ϕ_{j+7}	-139248	-2268	-38						
ϕ_{j+6}	375704	10424	389	9					
ϕ_{j+5}	-667800	-28560	-1800	-90					
ϕ_{j+4}	818874	51786	4956	396	-9				
ϕ_{j+3}	-704368	-65128	-8932	-952	128	38			
ϕ_{j+2}	422568	57288	10458	882	-1008	-603	-223		
ϕ_{j+1}	-165924	-28944	-2184	5796	8064	6984	5274	3044	
ϕ_j	32575	2135	-7900	-12460	-14350	-12460	-7900	2135	32575
ϕ_{j-1}		3044	5274	6984	8064	5796	-2184	-28944	-165924
ϕ_{j-2}			-223	-603	-1008	882	10458	57288	422568
ϕ_{j-3}				38	128	-952	-8932	-65128	-704368
ϕ_{j-4}					-9	396	4956	51786	818874
ϕ_{j-5}						-90	-1800	-28560	-667800
ϕ_{j-6}						9	389	10424	375704
ϕ_{j-7}							-38	-2268	-139248
ϕ_{j-8}								223	30663
ϕ_{j-9}									-3044
Denominator	$5040\Delta y^2$								

E. Example Results

To illustrate the effects of the order of the approximation, consider using a finite-difference algorithm to solve the differential equation

$$\frac{d^2\phi}{dy^2} = a(a-1)y^{a-2} - a^2y^{a-1} \quad (\text{D.35})$$

where a is an arbitrary constant and the boundary conditions are

$$\phi(0) = 0, \quad \frac{d\phi}{dy}(1) = 0 \quad (\text{D.36})$$

This can be solved analytically to give

$$\phi = y^a - \frac{a}{a+1} y^{a+1} \quad (\text{D.37})$$

Choosing $a = 2$ gives a third-order solution. However, as $y \rightarrow 0$, the solution approaches second-order. The solution for this case is easily solved using a second-order numerical method. However, choosing $a = 3$ gives a fourth-order solution that approaches third-order as $y \rightarrow 0$. A second-order numerical method cannot resolve the near-wall solution to this equation because the approximation truncates higher-order terms. Therefore, the truncation error of this method is too great to resolve cases of Eq. (D.35) where $a > 2$. Solutions to this equation for $2 \leq a \leq 6$ using a second-order finite-difference method are shown in Fig. D.1 using double-precision computations on a uniform grid with 1601 nodes. Note the evidence of truncation error in the higher-order solutions.

The turbulence model by Phillips suggests that k approaches the wall as y^6 . Therefore, the ability to capture the solution to Eq. (D.35) with $a = 6$ is of interest. Figures D.2–D.7 demonstrate the limitations of using lower-order differencing methods to approximate the equation

$$u = y^6 - \frac{6}{7} y^7 \quad (\text{D.38})$$

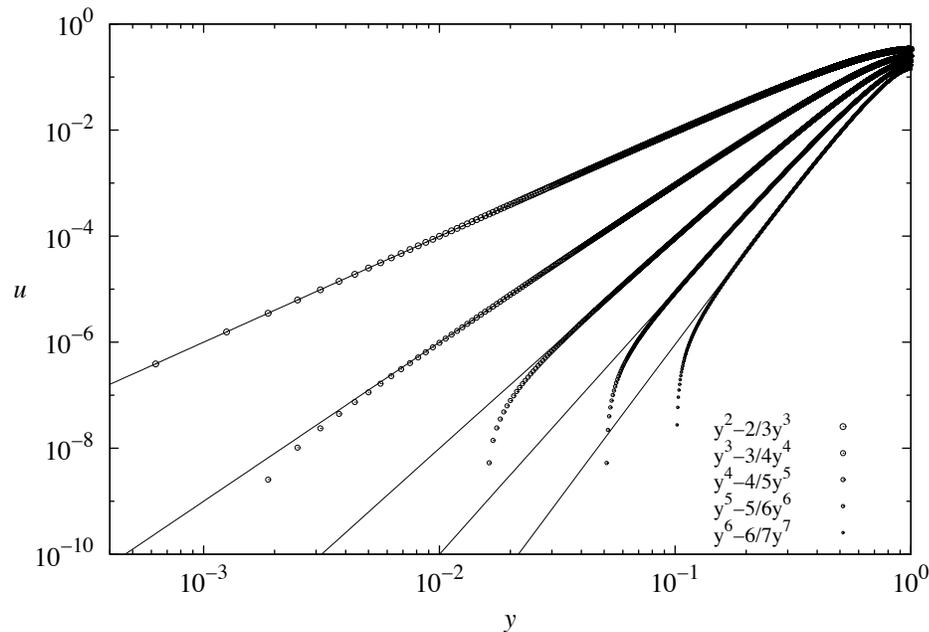


Fig. D.1 Near-wall results of a second-order finite-difference algorithm.

Figure D.2 shows the results of the second-order finite-difference solver with several grid densities using double-precision computations. Figure D.3 shows the same calculations using quad-precision computations. Note that regardless of the grid density, the first few points from the wall do not follow the 6th-order near-wall solution. This is true whether double- or quad-precision computations are used.

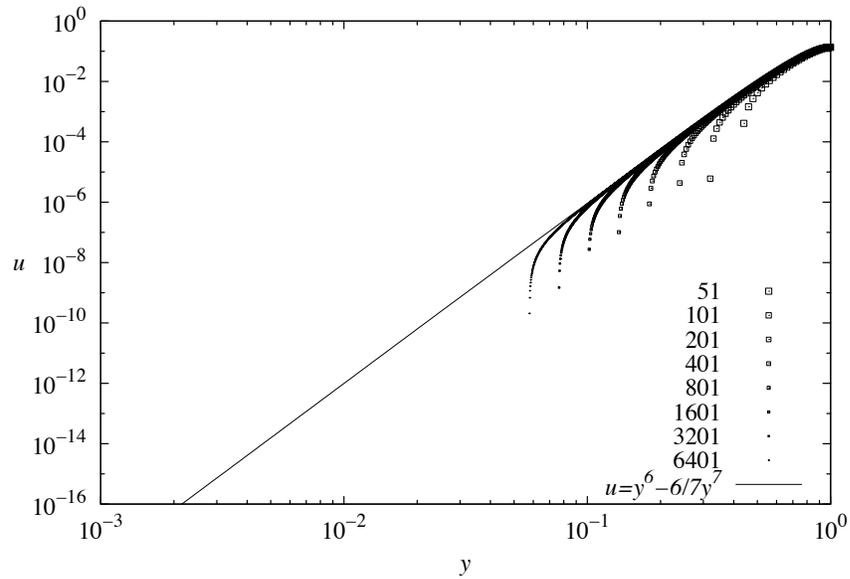


Fig. D.2 Second-order results using double-precision computations.

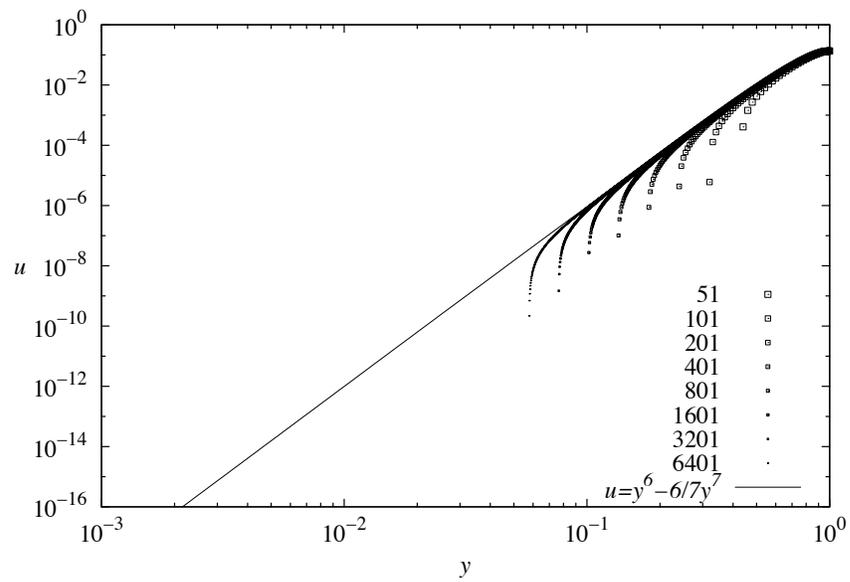


Fig. D.3 Second-order results using quad-precision computations.

Figure D.4 shows the results of the fourth-order finite-difference solver using double-precision computations, and Fig. D.5 shows the same computations as Fig. D.4, but with quad-precision. Again, note that the first few nodes nearest the wall do not follow the correct solution regardless of the grid density.

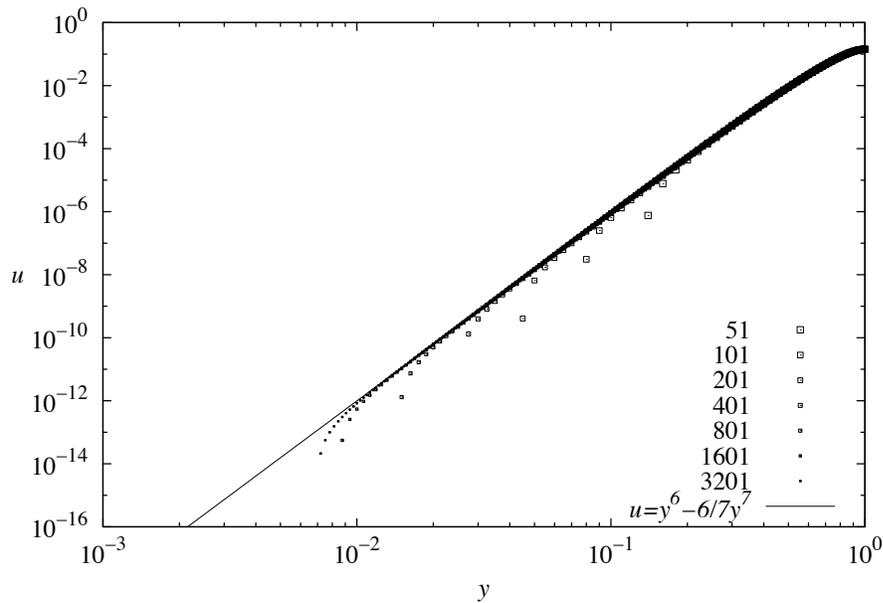


Fig. D.4 Fourth-order results using double-precision computations.

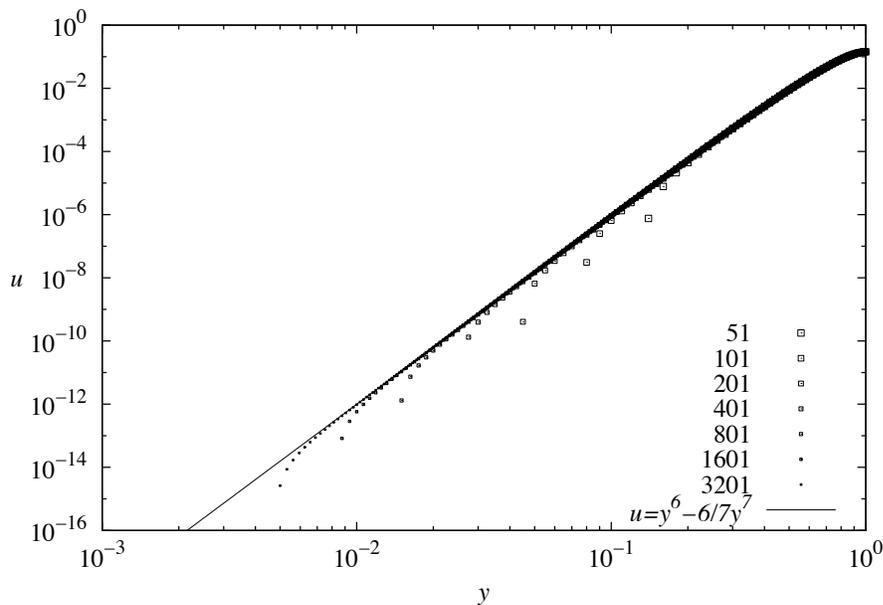


Fig. D.5 Fourth-order results using quad-precision computations.

Figure D.6 shows results using the eighth-order finite-difference solver with double-precision computations, and Fig. D.7 shows these same calculations using quad-precision computations. Here, the discrepancies seen in the finest grid of Fig. D.6 are no longer truncation errors, but round-off errors. This is shown by comparing the results of Figure 6 to those given in Fig. D.7 where the same computations were made using quad precision.

Figure D.8 shows the results of using the eighth-order method to approximate the equation

$$u = y^{10} - \frac{10}{11}y^{11} \quad (\text{D.39})$$

with varying grid resolutions using logarithmic grid clustering. Note that these results look very similar to the results of using a second-order method to approximate a higher-order equation. To demonstrate that the method used to obtain Fig. D.8 is in fact eighth-order, the RMS error of the solution is plotted as a function of Δy^8 in Fig. D.9. A line passing through the data points obtained from the two finest grids is superimposed on the plot. The fact that the error approaches this line as $\Delta y \rightarrow 0$ shows that the solution method is eighth-order.

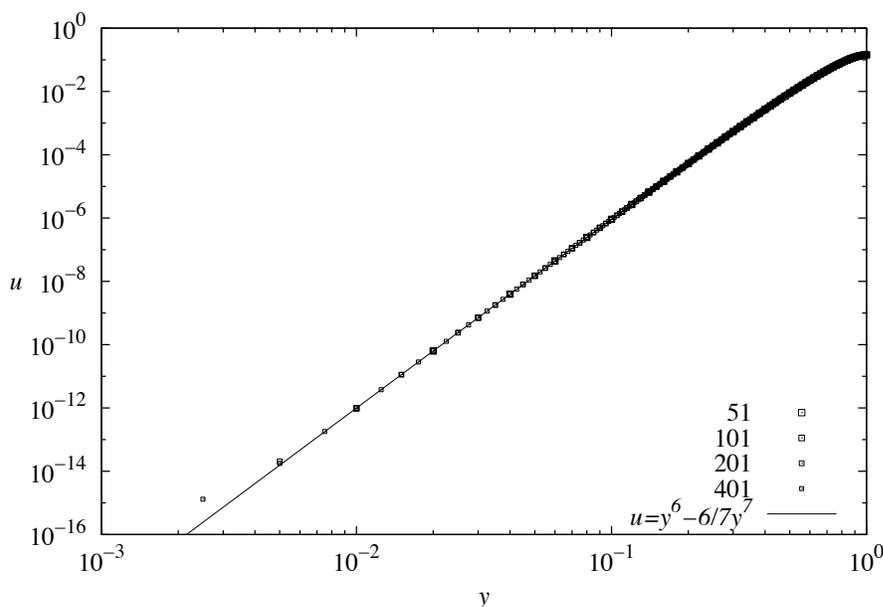


Fig. D.6 Eighth-order results using double-precision computations.

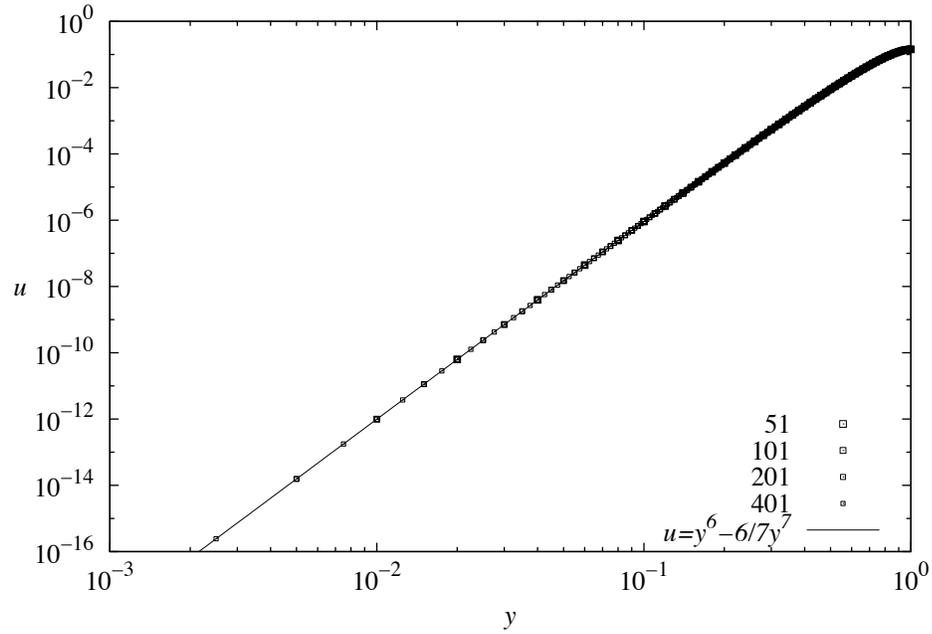


Fig. D.7 Eighth-order results using quad-precision computations.

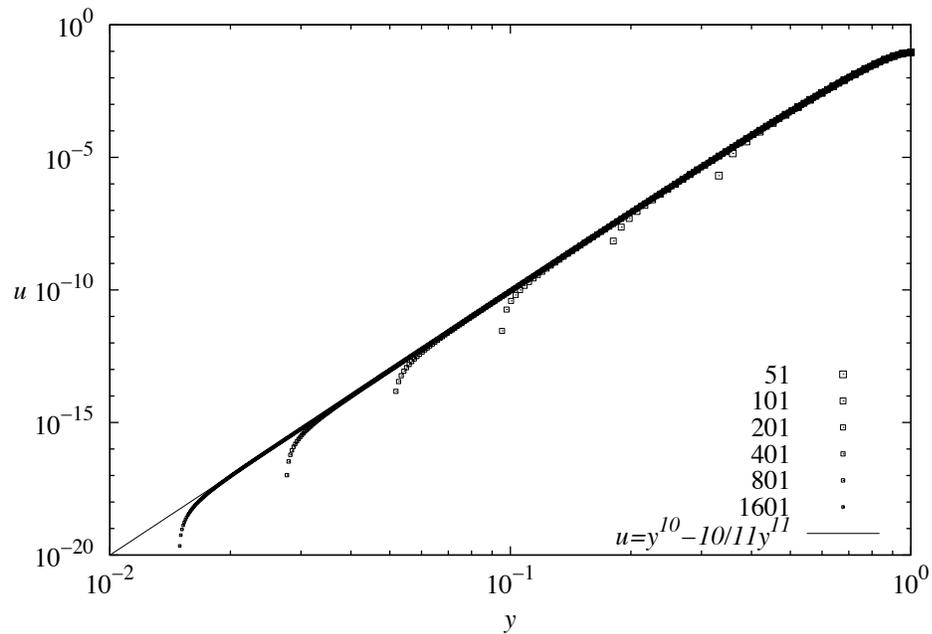


Fig. D.8 Eighth-order results of a tenth-order near-wall equation using quad-precision computations.

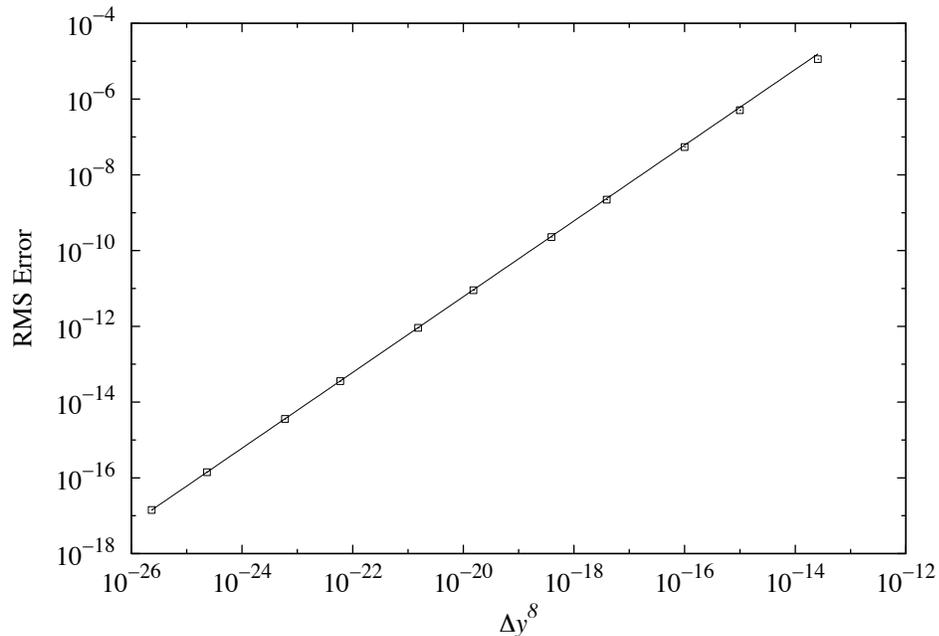


Fig. D.9 Eighth-order finite-difference RMS error as a function of grid spacing.

As a final note on the difficulties associated with using a lower-order method to estimate a higher-order solution, consider again the results shown in Fig D.8. Many of the most important approximations in fluid mechanics are based on the behavior of the data nearest the wall. The data points nearest the wall from any of the data sets in Fig. D.8 are actually negative, and therefore do not appear on the logarithmic plot shown in Fig. D.8. The fact that the near-wall solution is negative can be very troublesome because it appears that if the order of the solution near the wall is greater than the order of the numerical approximations, the correct sign of the solution nearest the wall can't even be obtained. In order to observe the behavior of the near-wall data obtained from the scenario shown in Fig. D.8, the near-wall data is plotted in Fig. D.10 on a logarithmic plot with a negative y -axis. Note that the error of the first node off of the wall still converges in an eighth-order fashion. However, it appears that regardless of how fine the grid is, the solution very near the wall will always have the wrong sign. For this reason it is important to ensure that the numerical method is always higher order than the order of the solution very near a wall.

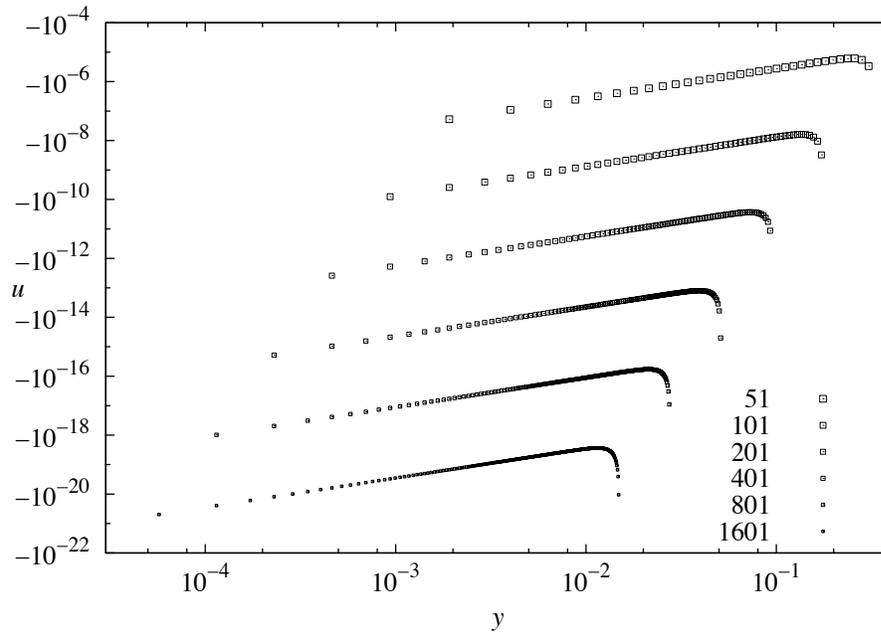


Fig. D.10 Negative near-wall results of an eighth-order approximation to a tenth-order near-wall solution.

III. Sample Code

The following Fortran code can be used to solve the Wilcox 1998 turbulence model for smooth walls using an arbitrary order of finite differencing. The variable “approx” in the main code is used to set the finite difference order. Subroutines for basic matrix operations such as inverting a matrix or solving a system of equations in matrix form have not been included.

```

PROGRAM Wilcox1998
IMPLICIT NONE
INTEGER::m,mh,mk,j,iter,itermax,limit_nu, limit_k,
limit_h, filesize, line, coord, ierror, tiTer, iprint, cont, eligiblepts, approx
REAL::Rtau, beta, u_bulk, Re, Cf, cbeta, fD, eta, fitness, dy, dzeta
REAL::gammak, omegak, sigmak, rmsk
REAL::gammah, omegah, sigmah, rmsh
REAL::fnu, Sk, Sh, fk0, fh0 !Function Subroutines
REAL::yval, uval, kval, hval, nuval, yvalold, uvalold, kvalold, hvalold, nuvalold, frac !File vars
REAL, allocatable, dimension(:)::y, B, nu, nuprime, zeta
REAL, allocatable, dimension(:)::k, k_old, kprime, h, h_old, hprime, u, uprime, P, udprime, kdprime
REAL, allocatable, dimension(:)::const, VDu, VDconst
REAL, ALLOCATABLE, DIMENSION(:,)::A, FD1, FD2
CHARACTER*(20):: rec, init
CHARACTER(LEN=100)::fn

ALLOCATE(const(4));
ALLOCATE(VDconst(2));

!----- Set Defaults -----
m = 51 ! Number of nodes
Rtau = 300.0 ! Length
mh = 7 ! Number of nodes to use asymptotic solution for h

```

```

mk = 1                ! Number of nodes to use asymptotic solution for k

approx = 6

const(1) = 0.52       !C_omega1
const(2) = 0.072      !C_omega2
const(3) = 0.0        !not used
const(4) = 0.09       !C_nu
sigmak = 2.0          !sigma_k
sigmah = 2.0          !sigma_omega

VDconst(1) = 0.41     !kappa
VDconst(2) = 26.0     !Aplus

gammak = 0.99         ! Blending Factor
gammah = 0.99         ! Blending Factor
omegak = 0.05         ! Relaxation Factor
omegah = 0.05         ! Relaxation Factor
itermax = 100
iprint = 1
beta = 1.02
init = 'none'
coord = 0              ! = 1 if pipe, =0 if channel
cont = 0               ! =1 if continue until convergence is reached is selected

200 FORMAT (1X, 1000I10)
100 FORMAT (1X, 1000ES20.12)
110 FORMAT (1X, I10, 4ES20.12, 3I10)

!----- User Inputs -----
write(*,*) 'Wilcox 1998'
write(*,*) 'Channel flow or pipe flow? (0=channel, 1=pipe) (' ,coord, '):'
read(5,'(a)') rec
if(rec .ne. ' ') then
    read(rec,*) coord
end if

write(*,*) 'Finite-Difference order (' ,approx, '):'
read(5,'(a)') rec
if(rec .ne. ' ') then
    read(rec,*) approx
end if

write(*,*) 'Number of nodes (' ,m, '):'
read(5,'(a)') rec
if(rec .ne. ' ') then
    read(rec,*) m
end if

write(*,*) 'R_tau (' ,Rtau, '):'
read(5,'(a)') rec
if(rec .ne. ' ') then
    read(rec,*) Rtau
end if

write(*,*) 'Stretching Factor (' ,beta, '):'
read(5,'(a)') rec
if(rec .ne. ' ') then
    read(rec,*) beta
end if

write(*,*) 'Enter C_omega1 (' ,const(1), '):'
read(5,'(a)') rec
if(rec .ne. ' ') then
    read(rec,*) const(1)
end if

write(*,*) 'Enter C_omega2 (' ,const(2), '):'
read(5,'(a)') rec
if(rec .ne. ' ') then
    read(rec,*) const(2)
end if

write(*,*) 'Enter C_omega3 (' ,const(3), '):'
read(5,'(a)') rec
if(rec .ne. ' ') then
    read(rec,*) const(3)
end if

```

```

write(*,*) 'Enter C_nu (' ,const(4), ' ):'
read(5,'(a)') rec
if(rec .ne. ' ') then
  read(rec,*) const(4)
end if

write(*,*) 'Enter sigma_k (' ,sigmak, ' ):'
read(5,'(a)') rec
if(rec .ne. ' ') then
  read(rec,*) sigmak
end if

write(*,*) 'Enter sigma_omega (' ,sigmah, ' ):'
read(5,'(a)') rec
if(rec .ne. ' ') then
  read(rec,*) sigmah
end if

write(*,*) 'Enter mh (' ,mh, ' ):'
read(5,'(a)') rec
if(rec .ne. ' ') then
  read(rec,*) mh
end if

write(*,*) 'Enter mk (' ,mk, ' ):'
read(5,'(a)') rec
if(rec .ne. ' ') then
  read(rec,*) mk
end if

write(*,*) 'Initialize from a file (enter [none] for automated guess) (' ,init, '):'
read(5,'(a)') rec
if(rec .ne. ' ') then
  read(rec,*) init
end if

!Allocate Memory
ALLOCATE(y(m)); ALLOCATE(B(m));
ALLOCATE(k(m)); ALLOCATE(k_old(m)); ALLOCATE(kprime(m)); ALLOCATE(kdprime(m));
ALLOCATE(h(m)); ALLOCATE(h_old(m)); ALLOCATE(hprime(m));
ALLOCATE(u(m)); ALLOCATE(uprime(m)); ALLOCATE(udprime(m));
ALLOCATE(nu(m)); ALLOCATE(nuprime(m)); ALLOCATE(zeta(m)); ALLOCATE(P(m));
ALLOCATE(VDu(m));
ALLOCATE(A(m,m));ALLOCATE(FD1(m,m));ALLOCATE(FD2(m,m))

dzeta = 1.0/real(m-1)
dy = Rtau*dzeta
!Create Grid
do j=1,m,1
  eta = real(j-1)/real(m-1)
  cbeta = ((beta+1.0)/(beta-1.0))**(1.0-eta)
  if(beta .eq. 0) then
    zeta(m+1-j) = 1.0 - real(j-1)*dzeta
  else
    zeta(m+1-j) = 1.0 - (beta+1.0 - (beta-1.0)*cbeta)/(1.0 + cbeta);
  end if
  y(m+1-j) = Rtau*(1.0-zeta(m+1-j))
  k(m+1-j) = 0.01*y(m+1-j)!fk0(m+1-
j,m,Rtau,y,k,h,nu,uprime,udprime,kprime,kdprime,hprime,const,coord)
  h(m+1-j) = fh0(m+1-
j,m,Rtau,y,k,h,nu,uprime,udprime,kprime,kdprime,hprime,const,coord)
end do

CALL FDMatrix(approx,m,zeta,1,FD1)
CALL FDMatrix(approx,m,zeta,2,FD2)

call VanDriestSolution(m,y,VDconst,VDu)

if(init .ne. 'none') then
  open(1,FILE=init)
  read(1,*) filesize
  read(1,*)
  read(1,*)
  read(1,*)
  read(1,*)yvalold,uvalold,kvalold,hvalold,nuvalold
  u(m) = uvalold; k(m) = kvalold; h(m) = hvalold; nu(m) = nuvalold;
  do line=2,filesize,1
    read(1,*)yval,uval,kval,hval,nuval
    do j=2,m-1,1

```

```

        if((y(m+1-j) < yval) .and. (y(m+1-j) > yvalold)) then
            frac = (y(m+1-j) - yvalold)/(yval-yvalold)
            u(m+1-j) = frac*(uval-uvalold)+uvalold
            k(m+1-j) = frac*(kval-kvalold)+kvalold
            h(m+1-j) = frac*(hval-hvalold)+hvalold
            nu(m+1-j) = frac*(nuval-nuvalold)+nuvalold
        end if
    end do
    yvalold = yval; uvalold = uval; kvalold = kval; hvalold = hval; nuvalold = nuval;
end do
u(1) = uval; k(1) = kval; h(1) = hval; nu(1) = nuval;
do j=1,m,1
    uprime(j) = -Rtau*zeta(j)/(1.0+nu(j));
end do
do j=m-mh+1,m
    h(j)=fh0(j,m,Rtau,y,k,h,nu,uprime,udprime,kprime,kdprime,hprime,const,coord)
end do
end if

write(*,*) 'Number of iterations before pause (' ,itermax, ' ):'
write(*,*) '   Enter zero to exit'
write(*,*) '   Enter -1 to run to convergence'
read(5,'(a)') rec
if(rec .ne. ' ') then
    read(rec,*) itermax
end if
if(itermax<0) then
    cont = 1; itermax = 1;
end if
titer=0 !total iterations
do while (itermax>0)
    write(*,*) 'Number of iterations per print (' ,iprint, ' ):'
    read(5,'(a)') rec
    if(rec .ne. ' ') then
        read(rec,*) iprint
    end if
    write(*,*) 'Blending Factor (' ,gammak, ' ):'
    read(5,'(a)') rec
    if(rec .ne. ' ') then
        read(rec,*) gammak
    end if
    gammah = gammak
    write(*,*) 'Relaxation Factor (' ,omegak, ' ):'
    read(5,'(a)') rec
    if(rec .ne. ' ') then
        read(rec,*) omegak
    end if
    omegah = omegak

    iter = 0
    write(*,*) 'iterations   k_rms           h_rms           centerline k           &
           &centerline h           limit_k   limit_h   limit_nu'
    do while (iter < itermax)
        limit_nu = 0; limit_k = 0; limit_h = 0;
        rmsk = 0.0; rmsksh = 0.0;

        !Update Arrays
        do j=1,m,1
            nu(j) = fnu(j,m,y,k,h,uprime,udprime,kprime,kdprime,hprime,const,coord);
            if(nu(j) > 1e5) then
                nu(j) = 1e5; limit_nu = limit_nu+1;
            end if
            uprime(j) = -Rtau*zeta(j)/(1.0+nu(j));
        end do

        call Derivative(m,FD1,nu,nuprime)
        call Derivative(m,FD1,uprime,udprime)
        call Derivative(m,FD1,k,kprime)
        call Derivative(m,FD2,k,kdprime)
        call Derivative(m,FD1,h,hprime)
        kprime(1)=0.0; hprime(1)=0.0; nuprime(1)=0.0; kprime(m)=0.0;
        !k solver
        if(coord .eq. 1) then
            do j=1,m,1
                P(j) = (1.0+nu(j)/sigmak)/zeta(j);
            end do
        else
            P=0.0;
        end if
    end while
end if

```

```

A(1,:) = FD1(1,:); B(1) = 0.0 !centerline boundary condition
do j=2,m-2
  A(j,:) = -(nuprime(j)/sigmak+P(j))*FD1(j,:) - (1.0 + nu(j)/sigmak)*FD2(j,:)
  B(j)=Sk(j,m,Rtau,y,k,h,nu,uprime,udprime,kprime,kdprime,hprime,const,coord)
end do
A(m-1,:) = FD1(m,:); B(m-1) = 0.0 !kprime=0 at wall
A(m,:) = 0.0; A(m,m) = 1.0; B(m) = 0.0 !k=0 at wall

k_old = k;
CALL MatrixBlend(m,gammak,A,k,B)
CALL AXB(m,A,B,k);

k = k_old + omegak*(k - k_old);
do j=1,m,1 !Usual limiter on k
  if(k(j) < 0.0) then
    k(j) = 1e-30;
    limit_k = limit_k + 1;
  end if
end do

!Update Arrays
do j=1,m,1
  nu(j) = fnu(j,m,y,k,h,uprime,udprime,kprime,kdprime,hprime,const,coord);
  if(nu(j) > 1e5) then
    nu(j) = 1e5; limit_nu = limit_nu+1;
  end if
  uprime(j) = -Rtau*zeta(j)/(1.0+nu(j));
end do
call Derivative(m,FD1,nu,nuprime)
call Derivative(m,FD1,uprime,udprime)
call Derivative(m,FD1,k,kprime)
call Derivative(m,FD2,k,kdprime)
call Derivative(m,FD1,h,hprime)
kprime(1)=0.0; hprime(1)=0.0; nuprime(1)=0.0; kprime(m)=0.0;

!h solver
if(coord .eq. 1) then
  do j=1,m,1
    P(j) = (1.0+nu(j)/sigmah)/zeta(j);
  end do
else
  P=0.0;
end if
A(1,:) = FD1(1,:); B(1) = 0.0 !centerline boundary condition
do j=2,m-mh
  A(j,:) = -(nuprime(j)/sigmah+P(j))*FD1(j,:) - (1.0 + nu(j)/sigmah)*FD2(j,:)
  B(j)=Sh(j,m,Rtau,y,k,h,nu,uprime,udprime,kprime,kdprime,hprime,const,coord)
end do
do j=m-mh+1,m
  A(j,:) = 0.0; A(j,j) = 1.0;
B(j)=fh0(j,m,Rtau,y,k,h,nu,uprime,udprime,kprime,kdprime,hprime,const,coord) !near-wall
end do

h_old = h;
CALL MatrixBlend(m,gammah,A,h,B)
CALL AXB(m,A,B,h);

h = h_old + omegah*(h - h_old);
do j=1,m-mh,1 !usual limiter on h
  if(h(j) < 0.0) then
    h(j) = 1e-30;
    limit_h = limit_h + 1;
  end if
end do

do j=1,m,1
  rmsk = rmsk + (k(j) - k_old(j))**2
end do
do j=1,m-mh,1
  rmsh = rmsh + (h(j) - h_old(j))**2
end do
rmsk = sqrt(rmsk/m); rmsh = sqrt(rmsh/m);
if(rmsk .ne. rmsk) exit; if(rmsh .ne. rmsh) exit;
if((rmsk < 1.0e-18) .and. (rmsh < 1.0e-18)) then
  write(*,110) titer,rmsk,rmsh,k(1),h(1),limit_k,limit_h,limit_nu; exit;
end if
iter = iter + 1; titer = titer+1;
if(cont .eq. 1) itermax = itermax + 1
if(mod(real(iter),real(iprint)).eq.0.0) then

```

```

        write(fn,*) iter
        fn = trim(adjustl(fn))//'.txt'
        open(unit = 10, File = "results.txt", status="replace", action = "write",
iostat = ierror)
        write(10,*) m,' nodes'
        write(10,*) '      y_plus          u_prime          k_plus
h_plus      nu_plus'
        do j=1,m
            write(10,100) y(m+1-j),uprime(m+1-j),k(m+1-j),h(m+1-j),nu(m+1-j)
        end do
        close(10)
        write(*,110) titer,rmsk,rmsk,h(1),h(1),limit_k,limit_h,limit_nu
    end if
end do

print *, char(7) !Makes a noise
if(cont.eq.1) exit;
write(*,*) 'Number of iterations before pause (' ,itermax, ' ):'
write(*,*) ' Enter zero to exit'
write(*,*) ' Enter -1 to run to convergence'
read(5,'(a)') rec
if(rec .ne. ' ') then
    read(rec,*) itermax
end if
if(itermax<0) then
    cont = 1; itermax = 1;
end if
end do

!integrate u
u(m) = 0.0;
do j=2,m
    u(m+1-j) = u(m-j+2) + 0.5*(zeta(m+1-j)-zeta(m-j+2))*(uprime(m+1-j) + uprime(m-j+2))
!Physical Domain
end do

if(coord .eq. 1) then !Pipe
    write(*,*) '----- Pipe Flow Results -----'
    u_bulk = 0.0;
    do j=2,m
        u_bulk = u_bulk + (zeta(j)-zeta(j-1))*(zeta(j)*u(j) + zeta(j-1)*u(j-1))
    end do
    Re = 2.0*u_bulk*Rtau
else !Channel flow
    write(*,*) '----- Channel Flow Results -----'
    u_bulk = 0.0;
    do j=2,m
        u_bulk = u_bulk + 0.5*(zeta(j)-zeta(j-1))*(u(j) + u(j-1))
    end do
    Re = 4.0*u_bulk*Rtau
end if
Cf = 2.0/u_bulk**2
fD = 4.0*Cf

!Calculate fitness against Van Driest Solution
fitness = 0.0; eligiblepts = 0;
do j=1,m
    if(y(m+1-j) < 500) then
        fitness = fitness + (u(m+1-j) - VD_u(m+1-j))**2
        eligiblepts = eligiblepts+1
    end if
end do
fitness = sqrt(fitness/eligiblepts)

write(*,*) '      R_tau = ',Rtau
write(*,*) '      nodes = ',m
write(*,*) '      u+_bulk = ',u_bulk
write(*,*) '      Reynolds number, Re = ',Re
write(*,*) ' Fanning friction factor, Cf = ',Cf
write(*,*) ' Darcy friction factor, f_D = ',fD
write(*,*) ' Fitness against Van Driest = ',fitness
write(*,*) '-----'

open(unit = 10, File = "results.txt", status="replace", action = "write", iostat =
ierror)
write(10,*) m,' nodes'
write(10,*) fitness,' fitness against Van Driest'
write(10,*) ' nodes=',m,' Cnu=',const(4),' C1=',const(1),' C2=',const(2),' C3=',const(3),'
sigmak=',sigmak,' sigmaw=',sigmah

```

```

write(10,*) '   y_plus           u_plus           k_plus           h_plus
nu_plus           Van Driest'
do j=1,m
  write(10,100) y(m+1-j),u(m+1-j),k(m+1-j),h(m+1-j),nu(m+1-j),VDu(m+1-j)
end do
close(10)
write(*,*)
write(*,*) 'Results written to results.txt in local directory.'
write(*,*)

!Deallocate Memory
DEALLOCATE(y); DEALLOCATE(B); DEALLOCATE(k)
DEALLOCATE(k_old); DEALLOCATE(h); DEALLOCATE(h_old); DEALLOCATE(nu)
STOP
END PROGRAM Wilcox1998

!----- FUNCTIONS -----
REAL FUNCTION fnu(j,m,y,k,h,uprime,udprime,kprime,kdprime,hprime,const,coord)
IMPLICIT NONE; INTEGER::j,m,coord;
REAL,DIMENSION(m)::y,k,h,nu,uprime,udprime,kprime,kdprime,hprime; REAL,DIMENSION(4)::const;
Real::Rt
Rt = k(j)/h(j)
if(j.eq.m) then
  fnu = 0.0
else
  fnu = Rt*(0.024 + Rt/6.0)/(1.0 + Rt/6.0)
end if
RETURN; END FUNCTION fnu

REAL FUNCTION Sk(j,m,Rtau,y,k,h,nu,uprime,udprime,kprime,kdprime,hprime,const,coord)
IMPLICIT NONE; INTEGER::j,m,coord;
REAL,DIMENSION(m)::y,k,h,nu,uprime,udprime,kprime,kdprime,hprime;
REAL,DIMENSION(4)::const;
Real::Rtau,r,Rt,chi
Rt = k(j)/h(j)
chi = kprime(j)*hprime(j)/(h(j)**3*Rtau**2)
if(chi > 0.0) then
  chi = (1.0+680.0*chi**2)/(1.0+400.0*chi**2)
else
  chi = 1.0;
end if
Sk = nu(j)*uprime(j)**2 - const(4)*Rtau**2*k(j)*h(j)*(4.0/15.0 + (Rt/8.0)**4)/(1.0 +
(Rt/8.0)**4)*chi
RETURN; END FUNCTION Sk

REAL FUNCTION Sh(j,m,Rtau,y,k,h,nu,uprime,udprime,kprime,kdprime,hprime,const,coord)
IMPLICIT NONE; INTEGER::j,m,coord;
REAL,DIMENSION(m)::y,k,h,nu,uprime,udprime,kprime,kdprime,hprime;
REAL,DIMENSION(4)::const;
Real::Rtau,r,Rt
Rt = k(j)/h(j)
Sh = const(1)*(1.0/9.0 + Rt/2.95)/(1.0 + Rt/2.95)*uprime(j)**2 - const(2)*Rtau**2*h(j)**2
RETURN; END FUNCTION Sh

REAL FUNCTION fk0(j,m,Rtau,y,k,h,nu,uprime,udprime,kprime,kdprime,hprime,const,coord)
IMPLICIT NONE; INTEGER::j,m,coord;
REAL,DIMENSION(m)::y,k,h,nu,uprime,udprime,kprime,kdprime,hprime;
REAL,DIMENSION(4)::const;
Real::Rtau
fk0 = 0.0
RETURN; END FUNCTION fk0

REAL FUNCTION fh0(j,m,Rtau,y,k,h,nu,uprime,udprime,kprime,kdprime,hprime,const,coord)
IMPLICIT NONE; INTEGER::j,m,coord;
REAL,DIMENSION(m)::y,k,h,nu,uprime,udprime,kprime,kdprime,hprime;
REAL,DIMENSION(4)::const;
Real::Rtau
if(y(j)<1e-15) then
  fh0 = 1.0e33
else
  fh0 = 6.0/(const(2)*y(j)**2)
end if
RETURN; END FUNCTION fh0

!----- Van Driest SOLUTION-----

SUBROUTINE VanDriestSolution(m,yplus,VDconst,VDu)
IMPLICIT NONE

```

```

INTEGER::m, j
REAL, DIMENSION(m)::yplus, VDu
REAL, DIMENSION(2)::VDconst
REAL::du(4), u, dy
VDu(m) = 0.0
do j=2,m,1
  dy = yplus(m-j+1)-yplus(m-j+2)
  call rnkta4(1,VDconst,yplus(m-j+2),VDu(m-j+2),dy,du,u)
  VDu(m-j+1) = u
end do
END SUBROUTINE VanDriestSolution

!----- Van Driest called from Runge-kutta-----
-----
REAL FUNCTION f(i,const,yplus,uplus)
implicit none
INTEGER::i
REAL::yplus,uplus,kappa,Aplus,a,b,c
REAL, DIMENSION(2)::const
kappa = const(1)
Aplus = const(2)
a = kappa**2*yplus**2*(1.0-exp(-yplus/Aplus))**2
b = 1.0
c = -1.0
f = max((-b + sqrt(b**2 - 4.0*a*c))/(2.0*a), (-b - sqrt(b**2 - 4.0*a*c))/(2.0*a))
if(yplus .eq. 0) then
  f = 1.0
end if
RETURN; END FUNCTION f

!-----Finite Difference Support -----
SUBROUTINE FDMatrix(order,m,y,deriv,ans)
!This subroutine calculates the 8th Order Coefficient matrix of Finite Differences
!m = size of vector of y coordinates
!y = vector of coordinates
!deriv = 1 for 1st Deriv, 2 for 2nd deriv
!ans = output matrix mxm of finite difference coefficients
IMPLICIT NONE
INTEGER::order,m,deriv,j,index,n,lower,upper
REAL, DIMENSION(m)::y
REAL, DIMENSION(m,m)::ans
REAL, DIMENSION(order+2)::ans_vec
if(deriv .eq. 1) then ! 1st Derivative
  n = order+1
  index=order/2
  lower = order/2
  upper = order/2
end if
if(deriv .eq. 2) then ! 2nd Derivative
  n = order+2
  index=n/2
  lower = (n-2)/2
  upper = n/2
end if

!write(*,*) 'n = ',n
!write(*,*) 'index = ',index
!write(*,*) 'lower = ',lower
!write(*,*) 'upper = ',upper

do j=1,index
  CALL FiniteDifference(n,j,y(1:n),deriv,ans_vec)
  ans(j,1:n)=ans_vec(:)
end do
do j=index+1,m-index
  CALL FiniteDifference(n,lower+1,y(j-lower:j+upper),deriv,ans_vec)
  ans(j,j-lower:j+upper) = ans_vec(:)
end do
do j=m-index+1,m
  CALL FiniteDifference(n,n-(m-j),y(m-n+1:m),deriv,ans_vec)
  ans(j,m-n+1:m) = ans_vec(:)
end do
RETURN
END SUBROUTINE FDMatrix

SUBROUTINE FiniteDifference(n, index, pts, deriv, ans)
!deriv = 1 for 1st Derivative
!deriv = 2 for 2nd Derivative
!n = size of array being passed in

```

```

!index = indice of "point of interest" within array
!pts = array of coordinate points
!ans = coefficients for each of the array of points
!Order of approximation is n-1 for 1st derivative
!Order of approximation is n-2 for 2nd derivative
  IMPLICIT NONE
  INTEGER::n,index,i,j,deriv
  REAL,DIMENSION(n)::pts,ans
  REAL,DIMENSION(n-1,n-1)::A,Ainv
  REAL::factorial
  A = 0.0
  do i=1,n-1
    do j=1,n-1
      A(i,j) = (pts(i)-pts(index))**j/Factorial(j)
    end do
  end do
  do i=index+1,n
    do j=1,n-1
      A(i-1,j) = (pts(i)-pts(index))**j/Factorial(j)
    end do
  end do
  CALL matinv(n-1,A,Ainv)
  ans(:) = Ainv(deriv,:)
  ans(index+1:n)=Ainv(deriv,index:n-1)
  ans(index) = -sum(Ainv(deriv,:))
  RETURN
END SUBROUTINE FiniteDifference

SUBROUTINE Derivative(m,FD,phi,ans)
!m = size of vector
!FD = Finite Difference Matrix size mxm
!phi = vector of funtion values
!ans = output vector size m of derivative values
  IMPLICIT NONE
  INTEGER::m,j
  REAL,DIMENSION(m,m)::FD
  REAL,DIMENSION(m)::phi,ans
  REAL::dot

  do j=1,m
    ans(j) = dot(m,FD(j,:),phi(:))
!write(*,*) 'ans(',j,')=',ans(j)
  end do
  RETURN
END SUBROUTINE Derivative

!----- Matrix Support -----
SUBROUTINE MatrixBlend(m,gamma,A,x,B)
!m = size of matrix (mxm)
!gamma = relaxation factor
!A = matrix mxm
!x = current solution vector
!B = Right-Hand Side vector
  IMPLICIT NONE
  INTEGER::m,j,i
  REAL::gamma,temp,dot
  REAL,DIMENSION(m,m)::A
  REAL,DIMENSION(m)::x,B

  do j=1,m
    temp = A(j,j)
    B(j) = B(j) - (1.0-gamma)*(dot(m,A(j,:),x)-temp*x(j))
    A(j,:) = gamma*A(j,:)
    A(j,j) = temp
  end do
  RETURN
END SUBROUTINE MatrixBlend

!----- FUNCTIONS -----

REAL FUNCTION Factorial(n)
  IMPLICIT NONE
  INTEGER::n,i
  Factorial=1.0
  do i=1,n
    Factorial = Factorial*real(i)
  end do

```

```

END FUNCTION Factorial

REAL FUNCTION dot(n,v1,v2)
  IMPLICIT NONE
  INTEGER::n,i
  REAL,DIMENSION(n)::v1,v2
  dot=0.0
  do i=1,n
    dot = dot + v1(i)*v2(i)
  end do
END FUNCTION dot

!----- Runge-Kutta Solver -----

      subroutine rkta4(n,a,t0,y0,dt,dy,y)
! This single precision subroutine computes a value for the n component
! vector y(t0+dt) from a known value of the vector y(t0)=y0. The solution
! is based on a fourth order Runge-Kutta solution to the system of n
! differential equations,
!
!           dy(i)/dt = f(i,a,t,y)           i = 1,2,3,...,n
!
! where a is a coefficient array passed to the function f. The single
! precision function subprogram f(i,a,t,y) must be provided by the user.
!
      real f,a(*),t0,y0(n),dt,dy(n,4),y(n),c(4)
      c(1) = 1.0/6.0
      c(2) = 1.0/3.0
      c(3) = c(2)
      c(4) = c(1)
      do j=1,n
        dy(j,1)=f(j,a,t0,y0)*dt
        y(j)=y0(j)+dy(j,1)/2.
      end do
      do j=1,n
        dy(j,2)=f(j,a,t0+dt/2.,y)*dt
      end do
      do j=1,n
        y(j)=y0(j)+dy(j,2)/2.
      end do
      do j=1,n
        dy(j,3)=f(j,a,t0+dt/2.,y)*dt
      end do
      do j=1,n
        y(j)=y0(j)+dy(j,3)
      end do
      do j=1,n
        dy(j,4)=f(j,a,t0+dt,y)*dt
      end do
      do j=1,n
        y(j)=y0(j)
        do i=1,4
          y(j)=y(j)+c(i)*dy(j,i)
        end do
      end do
      return
      end subroutine rkta4

```

APPENDIX E

ONE-DIMENSIONAL FINITE-VOLUME FORMULATION

I. Governing Equations

A fairly general system of equations for fully developed turbulent channel flow can be written in dimensional form for Cartesian coordinates including the momentum transport equation, turbulent kinetic energy equation, and one additional transport equation as

$$\begin{aligned}
 & -\frac{d}{dy} \left[(v + v_t) \left(\frac{du}{dy} \right) \right] = f_u(u, y) \\
 & -\frac{d}{dy} \left[(v + v_t / \sigma_k) \frac{dk}{dy} \right] = f_{k1}(k, h, v, y) \left(\frac{du}{dy} \right)^2 - f_{k2}(k, h, v, y) - \frac{f_{k3}(k, h, v, y)}{2k} \left(\frac{dk}{dy} \right)^2 \\
 & -\frac{d}{dy} \left[(v + v_t / \sigma_h) \frac{dh}{dy} \right] = f_{h1}(k, h, v, y) \left(\frac{du}{dy} \right)^2 - f_{h2}(k, h, v, y) + f_{h3}(k, h, v, y) \left(\frac{d^2u}{dy^2} \right)^2 \\
 & v_t = f_v(k, h, y)
 \end{aligned} \tag{E.1}$$

with the boundary conditions

$$u(0) = 0, \quad k(0) = 0, \quad \frac{dk}{dy}(0) = 0, \quad \frac{du}{dy}(l) = 0, \quad \frac{dk}{dy}(l) = 0, \quad \frac{dh}{dy}(l) = 0 \tag{E.2}$$

where h represents the second turbulence variable ε or ω and l is the channel half-width.

II. Coordinate Transformation

A general transport equation for any variable, ϕ , for fully developed channel flow can be written in Cartesian coordinates as

$$-\frac{d}{dy} \left[\Gamma \frac{d\phi}{dy} \right] = S(C, u, k, h, v, y) \tag{E.3}$$

where Γ is the diffusion coefficient, S includes the source terms, and C is an array of constants. We wish to solve this equation numerically using the finite-volume method. This method requires the domain to be discretized into a finite number of control volumes which will be referred to as cells. To facilitate variable

cell sizes, it is beneficial to rewrite Eq. (E.3) in terms of the independent variable η where $y = y(\eta)$. This provides a coordinate transformation from the physical domain in Cartesian coordinates to the computational domain in the transformed coordinate system. However, in order to apply this coordinate transformation, the one-dimensional physical domain must be divided into cells using variable spacing in y such that $y(\eta)$ and its first derivative are continuous. The Jacobian of the one-dimensional transformation is a scalar defined as

$$J \equiv \frac{d\eta}{dy} \quad (\text{E.4})$$

From this coordinate transformation, the first derivative of any continuous function, $\phi = \phi(y)$, can be written as

$$\frac{d\phi}{dy} = \frac{d\phi}{dy} \frac{dy}{d\eta} \frac{d\eta}{dy} = J \frac{d\phi}{d\eta} \quad (\text{E.5})$$

Applying Eq. (E.5) to Eq. (E.3) gives the general transport equation for any variable, ϕ , for fully developed channel flow in the transformed coordinate system

$$-J \frac{d}{d\eta} \left[\Gamma J \frac{d\phi}{d\eta} \right] = S \quad (\text{E.6})$$

This equation can be solved numerically using the finite volume method. Once a solution in the computational domain has been attained, the solution must be transformed to the physical domain to be of practical use. Details on a method for transforming from the computational domain to the physical domain or vice versa is included in Appendix F and will not be repeated here. This development will continue by presenting a method for solving Eq. (E.6) numerically in the computational domain.

III. Discretization

The finite-volume method is applied to this one-dimensional problem by integrating Eq. (E.6) across a control volume in the η -direction. This can be written as

$$\int \left\{ -J \frac{d}{d\eta} \left[\Gamma J \frac{d\phi}{d\eta} \right] = S \right\} d\eta \quad (\text{E.7})$$

Before continuing, the control volumes for the integration must be defined. In this case, the domain is divided into m cells with variable spacing in the physical domain. The definition of the coordinate transformation requires the transformation function to be continuous. Because the central difference scheme is only second order for midpoint calculations, uniform spacing in η is chosen. For simplicity, the grid spacing in the computational domain is set to $\Delta\eta = 1.0$. Each cell has a node P placed at the cell center in the computational domain at which the values of the dependent variables will be evaluated. Additionally, each cell has a north and south face denoted by points n and s . Neighboring north and south cell centers are denoted here as nodes N and S respectively. The subscript j is used here as the index variable. In the transformed coordinate system, node $j=1$ is located at $\eta = 0.5$, and node $j=m$ is located at $\eta = m - 0.5$. Figure E.1 shows the relationship between the physical domain and computational domain as well as the cell nomenclature. Note that the centers of the cells in the computational domain do not correspond to centers of the cells in the physical domain.

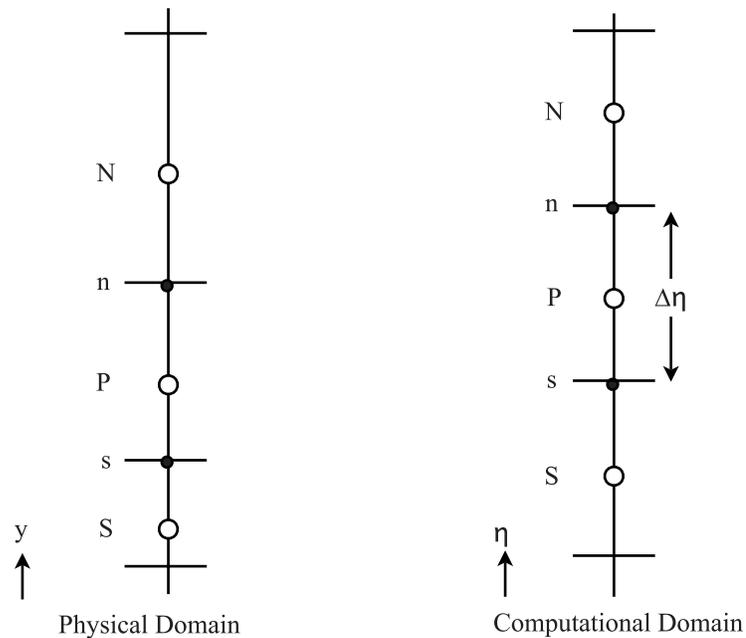


Fig. E.1 Comparison of physical and transformed domains.

Applying the transformation

$$y(\eta) = l \frac{(\beta + 1) - (\beta - 1) \{ [(\beta + 1)/(\beta - 1)]^{1-\eta/m} \}}{[(\beta + 1)/(\beta - 1)]^{1-\eta/m} + 1} \quad (\text{E.8})$$

where β is a grid-stretching factor with the limits $1 < \beta < \infty$, gives

$$\eta(y) = m - m \frac{\ln \{ [\beta + 1 - (y/l)] / [\beta - 1 + (y/l)] \}}{\ln [(\beta + 1)/(\beta - 1)]} \quad (\text{E.9})$$

The Jacobian must be nonzero over the domain to ensure that the transformation is not singular. Taking the derivative of Eq. (E.9) with respect to y gives

$$J \equiv \frac{d\eta}{dy} = \frac{2m\beta}{l\{\beta^2 - [1 - (y/l)]^2\} \ln [(\beta + 1)/(\beta - 1)]} \quad (\text{E.10})$$

The derivative of the Jacobian with respect to η is

$$J' = \frac{dJ}{d\eta} = \frac{dJ}{dy} \bigg/ \frac{d\eta}{dy} = \frac{1}{J} \frac{dJ}{dy} = \frac{-2[1 - (y/l)]}{l\{\beta^2 - [1 - (y/l)]^2\}} \quad (\text{E.11})$$

where the prime represents a derivative with respect to η . Equations (E.10) and (E.11) are exact for the transformation given in Eqs. (E.8) and (E.9). However, for an arbitrary transformation, the Jacobian and its derivative with respect to η can be estimated to second-order accuracy where the y -coordinates of the grid are known at the cell faces. For node P , second-order differencing gives

$$J_P = \frac{\Delta\eta}{y_n - y_s} \quad (\text{E.12})$$

Once the Jacobian is known at every node, the derivative of the Jacobian with respect to η can be calculated using the second-order differences given in Eqs. (H.13) and (H.16). Using j as the index,

$$\begin{aligned} J'_1 &= \frac{-4J_0 + 3J_1 + J_2}{3\Delta\eta}, & j = 1 \\ J'_j &= \frac{J_{j+1} - J_{j-1}}{2\Delta\eta}, & 2 \leq j \leq m-1 \\ J'_m &= \frac{J_m - J_{m-1}}{2\Delta\eta}, & j = m \end{aligned} \quad (\text{E.13})$$

where J_0 is the estimate for the Jacobian at the wall given by

$$J_0 = \frac{3J_1 - J_2}{2} \quad (\text{E.14})$$

The finite volume method continues by evaluating Eq. (E.7) for a single cell

$$-\left[\Gamma J \frac{d\phi}{d\eta} \right]_s^n = \int_s^n \frac{S}{J} d\eta \quad (\text{E.15})$$

Applying the central difference approximations given in Eqs. (H.18) and (H.19) to the left-hand side and the midpoint rule given in Eq. (H.29) to the right-hand side of Eq. (E.15) and simplifying gives the discretized form of the transport equation for the cells $2 \leq j \leq m-1$

$$-\frac{(\Gamma_N + \Gamma_P)(J_N + J_P)(\phi_N - \phi_P)}{2 \cdot 2 \cdot \Delta\eta} + \frac{(\Gamma_S + \Gamma_P)(J_S + J_P)(\phi_P - \phi_S)}{2 \cdot 2 \cdot \Delta\eta} = S_P \frac{\Delta\eta}{J_P} \quad (\text{E.16})$$

Using the subscript j as an index value, Eq. (E.16) can be rewritten as

$$\begin{aligned} & \text{for } 2 \leq j \leq m-1 \\ & \left[\frac{(\Gamma_{j+1} + \Gamma_j)(J_{j+1} + J_j)}{4\Delta\eta} + \frac{(\Gamma_{j-1} + \Gamma_j)(J_{j-1} + J_j)}{4\Delta\eta} \right] \phi_j \\ & - \frac{(\Gamma_{j+1} + \Gamma_j)(J_{j+1} + J_j)}{4\Delta\eta} \phi_{j+1} - \frac{(\Gamma_{j-1} + \Gamma_j)(J_{j-1} + J_j)}{4\Delta\eta} \phi_{j-1} = S_j \frac{\Delta\eta}{J_j} \end{aligned} \quad (\text{E.17})$$

For the cells $j=1$ and $j=m$, discretization of Eq. (E.15) must account for the influence of the boundary. Because all transport properties are symmetric about the centerline, $\phi_{m+1} = \phi_m$. Applying this to Eq. (E.17) gives the discretized equation for cell m

$$\left[\frac{(\Gamma_{m-1} + \Gamma_m)(J_{m-1} + J_m)}{4\Delta\eta} \right] \phi_m - \frac{(\Gamma_{m-1} + \Gamma_m)(J_{m-1} + J_m)}{4\Delta\eta} \phi_{m-1} = S_m \frac{\Delta\eta}{J_m} \quad (\text{E.18})$$

The wall boundary conditions affect the discretized equation for the first cell off of the boundary and are dependent on the transport property. These will be discussed in the appropriate sections below.

IV. Transport Equations

A. x -Momentum

The x -momentum equation is a special case of Eq. (E.3) where

$$\phi = u, \quad \Gamma = \nu + \nu_t, \quad S = f_u(u, y) \quad (\text{E.19})$$

Using these definitions, the right-hand side of Eq. (E.16) can be written for any cell, j as

$$S_j \frac{\Delta\eta}{J_j} = f_u(u_j, y_j) \frac{\Delta\eta}{J_j} \quad (\text{E.20})$$

For the cell adjacent to the wall boundary, Eq. (E.15) must be discretized using forward difference approximations. Here we define the points such that u_0 is the velocity at the wall, u_1 is the velocity at the center of the first cell, and u_2 is the velocity at the center of the second cell. Values and derivatives on the cell faces are estimated by using the forward difference approximations given in Eqs. (H.21) and (H.22) for the boundary face, and the central difference approximations given in Eqs. (H.18) and (H.19) for the face opposite the boundary face. Applying these relations gives the discretized form of the x -momentum equation for the cell adjacent to the wall boundary

$$-\frac{(\Gamma_2 + \Gamma_1)(J_2 + J_1)(u_2 - u_1)}{2 \cdot 2 \cdot \Delta\eta} + \frac{(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)(-8u_0 + 9u_1 - u_2)}{2 \cdot 2 \cdot 3\Delta\eta} = f_u(u_1, y_1) \frac{\Delta\eta}{J_1} \quad (\text{E.21})$$

This can be rewritten as

$$\begin{aligned} & \left[\frac{(\Gamma_2 + \Gamma_1)(J_2 + J_1)}{4\Delta\eta} + \frac{9(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)}{12\Delta\eta} \right] u_1 \\ & - \left[\frac{(\Gamma_2 + \Gamma_1)(J_2 + J_1)}{4\Delta\eta} + \frac{(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)}{12\Delta\eta} \right] u_2 = \frac{8(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)}{12\Delta\eta} u_0 + f_u(u_1, y_1) \frac{\Delta\eta}{J_1} \end{aligned} \quad (\text{E.22})$$

The no-slip boundary condition at the wall is imposed by setting $u_0 = 0.0$.

In summary, Eqs. (E.17), (E.18), and (E.22) can be used to get an improved estimate for the velocity distribution given an estimate for the dependent variables. These equations can be written as a tridiagonal system of equations including a conventional underrelaxation factor, Ω_u , and a blending factor, Γ_u . When $\Gamma_u = 1$, the method becomes a pure traditional tridiagonal algorithm. When $\Gamma_u = 0$, the method is a pointwise successive underrelaxation algorithm. Using the notation $T_{Lj}u_{j-1} + T_{Dj}u_j + T_{Uj}u_{j+1} = B_j$ for row j , the algorithm can be written as

for $1 \leq j \leq m$; $\Gamma_j = \nu + f_v(k_j, h_j, y_j)$; $B_j = f_u(u_j, y_j) \Delta\eta / J_j$;

$$T_{L1} = 0; \quad T_{D1} = \left[\frac{(\Gamma_2 + \Gamma_1)(J_2 + J_1)}{4\Delta\eta} + \frac{9(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)}{12\Delta\eta} \right];$$

$$T_{U1} = - \left[\frac{(\Gamma_2 + \Gamma_1)(J_2 + J_1)}{4\Delta\eta} + \frac{(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)}{12\Delta\eta} \right];$$

for $2 \leq j \leq m-1$;

$$T_{Lj} = - \frac{(\Gamma_{j-1} + \Gamma_j)(J_{j-1} + J_j)}{4\Delta\eta}; \quad T_{Uj} = - \frac{(\Gamma_{j+1} + \Gamma_j)(J_{j+1} + J_j)}{4\Delta\eta}; \quad T_{Dj} = -T_{Lj} - T_{Uj}; \quad (\text{E.23})$$

$$T_{Lm} = - \frac{(\Gamma_{m-1} + \Gamma_m)(J_{m-1} + J_m)}{4\Delta\eta}; \quad T_{Dm} = -T_{Lm}; \quad T_{Um} = 0;$$

$$B_1 = B_1 - (1 - \alpha_u)(T_{U1}u_2); \quad T_{U1} = \alpha_u T_{U1};$$

for $2 \leq j \leq m-1$; $B_j = B_j - (1 - \Gamma_u)(T_{Lj}u_{j-1} + T_{Uj}u_{j+1})$; $T_{Lj} = \Gamma_u T_{Lj}$; $T_{Uj} = \Gamma_u T_{Uj}$;

$$B_m = B_m - (1 - \Gamma_u)(T_{Lm}u_{m-1}); \quad T_{Lm} = \Gamma_u T_{Lm};$$

$$\mathbf{u}_{\text{old}} = \mathbf{u}; \quad \mathbf{u} = [\mathbf{T}]^{-1} \mathbf{B}; \quad \mathbf{u} = \mathbf{u}_{\text{old}} + \Omega_u (\mathbf{u} - \mathbf{u}_{\text{old}});$$

Once an estimate for the velocity profile has been obtained, an estimate for the shear velocity at the wall can be calculated. The shear velocity is related to the shear stress according to $u_\tau \equiv \sqrt{\tau_w / \rho}$. The shear stress can be calculated at the wall as

$$\tau_w \equiv \mu \frac{\partial u}{\partial y}(0) = J_0 \mu \frac{\partial u}{\partial \eta}(0) \quad (\text{E.24})$$

Using the second-order forward difference approximations given in Eqs. (H.21) and (H.22), the wall shear stress can be calculated for a given velocity profile as

$$\tau_w = \rho \frac{(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)}{4} \frac{(-8u_0 + 9u_1 - u_2)}{3\Delta\eta} = \rho \frac{(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)}{4} \frac{(9u_1 - u_2)}{3\Delta\eta} \quad (\text{E.25})$$

B. Turbulent Kinetic Energy

The turbulent-kinetic-energy equation is a special case of Eq. (E.3) where

$$\phi = k, \quad \Gamma = \nu + \nu_t / \sigma_k, \quad S = f_{k1}(k, h, \nu, y) \left(\frac{du}{dy} \right)^2 - f_{k2}(k, h, \nu, y) - \frac{f_{k3}(k, h, \nu, y)}{2k} \left(\frac{dk}{dy} \right)^2 \quad (\text{E.26})$$

Using these definitions and rearranging the last term in the source term, the right-hand side of Eq. (E.16) can be written for any cell, j as

$$S_j \frac{\Delta\eta}{J_j} = f_{k1}(k_j, h_j, \nu_j, y_j) (u'_j)^2 J_j \Delta\eta - f_{k2}(k_j, h_j, \nu_j, y_j) \frac{\Delta\eta}{J_j} - 2f_{k3}(k_j, h_j, \nu_j, y_j) [(k^{1/2})']^2 J_j \Delta\eta \quad (\text{E.27})$$

where the prime represents a derivative with respect to η , and the relationship $k'/(2k^{1/2}) = (k^{1/2})'$ has been applied. The first derivative for any property is estimated by the central difference approximation given in Eq. (H.13) for cells $2 \leq j \leq m$ and the forward difference approximation given in Eq. (H.16) for the cell $j = 1$. Applying these equations to the derivatives in Eq. (E.27) as well as the symmetry boundary condition at node m , gives the finite difference approximations

$$\begin{aligned} u'_1 &= \frac{-4u_0 + 3u_1 + u_2}{3\Delta\eta}, & (k^{1/2})'_1 &= \frac{-4k_0^{1/2} + 3k_1^{1/2} + k_2^{1/2}}{3\Delta\eta}, & j &= 1 \\ u'_j &= \frac{u_{j+1} - u_{j-1}}{2\Delta\eta}, & (k^{1/2})'_j &= \frac{k_{j+1}^{1/2} - k_{j-1}^{1/2}}{2\Delta\eta}, & 2 \leq j &\leq m-1 \\ u'_m &= \frac{u_m - u_{m-1}}{2\Delta\eta}, & (k^{1/2})'_m &= \frac{k_m^{1/2} - k_{m-1}^{1/2}}{2\Delta\eta}, & j &= m \end{aligned} \quad (\text{E.28})$$

The no-slip boundary condition for the turbulent kinetic energy requires that k and its first derivative both be zero at the wall. Using the forward difference approximation given in Eq. (H.22) imposes the following relationship between the wall value of k , denoted here as k_0 , and the first two nodes off of the wall, k_1 and k_2 respectively

$$\begin{aligned} k_0 &= 0 \\ -8k_0 + 9k_1 - k_2 &= 0 \end{aligned} \quad (\text{E.29})$$

This relationship constrains the value of k_1 relative to k_2 . Therefore, the governing transport equation for k need not be solved for the first node off of the wall when the correct boundary conditions are applied. Some implementations of turbulence models neglect to enforce the second equation in Eq. (E.29) and solve the governing turbulent kinetic energy equation with $k_0 = 0$ as the only boundary condition. In that case, the governing equation is discretized similar to Eq. (E.24). However, this is incorrect. Both equations in Eq. (E.29) must be enforced to correctly model the no-slip boundary condition. Correct implementation of the no-slip boundary condition for k at the wall eliminates the need for the first equations in Eq. (E.28) because the source term given in Eq. (E.27) need not be evaluated for the cell adjacent to the boundary.

In summary, Eqs. (E.17), (E.27), and (E.29) can be used to get an improved estimate for the turbulent kinetic energy distribution given an estimate for the dependent variables. Just as was done with the x -

momentum equations, these equations can be written as a tridiagonal system of equations including a conventional underrelaxation factor, Ω_k , and a blending factor, Γ_k . The algorithm can be written as

$$\text{for } 1 \leq j \leq m; \quad \Gamma_j = \nu + f_\nu(k_j, h_j, y_j) / \sigma_k;$$

(correct no - slip boundary condition)

$$T_{L1} = 0; \quad T_{D1} = 9; \quad T_{U1} = -1; \quad B_1 = 0$$

(incorrect no - slip boundary condition)

$$u'_1 = \frac{3u_1 + u_2}{3\Delta\eta}; \quad (k^{1/2})'_1 = \frac{3k_1^{1/2} + k_2^{1/2}}{3\Delta\eta};$$

$$T_{L1} = 0; \quad T_{D1} = \left[\frac{(\Gamma_2 + \Gamma_1)(J_2 + J_1)}{4\Delta\eta} + \frac{9(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)}{12\Delta\eta} \right];$$

$$T_{U1} = - \left[\frac{(\Gamma_2 + \Gamma_1)(J_2 + J_1)}{4\Delta\eta} + \frac{(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)}{12\Delta\eta} \right];$$

$$B_1 = f_{k1}(k_1, h_1, \nu_1, y_1) u_1'^2 J_1 \Delta\eta - f_{k2}(k_1, h_1, \nu_1, y_1) \frac{\Delta\eta}{J_1} - 2f_{k3}(k_1, h_1, \nu_1, y_1) [(k^{1/2})'_1]^2 J_1 \Delta\eta;$$

for $2 \leq j \leq m - 1$

$$u'_j = \frac{u_{j+1} - u_{j-1}}{2\Delta\eta}; \quad (k^{1/2})'_j = \frac{k_{j+1}^{1/2} - k_{j-1}^{1/2}}{2\Delta\eta};$$

$$T_{Lj} = - \frac{(\Gamma_{j-1} + \Gamma_j)(J_{j-1} + J_j)}{4\Delta\eta}; \quad T_{Uj} = - \frac{(\Gamma_{j+1} + \Gamma_j)(J_{j+1} + J_j)}{4\Delta\eta}; \quad T_{Dj} = -T_{Lj} - T_{Uj};$$

$$u'_m = \frac{u_m - u_{m-1}}{2\Delta\eta}; \quad (k^{1/2})'_m = \frac{k_m^{1/2} - k_{m-1}^{1/2}}{2\Delta\eta}; \quad T_{Lm} = - \frac{(\Gamma_{m-1} + \Gamma_m)(J_{m-1} + J_m)}{4\Delta\eta}; \quad T_{Dm} = -T_{Lm}; \quad T_{Um} = 0;$$

for $2 \leq j \leq m$

$$B_j = f_{k1}(k_j, h_j, \nu_j, y_j) u_j'^2 J_j \Delta\eta - f_{k2}(k_j, h_j, \nu_j, y_j) \frac{\Delta\eta}{J_j} - 2f_{k3}(k_j, h_j, \nu_j, y_j) [(k^{1/2})'_j]^2 J_j \Delta\eta; \quad (\text{E.30})$$

$$B_1 = B_1 - (1 - \alpha_k)(T_{U1} k_2); \quad T_{U1} = \alpha_k T_{U1};$$

$$\text{for } 2 \leq j \leq m - 1; \quad B_j = B_j - (1 - \Gamma_k)(T_{Lj} k_{j-1} + T_{Uj} k_{j+1}); \quad T_{Lj} = \Gamma_k T_{Lj}; \quad T_{Uj} = \Gamma_k T_{Uj};$$

$$B_m = B_m - (1 - \Gamma_k)(T_{Lm} k_{m-1}); \quad T_{Lm} = \Gamma_k T_{Lm};$$

$$\mathbf{k}_{\text{old}} = \mathbf{k}; \quad \mathbf{k} = [\mathbf{T}]^{-1} \mathbf{B}; \quad \mathbf{k} = \mathbf{k}_{\text{old}} + \Omega_k (\mathbf{k} - \mathbf{k}_{\text{old}});$$

where both the correct and incorrect implementations of the no-slip boundary condition on k are given. This gives an improved estimate for k based on an estimate for the dependent variables.

C. Second Turbulence Variable, h

The second turbulence variable transport equation is a special case of Eq. (E.3) where

$$\phi = h, \quad \Gamma = \nu + \nu_t / \sigma_h, \quad S = f_{h1}(k, h, \nu, y) \left(\frac{du}{dy} \right)^2 - f_{h2}(k, h, \nu, y) + f_{h3}(k, h, \nu, y) \left(\frac{d^2u}{dy^2} \right)^2 \quad (\text{E.31})$$

Using these definitions and expanding the second derivative in the last term in the source terms, the right-hand side of Eq. (E.16) can be written for any cell, j as

$$S_j \frac{\Delta\eta}{J_j} = f_{h1}(k_j, h_j, v_j, y_j)(u'_j)^2 J_j \Delta\eta - f_{h2}(k_j, h_j, v_j, y_j) \frac{\Delta\eta}{J_j} + f_{h3}(k_j, h_j, v_j, y_j)(J'_j u'_j + J_j u''_j)^2 J_j \Delta\eta \quad (\text{E.32})$$

where the prime represents a derivative with respect to η . The derivative in the first term on the right-hand side is calculated using Eq. (E.28). The last term on the right-hand side can be calculated using the central difference approximations given in Eqs. (H.13) and (H.14) for cells $2 \leq j \leq m$ and the forward difference approximations given in Eqs. (H.16) and (H.17) for the cell $j=1$. Applying these equations to the derivatives in Eq. (E.32) as well as the symmetry boundary condition at node m , and the no-slip boundary condition $u_0 = 0$, gives the finite difference approximations

$$\begin{aligned} u''_1 &= \frac{(-75u_1 + 30u_2 - 3u_3)}{15\Delta\eta^2}, & j=1 \\ u''_j &= \frac{(u_{j+1} - 2u_j + u_{j-1})}{\Delta\eta^2}, & 2 \leq j \leq m-1 \\ u''_m &= \frac{(-u_m + u_{m-1})}{\Delta\eta^2}, & j=m \end{aligned} \quad (\text{E.33})$$

Some cases are singular near a wall and require special treatment. In these cases, the asymptotic behavior of the h equation near the wall can be used in the near-wall region rather than solving the differential equation. Depending on the second transport variable, a predetermined number of cells off of the wall, m_h , will simply be set to the asymptotic solution for h

$$h_j = f_{h0}(j, \mathbf{k}, \mathbf{h}, \mathbf{y}), \quad 1 \leq j \leq m_h \quad (\text{E.34})$$

If the second turbulence variable and its governing equation are not singular at the wall, the governing equation is discretized similar to Eq. (E.22) and the asymptotic value of h is only needed to calculate h at the wall, $h_0 = f_{h0}(0, \mathbf{k}, \mathbf{h}, \mathbf{y})$.

In summary, an algorithm can be written to get an improved estimate for the second turbulence variable distribution given an estimate for the dependent variables. Just as was done above, the equations can be written as a tridiagonal system of equations including a conventional underrelaxation factor, Ω_h , and

a blending factor, Γ_h . If $m_h > 0$, the near-wall asymptotic solution is used for the cells $j \leq m_h$. Otherwise, the asymptotic solution is only used at the wall. The algorithm can be written as

$$\begin{aligned}
& \text{for } 1 \leq j \leq m; \quad \Gamma_j = \nu + f_\nu(k_j, h_j, y_j) / \sigma_h; \\
& \text{if } m_h > 0 \\
& \quad \text{for } 1 \leq j \leq m_h; \quad T_{Lj} = 0; \quad T_{Dj} = 1; \quad T_{Uj} = 0; \quad B_j = f_{h0}(j, \mathbf{k}, \mathbf{h}, \mathbf{y}); \\
& \text{else} \\
& \quad h_0 = f_{h0}(0, \mathbf{k}, \mathbf{h}, \mathbf{y}); \\
& \quad u'_1 = \frac{3u_1 + u_2}{3\Delta\eta}; \quad u''_1 = \frac{(-75u_1 + 30u_2 - 3u_3)}{15\Delta\eta^2}; \quad J'_1 = \frac{(-4J_0 + 3J_1 + J_2)}{3\Delta\eta}; \\
& \quad T_{L1} = 0; \quad T_{D1} = \left[\frac{(\Gamma_2 + \Gamma_1)(J_2 + J_1)}{4\Delta\eta} + \frac{9(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)}{12\Delta\eta} \right]; \\
& \quad T_{U1} = - \left[\frac{(\Gamma_2 + \Gamma_1)(J_2 + J_1)}{4\Delta\eta} + \frac{(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)}{12\Delta\eta} \right]; \\
& \quad B_1 = \frac{8(3\Gamma_1 - \Gamma_2)(3J_1 - J_2)}{12\Delta\eta} h_0 + f_{h1}(k_1, h_1, \nu_1, y_1)(u'_1)^2 J_1 \Delta\eta - f_{h2}(k_1, h_1, \nu_1, y_1) \frac{\Delta\eta}{J_1} \\
& \quad \quad + f_{h3}(k_1, h_1, \nu_1, y_1)(J'_1 u'_1 + J_1 u''_1)^2 J_1 \Delta\eta; \\
& \quad \text{for } \max(m_h + 1, 2) \leq j \leq m - 1 \\
& \quad \quad u'_j = \frac{u_{j+1} - u_{j-1}}{2\Delta\eta}; \quad u''_j = \frac{(u_{j+1} - 2u_j + u_{j-1})}{\Delta\eta^2}; \quad J'_j = \frac{(J_{j+1} - J_{j-1})}{2\Delta\eta}; \\
& \quad \quad T_{Lj} = - \frac{(\Gamma_{j-1} + \Gamma_j)(J_{j-1} + J_j)}{4\Delta\eta}; \quad T_{Uj} = - \frac{(\Gamma_{j+1} + \Gamma_j)(J_{j+1} + J_j)}{4\Delta\eta}; \quad T_{Dj} = -T_{Lj} - T_{Uj}; \\
& \quad \quad u'_m = \frac{u_m - u_{m-1}}{2\Delta\eta}; \quad u''_m = \frac{(-u_m + u_{m-1})}{\Delta\eta^2}; \quad J'_m = \frac{(J_m - J_{m-1})}{2\Delta\eta}; \\
& \quad \quad T_{Lm} = - \frac{(\Gamma_{m-1} + \Gamma_m)(J_{m-1} + J_m)}{4\Delta\eta}; \quad T_{Dm} = -T_{Lm}; \quad T_{Um} = 0; \\
& \quad \text{for } \max(m_h + 1, 2) \leq j \leq m \\
& \quad \quad B_j = f_{h1}(k_j, h_j, \nu_j, y_j)(u'_j)^2 J_j \Delta\eta - f_{h2}(k_j, h_j, \nu_j, y_j) \frac{\Delta\eta}{J_j} \\
& \quad \quad \quad + f_{h3}(k_j, h_j, \nu_j, y_j)(J'_j u'_j + J_j u''_j)^2 J_j \Delta\eta; \\
& \quad \quad B_1 = B_1 - (1 - \alpha_h)(T_{U1} h_2); \quad T_{U1} = \alpha_h T_{U1}; \\
& \quad \quad \text{for } 2 \leq j \leq m - 1; \quad B_j = B_j - (1 - \Gamma_h)(T_{Lj} h_{j-1} + T_{Uj} h_{j+1}); \quad T_{Lj} = \Gamma_h T_{Lj}; \quad T_{Uj} = \Gamma_h T_{Uj}; \\
& \quad \quad B_m = B_m - (1 - \Gamma_h)(T_{Lm} h_{m-1}); \quad T_{Lm} = \Gamma_h T_{Lm}; \\
& \quad \quad \mathbf{h}_{\text{old}} = \mathbf{h}; \quad \mathbf{h} = [\mathbf{T}]^{-1} \mathbf{B}; \quad \mathbf{h} = \mathbf{h}_{\text{old}} + \Omega_h (\mathbf{h} - \mathbf{h}_{\text{old}});
\end{aligned} \tag{E.35}$$

This gives an improved estimate for h given estimates for the dependent variables.

V. Solution Procedure

Equations (E.23), (E.30), and (E.35) provide algorithms for obtaining an improved estimate for the dependent variables u , k , and h respectively given an estimate for all the dependent variables. Given an initial guess, these three equations are used in turn to improve the estimate for the dependent variables. The process is repeated iteratively until a converged solution is obtained.

VI. Sample Results

It is helpful to test the code on equations with a known analytical solution such that the accuracy of the code can be determined. Three such cases are included here. Results for four turbulence models which were solved using this finite-volume algorithm can be found in Appendix B.

A. Laminar Flow

The governing equation and boundary conditions for fully developed laminar flow in a channel can be written

$$\frac{d}{dy} \left[\nu \left(\frac{du}{dy} \right) \right] = \frac{1}{\rho} \frac{d\hat{p}}{dx} \quad (\text{E.36})$$

$$u(0) = 0, \quad \frac{du}{dy}(l) = 0 \quad (\text{E.37})$$

Equation (E.36) is a special case of Eq. (E.1) where

$$f_v = 0, \quad f_u = -\frac{1}{\rho} \frac{d\hat{p}}{dx} \quad (\text{E.38})$$

and k and h are irrelevant. Because it is a second-order equation and f_u does not contain any “lagging” variables, the numerical solution to this case can be obtained by applying a single iteration of Eq. (E.23).

The closed-form solution to this case is

$$u(y) = \frac{1}{\mu} \frac{d\hat{p}}{dx} \left(\frac{y^2}{2} - ly \right) \quad (\text{E.39})$$

which can be used to evaluate the accuracy of the method.

B. A Second-Order Test Case

Consider the system of equations

$$-\left(\frac{d^2u}{dy^2}\right) = 1, \quad -\left(\frac{d^2k}{dy^2}\right) = 8, \quad -\left(\frac{d^2h}{dy^2}\right) = -0.3 \quad (\text{E.40})$$

with the boundary conditions

$$u(0) = 0, \quad k(0) = 0, \quad \frac{dk}{dy}(0) = 8, \quad \frac{du}{dy}(1) = 0, \quad \frac{dk}{dy}(1) = 0, \quad \frac{dh}{dy}(1) = 0 \quad (\text{E.41})$$

This system of equations is second-order in each variable. Equation (E.40) is a special case of Eq. (E.1)

where

$$l = 1, \quad \nu = 1, \quad f_v = 0, \quad f_u = 1, \quad f_{k1} = 0, \quad f_{k2} = -8, \quad f_{k3} = 0, \quad f_{h1} = 0, \quad f_{h2} = 0.3, \quad f_{h3} = 0 \quad (\text{E.42})$$

The closed-form solution to this system of equations is

$$\begin{aligned} u(y) &= y - \frac{y^2}{2} \\ k(y) &= 8y - 4y^2 \\ h(y) &= 0.20 - 0.30y + 0.15y^2 \end{aligned} \quad (\text{E.43})$$

This can be used to evaluate the accuracy of numerical solutions. Because h is not singular at the wall, the near-wall solution only needs to be used for the value of h at the wall. Therefore, we set $m_{h_i} = 0$. In the limit as $y \rightarrow 0$,

$$h_0 = 0.20 \quad (\text{E.44})$$

C. A Closed-Form k - ε Analogy

Consider the system of equations

$$-\left(\frac{d^2u}{dy^2}\right) = 1, \quad -\left(\frac{d^2k}{dy^2}\right) = y^4 \left(\frac{du}{dy}\right)^2 - h, \quad -\left(\frac{d^2h}{dy^2}\right) = y^2 \left(\frac{du}{dy}\right)^2 - y^6 \quad (\text{E.45})$$

with the boundary conditions

$$u(0) = 0, \quad k(0) = 0, \quad \frac{dk}{dy}(0) = 0, \quad \frac{du}{dy}(1) = 0, \quad \frac{dk}{dy}(1) = 0, \quad \frac{dh}{dy}(1) = 0 \quad (\text{E.46})$$

This system of equations is similar in order to the Lam-Bremhorst [68] turbulence model. Equation (E.45) is a special case of Eq. (E.1) where

$$\nu = 1, \quad f_\nu = 0, \quad f_u = 1, \quad f_{k1} = y^4, \quad f_{k2} = h, \quad f_{k3} = 0, \quad f_{h1} = y^2, \quad f_{h2} = y^6, \quad f_{h3} = 0 \quad (\text{E.47})$$

Because h is not singular at the wall, the near-wall solution only needs to be used for the value of h at the wall. Therefore, we set $m_h = 0$. In the limit as $y \rightarrow 0$, the k equation reduces to

$$h_0 = \left(\frac{d^2 k}{dy^2} \right)_0 \quad (\text{E.48})$$

This is not a boundary condition. However, it can be used to calculate h_0 . Using Eq. (E.5) along with the forward difference approximations given in Eqs. (H.22) and (H.23), the asymptotic solution for h at the wall is discretized as

$$\begin{aligned} h_0 &= \left(\frac{d^2 k}{dy^2} \right)_0 \\ &= J_0 (J'_0 k'_0 + J_0 k''_0) \\ &= J_0 \left[\left(\frac{-8J_0 + 9J_1 - J_2}{3\Delta\eta} \right) \left(\frac{-8k_0 + 9k_1 - k_2}{3\Delta\eta} \right) + J_0 \left(\frac{72k_0 - 120k_1 + 60k_2 - 12k_3}{15\Delta\eta^2} \right) \right] \end{aligned} \quad (\text{E.49})$$

where $J_0 = (3J_1 - J_2)/2$ is the finite difference approximation for the Jacobian at the wall. The closed-form solution to this system of equations is

$$\begin{aligned} u &= y - \frac{y^2}{2} \\ k &= \frac{169y^2 - 92y^3 - 182y^6 + 252y^7 - 93y^8 + y^{10}}{5040} \\ h &= \frac{169 - 276y - 210y^4 + 252y^5 - 84y^6 + 45y^8}{2520} \end{aligned} \quad (\text{E.50})$$

This can be used to evaluate the accuracy of numerical solutions.

VII. Sample Code

The following sample C++ code can be used to solve the Wilcox 1998 model for smooth walls.

```
#include <iostream>
#include <math.h>
using namespace std;
```

```

//kw model
double fu(double L, int j, double rho, double nu, double dpdx, double y, double k, double
h, double ut, int &limit){
    return -1.0/rho*dpdx; }
double fnu(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    double Rt = k/(nu*h);
    return k/h*(.024 + Rt/6.0)/(1.0 + Rt/6.0);}
double fk1(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    return fnu(L, j, rho, nu, dpdx, y, k, h, ut, limit);}
double fk2(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    double Rt = k/(nu*h);
    return 0.09*k*h*(4.0/15.0 + pow(Rt/8.0,4))/(1.0 + pow(Rt/8.0,4));}
double fk3(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    return 0.0;}
double fh1(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    double Rt = k/(nu*h);
    return 0.52*(1.0/9.0 + Rt/2.95)/(1.0 + Rt/2.95);}
double fh2(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    return 0.072*h*h;}
double fh3(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    return 0.0;}
double fh0(int j, double* k, double* h, double* y, double* J, double nu, double ut){
    return 6.0*nu/(.072*y[j]*y[j]);}
double U Exact(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    return 1.0/rho/nu*dpdx*(y*y/2.0 - L*y);} //laminar solution
double K Exact(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    return 0.0;}
double H Exact(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    return 6.0*.01/(.072*y*y);}
double U Init(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    return 1.0/rho/nu*dpdx*(y*y/2.0 - L*y);}
double K Init(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    return y/L*ut*ut;}
double H Init(double L, int j, double rho, double nu, double dpdx, double y, double k,
double h, double ut, int &limit){
    return 6.0*.01/(.072*y*y);}

int TriDiagSolver(int n, double* a, double* b, double* c, double* x, double* d)
//Returns x=[A]^-1 B
{
    // n = size of matrix
    // a = Lower Diagonal
    // b = Diagonal
    // c = Upper Diagonal
    // d = Right-Hand Side
    // x = vector of unknowns
    int k;
    double m;
    for(k = 2; k <=n; k++){
        {
            m = a[k]/b[k-1];
            b[k] = b[k] - m*c[k-1];
            d[k] = d[k] - m*d[k-1];
        }
        x[n] = d[n]/b[n];
        for(k = n-1; k > 0; k--) x[k] = (d[k] - c[k]*x[k+1])/b[k];
        return 0;
    }
}

int initzero(int size, double* a)
{
    for(int i = 0; i < size; i++) a[i] = 0.0;
    return 0;
}

```

```

int Jacobian_Exact(int n, double length, double growth, double* y_face, double* ynodes,
double* Jfaces, double* Jnodes) //Exact Roberts
{
    int i;
    //Exact Roberts Transformation
    for(i = 0; i <= n; i++) Jfaces[i] = double(n)*2.0*growth/(length*(growth*growth -
pow((1.0 - (y_face[i]/length)),2))*log((growth + 1.0)/(growth - 1.0)));
    for(i = 1; i <= n; i++) Jnodes[i] = double(n)*2.0*growth/(length*(growth*growth -
pow((1.0 - (ynodes[i]/length)),2))*log((growth + 1.0)/(growth - 1.0)));
    return 0;
}

int Jacobian_Difference(int n, double length, double growth, double* y_face, double*
ynodes, double* Jfaces, double* Jnodes) //Central Difference Approximation
{
    int i;
    //Numerical approximation for J
    Jfaces[0] = 2.0/(-3.0*y_face[0] + 4.0*y_face[1] - y_face[2]);
    for(i = 1; i < n; i++) Jfaces[i] = 2.0/(y_face[i+1] - y_face[i-1]);
    Jfaces[n] = -2.0/(-3.0*y_face[n] + 4.0*y_face[n-1] - y_face[n-2]);

    Jnodes[1] = 3.0/(-4.0*ynodes[0] + 3.0*ynodes[1] + ynodes[2]);
    for(i = 2; i < n; i++) Jnodes[i] = 2.0/(ynodes[i+1] - ynodes[i-1]);
    Jnodes[n] = -3.0/(-4.0*ynodes[n+1] + 3.0*ynodes[n] + ynodes[n-1]);

    return 0;
}

int Jacobian_Coarse(int n, double length, double growth, double* y_face, double* ynodes,
double* Jfaces, double* Jnodes) //Central Difference Approximation - very coarse
{
    int i;
    //Numerical approximation for J
    Jfaces[0] = 2.0/(-3.0*y_face[0] + 4.0*y_face[1] - y_face[2]);
    for(i = 1; i < n; i++) Jfaces[i] = 2.0/(y_face[i+1] - y_face[i-1]);
    Jfaces[n] = -2.0/(-3.0*y_face[n] + 4.0*y_face[n-1] - y_face[n-2]);

    for(i = 1; i <= n; i++) Jnodes[i] = 1.0/(y_face[i] - y_face[i-1]); //very coarse
approximation!
    return 0;
}

int UniformGrid(int n, double length, double growth, double* y_face, double* ynodes,
double* Jfaces, double* Jnodes) //Central Difference Approximation - very coarse
{
    int i;
    //uniform grid
    for(i = 0; i <= n; i++) //faces
    {
        y_face[i] = length/double(n)*double(i);
        Jfaces[i] = double(n)/length;
    }
    for(i = 1; i <= n; i++) //nodes
    {
        ynodes[i] = length/double(n)*(double(i)-.5);
        Jnodes[i] = double(n)/length;
    }
    ynodes[0] = y_face[0];
    ynodes[n+1] = y_face[n];
    Jnodes[0] = Jfaces[0];
    Jnodes[n+1] = Jfaces[n];
    return 0;
}

int makegrid(int n, double length, double growth, double* y_face, double* ynodes, double*
Jfaces, double* Jnodes)
{
    int i;
    double sum = 0.0;
    for(i = 1; i <= n; i++) sum += pow(growth,i);
    double mult = length/sum;

    // y_face[0] = 0.0;
    // for (i = 1; i < n; i++) y_face[i] = y_face[i-1]+pow(growth,i)*mult; //exponential
growth
    // y_face[n] = length;
    // for(i = 0; i <= n; i++) y_face[i] = length*double(i*i)/double(n*n); //quadratic
growth

```

```

//Exact Roberts Transformation
for(i = 0; i <= n; i++) y_face[i] = length*((growth + 1.0) - (growth -
1.0)*(pow(((growth + 1.0)/(growth-1.0)),1.0-double(i)/double(n)))/pow(((growth +
1.0)/(growth - 1.0)),1.0-double(i)/double(n)) + 1.0);
for(i = 1; i <= n; i++) ynodes[i] = length*((growth + 1.0) - (growth -
1.0)*(pow(((growth + 1.0)/(growth-1.0)),1.0-(double(i)-.5)/double(n)))/pow(((growth +
1.0)/(growth - 1.0)),1.0-(double(i)-.5)/double(n)) + 1.0);
ynodes[0] = y_face[0];
ynodes[n+1] = y_face[n];

//    Jacobian_Exact(n,length,growth,y_face,ynodes,Jfaces,Jnodes);
//    Jacobian_Difference(n,length,growth,y_face,ynodes,Jfaces,Jnodes);
Jacobian_Coarse(n,length,growth,y_face,ynodes,Jfaces,Jnodes);

Jnodes[0] = Jfaces[0];
Jnodes[n+1] = Jfaces[n];

//    UniformGrid(n,length,growth,y_face,ynodes,Jfaces,Jnodes);

return 0;
}

int Transport(int n, double* phi, double alpha, double omega, double* TU, double* TL,
double* TD, double* B, int print, int& limit)
{
    int j;
    double* phi_old = new double[n+2];

    if(print)
    {
        for(j = 1; j <= n; j++)
        {
            printf("%13.14f * %13.14f + %13.14f * %13.14f + %13.14f *
%13.14f\n",TL[j],phi[j-1],TD[j],phi[j],TU[j],phi[j+1]);
            printf(" = %13.14f vs %13.14f\n", B[j],TL[j]*phi[j-
1]+TD[j]*phi[j]+TU[j]*phi[j+1]);
        }
    }

    B[1] = B[1] - (1.0 - alpha)*TU[1]*phi[2]; TU[1] = alpha*TU[1];
    for(j = 2; j < n; j++)
    {
        B[j] = B[j] - (1.0-alpha)*(TL[j]*phi[j-1] + TU[j]*phi[j+1]);
        TL[j] = alpha*TL[j];
        TU[j] = alpha*TU[j];
    }
    B[n] = B[n] - (1.0 - alpha)*TL[n]*phi[n-1]; TL[n] = alpha*TL[n];
    for(j = 1; j <= n; j++) phi_old[j] = phi[j];

    TriDiagSolver(n,TL,TD,TU,phi,B);
    for(j = 1; j <= n; j++)
    {
        if(phi[j] != phi_old[j])
        {
            cout<<endl<<" -----> NAN detected. "<<j<<" "<<phi[j]<<endl;
            return 1;
        }
        phi[j] = phi_old[j] + omega*(phi[j] - phi_old[j]);
        if(limit > 0)
        {
            if(phi[j] < 0.0)
            {
                phi[j] = 1.0e-16;
                limit ++;
            }
        }
    }
}

delete phi_old;

return 0;
}

int main (int argc, char * const argv[])
{
    int i,j,iter,ulimit,klimit,hlimit,limit;
    double alphau, alphak, alphah;
    //    double omegau, omegak, omegah;

```

```

double omega,J0;
double urms,krms,hrms, urms_face, krms_face, hrms_face;
double L = 0.5;
double growth = 1.02;
int n = 25;    //number of cells

int itermax = 100000;
int print = 1000;

//Blending factors (1 = TDA) (0 = PSUR)
alphau = alphak = alphah = 0.5;

//Relaxation Factors
// omegau = omegak = omegah = .001;
omega = 0.05;

FILE* pfile;
int initfile = 1; //set to 0 for algebraic initialization

int m h = 1;
int forcekp = 1; //force k prime if this is set to 1
double dkpdyp = 0.0; //dk+/dy+ if it is forced
int ulimiter = 0; // 1 limits to positive numbers
int klimiter = 1;
int hlimiter = 1;

double nu = .01;
double rho = .1;
double dpdx = -7.2;
double tw = -dpdx*L;
double ut = sqrt(tw/rho);

double sigmak = 2.0;
double sigmah = 2.0;

double* y_face = new double[n+2];
double* y_ = new double[n+2];
double* Jfaces = new double[n+2];
double* J = new double[n+2];
double* dy = new double[n+2];
double* u = new double[n+2];
double* k = new double[n+2];
double* h = new double[n+2];
double* u_face = new double[n+2];
double* k_face = new double[n+2];
double* h_face = new double[n+2];
double* up = new double[n+2];
double* upp = new double[n+2];
double* kp = new double[n+2];
double* Jp = new double[n+2];
double* Gamma = new double[n+2];

double* TU = new double[n+2];
double* TD = new double[n+2];
double* TL = new double[n+2];
double* B = new double[n+2];

double* exactu = new double[n+2];
double* exactk = new double[n+2];
double* exacth = new double[n+2];

double* exactu_face = new double[n+2];
double* exactk_face = new double[n+2];
double* exacth_face = new double[n+2];

initzero(n+2,y_face);
initzero(n+2,y_);
initzero(n+2,Jfaces);
initzero(n+2,J);
initzero(n+2,dy);
initzero(n+2,u);
initzero(n+2,up);
initzero(n+2,upp);
initzero(n+2,k);
initzero(n+2,kp);
initzero(n+2,Jp);
initzero(n+2,h);
initzero(n+2,Gamma);
initzero(n+2,TU);

```

```

initzero(n+2,TD);
initzero(n+2,TL);
initzero(n+2,B);
initzero(n+2,exactu);
initzero(n+2,exactk);
initzero(n+2,exacth);
initzero(n+2,exactu_face);
initzero(n+2,exactk_face);
initzero(n+2,exacth_face);

makegrid(n,L,growth,y_face,y,Jfaces,J);
J0 = (3.0*J[1] - J[2])/2.0;

//write Fluent grid file
double dx = L/double(n);
FILE* ffile;
ffile = fopen("msh_paste.txt","w");
fprintf(ffile,"info: %d x %d grid cells\n",n,5);
fprintf(ffile,"dx = %13.14e\n",dx);
fprintf(ffile,"growth (beta) = %13.14e\n",growth);
fprintf(ffile,"length (L) = %13.14e\n",L);

fprintf(ffile,"      %13.14e      %13.14e\n",5.0*dx,y_face[0]);
fprintf(ffile,"      %13.14e      %13.14e\n",0.0,y_face[0]);
for(j = 1; j < 5; j++) fprintf(ffile,"      %13.14e
%13.14e\n",double(j)*dx,y_face[0]);
fprintf(ffile,"      %13.14e      %13.14e\n",0.0,y_face[n]);
fprintf(ffile,"      %13.14e      %13.14e\n",5.0*dx,y_face[n]);
for(j = 4; j > 0; j--) fprintf(ffile,"      %13.14e
%13.14e\n",double(j)*dx,y_face[n]);
for(j = 1; j < n; j++) fprintf(ffile,"      %13.14e      %13.14e\n",0.0,y_face[j]);
for(j = 1; j < n; j++) fprintf(ffile,"      %13.14e      %13.14e\n",5.0*dx,y_face[j]);
for(i = n-1; i > 0; i--)
{
    for(j = 1; j < 5; j++) fprintf(ffile,"      %13.14e
%13.14e\n",double(j)*dx,y_face[i]);
}
fclose(ffile);

//initialize
for(j = 0; j <= n+1; j++)
{
    exactu[j] = U_Exact(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit);
    exactk[j] = K_Exact(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit);
    exacth[j] = H_Exact(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit);
    exactu_face[j] = U_Exact(L,j,rho,nu,dpdx,y_face[j],k[j],h[j],ut,limit);
    exactk_face[j] = K_Exact(L,j,rho,nu,dpdx,y_face[j],k[j],h[j],ut,limit);
    exacth_face[j] = H_Exact(L,j,rho,nu,dpdx,y_face[j],k[j],h[j],ut,limit);

    u[j] = U_Init(L,j,rho,nu,dpdx,y_face[j],k[j],h[j],ut,limit);
    k[j] = K_Init(L,j,rho,nu,dpdx,y_face[j],k[j],h[j],ut,limit);
    h[j] = H_Init(L,j,rho,nu,dpdx,y_face[j],k[j],h[j],ut,limit);
}
if(initfile)
{
    string enter;
    pfile = fopen("rstrt_1998_100.txt","r");
    double ycur,yprev,ucur,uprev,kcur,kprev,hcur,hprev;
    fscanf(pfile,"%lf %lf %lf %lf",&ycur,&ucur,&kcur,&hcur);
    u[0] = ucur; k[0] = kcur; h[0] = hcur;
    j = 1;
    while(j<n+1)
    {
        while(ycur > y[j])
        {
            u[j] = (y[j] - yprev)/(ycur - yprev)*(ucur - uprev) + uprev;
            k[j] = (y[j] - yprev)/(ycur - yprev)*(kcur - kprev) + kprev;
            h[j] = (y[j] - yprev)/(ycur - yprev)*(hcur - hprev) + hprev;
            j++;
        }
        yprev = ycur; uprev = ucur; kprev = kcur; hprev = hcur;
        fscanf(pfile,"%lf %lf %lf %lf",&ycur,&ucur,&kcur,&hcur);
    }
    fclose(pfile);
}
u[n+1] = u[n]; k[n+1] = k[n]; h[n+1] = h[n];
do{
    for(iter = 0; iter < itermax; iter++)

```

```

{
    ulimit = ulimiter;
    klimit = klimiter;
    hlimit = hlimiter;

    //u solver
    limit = 0;
    for(j = 1; j <= n; j++)
    {
        Gamma[j] = nu + fnu(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit);
        B[j] = fu(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit)/J[j];
    }
    TL[1] = 0.0;
    TD[1] = (Gamma[2] + Gamma[1])*(J[2] + J[1])/4.0 + 9.0*(3.0*Gamma[1] -
Gamma[2])*(3.0*J[1] - J[2])/12.0;
    TU[1] = -(Gamma[2] + Gamma[1])*(J[2] + J[1])/4.0 - 1.0*(3.0*Gamma[1] -
Gamma[2])*(3.0*J[1] - J[2])/12.0;
    for(j = 2; j <= n-1; j++)
    {
        TL[j] = -(Gamma[j-1] + Gamma[j])*(J[j-1] + J[j])/4.0;
        TU[j] = -(Gamma[j+1] + Gamma[j])*(J[j+1] + J[j])/4.0;
        TD[j] = -TL[j]-TU[j];
    }
    TL[n] = -(Gamma[n-1] + Gamma[n])*(J[n-1] + J[n])/4.0;
    TD[n] = -TL[n];
    TU[n] = 0.0;

    if(Transport(n, u, alphau, omega, TU, TL, TD, B, 0, ulimit)) break;
    tw = rho*(3.0*Gamma[1] - Gamma[2])*J0/2.0*(9.0*u[1] - u[2])/3.0;
    ut = sqrt(tw/rho);

    //k solver
    for(j = 1; j <= n; j++) Gamma[j] = nu +
fnu(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit)/sigmak;
    if(forcekp) //correct no-slip boundary conditions
    {
        TL[1] = 0.0; TD[1] = 9.0; TU[1] = -1.0; B[1] =
3.0*dkpdyp*pow(ut,3)/(nu*J0);
    }
    else //incorrect no-slip boundary conditions
    {
        up[1] = (3.0*u[1] + u[2])/3.0; kp[1] = (3.0*sqrt(k[1]) + sqrt(k[2]))/3.0;
        TL[1] = 0.0;
        TD[1] = (Gamma[2] + Gamma[1])*(J[2] + J[1])/4.0 + 9.0*(3.0*Gamma[1] -
Gamma[2])*(3.0*J[1] - J[2])/12.0;
        TU[1] = -(Gamma[2] + Gamma[1])*(J[2] + J[1])/4.0 - 1.0*(3.0*Gamma[1] -
Gamma[2])*(3.0*J[1] - J[2])/12.0;
        B[1] = fk1(L,1,rho,nu,dpdx,y[1],k[1],h[1],ut,limit)*up[1]*up[1]*J[1] -
fk2(L,1,rho,nu,dpdx,y[1],k[1],h[1],ut,limit)/J[1] -
2.0*fk3(L,1,rho,nu,dpdx,y[1],k[1],h[1],ut,limit)*kp[1]*kp[1]*J[1];
    }

    for(j = 2; j <= n-1; j++)
    {
        up[j] = (u[j+1] - u[j-1])/2.0; kp[j] = (sqrt(k[j+1]) - sqrt(k[j-1]))/2.0;
        TL[j] = -(Gamma[j-1] + Gamma[j])*(J[j-1] + J[j])/4.0;
        TU[j] = -(Gamma[j+1] + Gamma[j])*(J[j+1] + J[j])/4.0;
        TD[j] = -TL[j]-TU[j];
    }
    up[n] = (u[n] - u[n-1])/2.0; kp[n] = (sqrt(k[n]) - sqrt(k[n-1]))/2.0;
    TL[n] = -(Gamma[n-1] + Gamma[n])*(J[n-1] + J[n])/4.0;
    TD[n] = -TL[n];
    TU[n] = 0.0;
    for(j = 2; j <= n; j++)
    {
        B[j] = fk1(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit)*up[j]*up[j]*J[j] -
fk2(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit)/J[j] -
2.0*fk3(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit)*kp[j]*kp[j]*J[j];
    }
    if(Transport(n, k, alphak, omega, TU, TL, TD, B, 0, klimit)) break;

    //h solver
    for(j = 1; j <= n; j++) Gamma[j] = nu +
fnu(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit)/sigmah;
    if(m_h > 0) //near-wall asymptotic solution is used
    {
        for(j = 1; j <= m_h; j++)
        {
            TL[j] = 0.0; TD[j] = 1.0; TU[j] = 0.0; B[j] = fh0(j,k,h,y,J,nu,ut);

```

```

    }
  }
  else //near-wall asymptotic solution only used directly on boundary
  {
    h[0] = fh0(0,k,h,y,J,nu,ut);
    up[1] = (3.0*u[1] + u[2])/3.0; upp[1] = (-75.0*u[1] + 30.0*u[2] -
3.0*u[3])/15.0; Jp[1] = (-4.0*J[0] + 3.0*J[1] + J[2])/3.0;
    TL[1] = 0.0;
    TD[1] = (Gamma[2] + Gamma[1])*(J[2] + J[1])/4.0 + 9.0*(3.0*Gamma[1] -
Gamma[2])*(3.0*J[1] - J[2])/12.0;
    TU[1] = -(Gamma[2] + Gamma[1])*(J[2] + J[1])/4.0 - 1.0*(3.0*Gamma[1] -
Gamma[2])*(3.0*J[1] - J[2])/12.0;
    B[1] = 8.0/12.0*(3.0*Gamma[1] - Gamma[2])*(3.0*J[1] - J[2])*h[0] +
fh1(L,1,rho,nu,dpdx,y[1],k[1],h[1],ut,limit)*up[1]*up[1]*J[1] -
fh2(L,1,rho,nu,dpdx,y[1],k[1],h[1],ut,limit)/J[1];
    B[1] += fh3(L,1,rho,nu,dpdx,y[1],k[1],h[1],ut,limit)*pow((Jp[1]*up[1] +
J[1]*upp[1]),2)*J[1];
  }

  for(j = max(m_h+1,2); j <= n-1; j++)
  {
    up[j] = (u[j+1] - u[j-1])/2.0; upp[j] = (u[j+1] - 2.0*u[j] + u[j-1]);
Jp[j] = (J[j+1] - J[j-1])/2.0;
    TL[j] = -(Gamma[j-1] + Gamma[j])*(J[j-1] + J[j])/4.0;
    TU[j] = -(Gamma[j+1] + Gamma[j])*(J[j+1] + J[j])/4.0;
    TD[j] = -TL[j]-TU[j];
  }
  up[n] = (u[n] - u[n-1])/2.0; upp[n] = (-u[n] + u[n-1]); Jp[n] = (J[n] - J[n-
1])/2.0;
  TL[n] = -(Gamma[n-1] + Gamma[n])*(J[n-1] + J[n])/4.0;
  TD[n] = -TL[n];
  TU[n] = 0.0;
  for(j = max(m_h+1,2); j <= n; j++)
  {
    B[j] = fh1(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit)*up[j]*up[j]*J[j] -
fh2(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit)/J[j];
    B[j] += fh3(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit)*pow((Jp[j]*up[j] +
J[j]*upp[j]),2)*J[j];
  }
  if(Transport(n, h, alphah, omega, TU, TL, TD, B, 0, hlimit)) break;

  //Interpolate to faces
  u_face[0] = 0.0;
  k_face[0] = 0.0;
  h_face[0] = fh0(0,k,h,y,J,nu,ut);
  for(j = 1; j < n; j++)
  {
    u_face[j] = .5*(u[j] + u[j+1]);
    k_face[j] = .5*(k[j] + k[j+1]);
    h_face[j] = .5*(h[j] + h[j+1]);
  }
  u_face[n] = (3.0*u[n] - u[n-1])/2.0;
  k_face[n] = (3.0*k[n] - k[n-1])/2.0;
  h_face[n] = (3.0*h[n] - h[n-1])/2.0;

  //Calculate Error
  urms = krms = hrms = urms_face = krms_face = hrms_face = 0.0;
  for(j = 1; j <= n; j++)
  {
    urms += pow(exactu[j] - u[j],2);
    krms += pow(exactk[j] - k[j],2);
    hrms += pow(exacth[j] - h[j],2);
  }
  for(j = 0; j <= n; j++)
  {
    urms_face += pow(exactu_face[j] - u_face[j],2);
    krms_face += pow(exactk_face[j] - k_face[j],2);
    hrms_face += pow(exacth_face[j] - h_face[j],2);
  }
  urms = sqrt(urms/n); krms = sqrt(krms/n); hrms = sqrt(hrms/n);
  urms_face = sqrt(urms_face/(n+1)); krms_face = sqrt(krms_face/(n+1));
hrms_face = sqrt(hrms_face/(n+1));
}

cout<<"writing face.txt"<<endl;
//write results to file
pfile = fopen("face.txt","w");
fprintf(pfile,"i\ty faces\tu\tk\th\ty plus\tu plus\tk plus\te plus\tw plus\tnut\n");
for(j = 0; j <= n; j++) fprintf(pfile,"%d %13.14e %13.14e %13.14e %13.14e %13.14e

```

```

%13.14e %13.14e %13.14e %13.14e %13.14e\n",j,y_face[j],u_face[j],k_face[j],h_face[j],
y_face[j]*ut/nu, u_face[j]/ut, k_face[j]/ut/ut,
h_face[j]*nu/pow(ut,4),h_face[j]*nu/ut/ut,fnu(L,j,rho,nu,dpdx,y_face[j],k_face[j],h_face[
j],ut,limit));
fclose(pfile);

cout<<"writing cell.txt"<<endl;
//write results to file
pfile = fopen("cell.txt","w");
fprintf(pfile,"i\ty_cell\tu\tk\tth\ty_plus\tu_plus\tk_plus\te_plus\tw_plus\tnut\n");
for(j = 1; j <= n; j++) fprintf(pfile,"%d %13.14e %13.14e %13.14e %13.14e %13.14e
%13.14e %13.14e %13.14e %13.14e\n",j,y[j],u[j],k[j],h[j], y[j]*ut/nu, u[j]/ut,
k[j]/ut/ut,
h[j]*nu/pow(ut,4),h[j]*nu/ut/ut,fnu(L,j,rho,nu,dpdx,y[j],k[j],h[j],ut,limit));
fclose(pfile);

cout<<"writing rstrt.txt"<<endl;
//write rstrt file
pfile = fopen("rstrt.txt","w");
for(j = 0; j <= n; j++) fprintf(pfile,"%13.14e %13.14e %13.14e
%13.14e\n",y_face[j],u_face[j],k_face[j],h_face[j]);
fclose(pfile);

printf("\n iter %d exact u = %13.14e k = %13.14e h =
%13.14e\n",iter,exactu_face[n],exactk_face[n],exacth_face[n]);
printf(" iter %d calc u = %13.14e k = %13.14e h =
%13.14e\n",iter,u_face[n],k_face[n],h_face[n]);

cout<<"error info at nodes: "<<endl;
printf("%13.14e %13.14e %13.14e %13.14e\n",double(L)/double(n),urms,krms,hrms);
cout<<"error info at faces: "<<endl;
printf("%13.14e %13.14e %13.14e
%13.14e\n",double(L)/double(n),urms_face,krms_face,hrms_face);
cout<<endl<<"tw \t\t y+(1) \t\t u+(1) \t\t k+(1) \t\t h+(1)"<<endl;
printf("%13.14e %13.14e %13.14e %13.14e %13.14e\n", tw, y_face[n]*ut/nu,
u_face[n]/ut, k_face[n]/ut/ut, h_face[n]*nu/pow(ut,4));
cout<<"enter omega ("<<omega<<") (0 -> quit):";
cin>>omega;
}while(omega > 0.0);

return 0;
}

```

APPENDIX F

COORDINATE TRANSFORMATIONS IN CARTESIAN COORDINATES

I. Transformation Definitions

Complex geometries are often modeled using a body-fitted coordinate system known as curvilinear coordinates. In two dimensions, this coordinate system is created by assuming that a computational domain (ξ, η) can be defined by a transformation of the physical domain (x, y) where

$$\begin{aligned} x &= x(\xi, \eta) \\ y &= y(\xi, \eta) \end{aligned} \quad (\text{F.1})$$

The coordinate transformation function must be continuous over the entire domain. The determinant of the Jacobian transformation matrix, referred to here as simply the Jacobian and given the symbol J , is defined as

$$J \equiv \frac{\partial(\xi, \eta)}{\partial(x, y)} = \begin{vmatrix} \xi_{,x} & \xi_{,y} \\ \eta_{,x} & \eta_{,y} \end{vmatrix} = \xi_{,x}\eta_{,y} - \xi_{,y}\eta_{,x} \quad (\text{F.2})$$

The Jacobian can be evaluated from

$$J = \frac{1}{J^{-1}} = 1 / \frac{\partial(x, y)}{\partial(\xi, \eta)} = 1 / \begin{vmatrix} x_{,\xi} & x_{,\eta} \\ y_{,\xi} & y_{,\eta} \end{vmatrix} = \frac{1}{x_{,\xi}y_{,\eta} - x_{,\eta}y_{,\xi}} \quad (\text{F.3})$$

by using second-order central differences in the computational domain. Using the chain rule, partial derivatives can be expressed as

$$\begin{aligned} \frac{\partial}{\partial x} &= \xi_{,x} \frac{\partial}{\partial \xi} + \eta_{,x} \frac{\partial}{\partial \eta} \\ \frac{\partial}{\partial y} &= \xi_{,y} \frac{\partial}{\partial \xi} + \eta_{,y} \frac{\partial}{\partial \eta} \end{aligned} \quad (\text{F.4})$$

The metrics $\xi_{,x}$, $\xi_{,y}$, $\eta_{,x}$, and $\eta_{,y}$ can be evaluated by applying the relationship

$$\begin{bmatrix} \xi_{,x} & \xi_{,y} \\ \eta_{,x} & \eta_{,y} \end{bmatrix} = \begin{bmatrix} x_{,\xi} & x_{,\eta} \\ y_{,\xi} & y_{,\eta} \end{bmatrix}^{-1} = J \begin{bmatrix} y_{,\eta} & -x_{,\eta} \\ -y_{,\xi} & x_{,\xi} \end{bmatrix} \quad (\text{F.5})$$

This gives

$$\begin{aligned}
\xi_{,x} &= Jy_{,\eta} \\
\xi_{,y} &= -Jx_{,\eta} \\
\eta_{,x} &= -Jy_{,\xi} \\
\eta_{,y} &= Jx_{,\xi}
\end{aligned}
\tag{F.6}$$

Therefore, partial derivatives of any continuously differentiable scalar, ϕ , in the physical domain can be written as partial derivatives in the computational domain according to

$$\begin{aligned}
\frac{\partial \phi}{\partial x} &= \phi_{,x} = J(y_{,\eta}\phi_{,\xi} - y_{,\xi}\phi_{,\eta}) \\
\frac{\partial \phi}{\partial y} &= \phi_{,y} = J(x_{,\xi}\phi_{,\eta} - x_{,\eta}\phi_{,\xi})
\end{aligned}
\tag{F.7}$$

In this work, the grids are confined to orthogonal, rectilinear grids. Therefore, $x = x(\xi)$, $y = y(\eta)$, and $\xi_{,y} = \eta_{,x} = 0$. Equations (F.2) and (F.3) simplify to

$$J \equiv \frac{\partial(\xi, \eta)}{\partial(x, y)} = \begin{vmatrix} \xi_{,x} & 0 \\ 0 & \eta_{,y} \end{vmatrix} = \xi_{,x}\eta_{,y}
\tag{F.8}$$

$$J = \frac{1}{J^{-1}} = 1 / \frac{\partial(x, y)}{\partial(\xi, \eta)} = 1 / \begin{vmatrix} x_{,\xi} & 0 \\ 0 & y_{,\eta} \end{vmatrix} = \frac{1}{x_{,\xi}y_{,\eta}}
\tag{F.9}$$

Partial derivatives can be expressed as

$$\begin{aligned}
\frac{\partial}{\partial x} &= \xi_{,x} \frac{\partial}{\partial \xi} \\
\frac{\partial}{\partial y} &= \eta_{,y} \frac{\partial}{\partial \eta}
\end{aligned}
\tag{F.10}$$

where

$$\begin{aligned}
\xi_{,x} &= Jy_{,\eta} \\
\eta_{,y} &= Jx_{,\xi}
\end{aligned}
\tag{F.11}$$

In this work, the contravariant velocity components are defined as

$$\begin{aligned}
\bar{V}_{\xi} &\equiv y_{,\eta} \bar{V}_x \\
\bar{V}_{\eta} &\equiv x_{,\xi} \bar{V}_y
\end{aligned}
\tag{F.12}$$

II. Scalar Transport Coordinate Transformations

A. General Scalar Transport

The general steady-state scalar transport equation is written in vector form as

$$(\bar{\mathbf{V}} \cdot \nabla)\phi = \nabla \cdot (\Gamma \nabla \phi) + S \quad (\text{F.13})$$

The two-dimensional steady-state transport equation for any scalar, ϕ , can be written in Cartesian coordinates as

$$\frac{\partial(\bar{V}_x \phi)}{\partial x} + \frac{\partial(\bar{V}_y \phi)}{\partial y} = \frac{\partial}{\partial x} \left(\Gamma \frac{\partial \phi}{\partial x} \right) + \frac{\partial}{\partial y} \left(\Gamma \frac{\partial \phi}{\partial y} \right) + S \quad (\text{F.14})$$

where Γ is the diffusion coefficient and S represents the source terms. Applying Eq. (F.10) to Eq. (F.14) gives

$$\xi_{,x} \frac{\partial(\bar{V}_x \phi)}{\partial \xi} + \eta_{,y} \frac{\partial(\bar{V}_y \phi)}{\partial \eta} = \xi_{,x} \frac{\partial}{\partial \xi} \left(\Gamma \xi_{,x} \frac{\partial \phi}{\partial \xi} \right) + \eta_{,y} \frac{\partial}{\partial \eta} \left(\Gamma \eta_{,y} \frac{\partial \phi}{\partial \eta} \right) + S \quad (\text{F.15})$$

Applying Eq. (F.11) gives

$$Jy_{,\eta} \frac{\partial(\bar{V}_x \phi)}{\partial \xi} + Jx_{,\xi} \frac{\partial(\bar{V}_y \phi)}{\partial \eta} = Jy_{,\eta} \frac{\partial}{\partial \xi} \left(\Gamma Jy_{,\eta} \frac{\partial \phi}{\partial \xi} \right) + Jx_{,\xi} \frac{\partial}{\partial \eta} \left(\Gamma Jx_{,\xi} \frac{\partial \phi}{\partial \eta} \right) + S \quad (\text{F.16})$$

Simplifying gives the two-dimensional steady-state scalar transport equation in the transformed coordinate system

$$\frac{\partial(\bar{V}_\xi \phi)}{\partial \xi} + \frac{\partial(\bar{V}_\eta \phi)}{\partial \eta} = y_{,\eta}^2 \frac{\partial}{\partial \xi} \left(\Gamma J \frac{\partial \phi}{\partial \xi} \right) + x_{,\xi}^2 \frac{\partial}{\partial \eta} \left(\Gamma J \frac{\partial \phi}{\partial \eta} \right) + \frac{S}{J} \quad (\text{F.17})$$

B. Source Terms

Each transport equation has its own definition for a transport property, ϕ , diffusion coefficient, Γ , and source term, S . Table F.1 shows these definitions in Cartesian coordinates for the transport equations used in this research. The source term in many of the turbulence models includes the squared magnitude of the strain-rate tensor. For conciseness, this magnitude is given the symbol S_V^2 and is defined as

$$S_V^2 \equiv \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \quad (\text{F.18})$$

Table F.1. Transport equation terms

Transport Equation	ϕ	Γ	S
Continuity	1	0	0
X-Momentum	\bar{V}_x	$\nu + \nu_t$	$-\frac{1}{\rho} \frac{\partial \hat{p}}{\partial x} + \frac{\partial}{\partial x} \left(\Gamma \frac{\partial \bar{V}_x}{\partial x} \right) + \frac{\partial}{\partial y} \left(\Gamma \frac{\partial \bar{V}_y}{\partial x} \right)$
Y-Momentum	\bar{V}_y	$\nu + \nu_t$	$-\frac{1}{\rho} \frac{\partial \hat{p}}{\partial y} + \frac{\partial}{\partial y} \left(\Gamma \frac{\partial \bar{V}_y}{\partial y} \right) + \frac{\partial}{\partial x} \left(\Gamma \frac{\partial \bar{V}_x}{\partial y} \right)$
Kinetic Energy	k	$\nu + \nu_t / \sigma_k$	$f_{k1} S_V^2 - f_{k2}$
Dissipation	ε	$\nu + \nu_t / \sigma_\varepsilon$	$f_{\varepsilon 1} S_V^2 - f_{\varepsilon 2}$
Dissipation Frequency	ω	$\nu + \nu_t / \sigma_\omega$	$f_{\omega 1} S_V^2 - f_{\omega 2}$

The source term in some turbulence models includes the squared magnitude of the rotation tensor. For conciseness, this magnitude is given the symbol Ω_V^2 and is defined as

$$\Omega_V^2 \equiv \bar{\bar{\Omega}}(\mathbf{V}) \cdot \bar{\bar{\Omega}}(\mathbf{V}) \quad (\text{F.19})$$

All derivatives in the source terms must be converted from Cartesian coordinates to the transformed coordinate system. These transformations are shown in this section.

1. Continuity

Applying the continuity equation definitions given in Table F.1 to Eq. (F.17) gives the continuity equation in curvilinear coordinates

$$\frac{\partial \bar{V}_\xi}{\partial \xi} + \frac{\partial \bar{V}_\eta}{\partial \eta} = 0 \quad (\text{F.20})$$

2. x -Momentum

Applying Eqs. (F.10) and (F.11) to the x -momentum source terms gives

$$\begin{aligned}
 S_x &= -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial x} + \frac{\partial}{\partial x} \left(\Gamma \frac{\partial \bar{V}_x}{\partial x} \right) + \frac{\partial}{\partial y} \left(\Gamma \frac{\partial \bar{V}_y}{\partial x} \right) \\
 &= -\xi_{,x} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \xi} + \xi_{,x} \frac{\partial}{\partial \xi} \left(\Gamma \xi_{,x} \frac{\partial \bar{V}_x}{\partial \xi} \right) + \eta_{,y} \frac{\partial}{\partial \eta} \left(\Gamma \xi_{,x} \frac{\partial \bar{V}_y}{\partial \xi} \right) \\
 &= -J_{y,\eta} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \xi} + J_{y,\eta} \frac{\partial}{\partial \xi} \left(\Gamma J_{y,\eta} \frac{\partial \bar{V}_x}{\partial \xi} \right) + J_{x,\xi} \frac{\partial}{\partial \eta} \left(\Gamma \xi_{,x} \frac{\partial \bar{V}_y}{\partial \xi} \right) \\
 &= -J_{y,\eta} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \xi} + J_{y,\eta}^2 \frac{\partial}{\partial \xi} \left(\Gamma J \frac{\partial \bar{V}_x}{\partial \xi} \right) + J \frac{\partial}{\partial \eta} \left(\Gamma \frac{\partial \bar{V}_y}{\partial \xi} \right)
 \end{aligned} \tag{F.21}$$

3. y -Momentum

Applying Eqs. (F.10) and (F.11) to the y -momentum source terms gives

$$\begin{aligned}
 S_y &= -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial y} + \frac{\partial}{\partial y} \left(\Gamma \frac{\partial \bar{V}_y}{\partial y} \right) + \frac{\partial}{\partial x} \left(\Gamma \frac{\partial \bar{V}_x}{\partial y} \right) \\
 &= -\eta_{,y} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \eta} + \eta_{,y} \frac{\partial}{\partial \eta} \left(\Gamma \eta_{,y} \frac{\partial \bar{V}_y}{\partial \eta} \right) + \xi_{,x} \frac{\partial}{\partial \xi} \left(\Gamma \eta_{,y} \frac{\partial \bar{V}_x}{\partial \eta} \right) \\
 &= -J_{x,\xi} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \eta} + J_{x,\xi} \frac{\partial}{\partial \eta} \left(\Gamma J_{x,\xi} \frac{\partial \bar{V}_y}{\partial \eta} \right) + J_{x,\xi} \xi_{,x} \frac{\partial}{\partial \xi} \left(\Gamma \frac{\partial \bar{V}_x}{\partial \eta} \right) \\
 &= -J_{x,\xi} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \eta} + J_{x,\xi}^2 \frac{\partial}{\partial \eta} \left(\Gamma J \frac{\partial \bar{V}_y}{\partial \eta} \right) + J \frac{\partial}{\partial \xi} \left(\Gamma \frac{\partial \bar{V}_x}{\partial \eta} \right)
 \end{aligned} \tag{F.22}$$

4. Strain-Rate Tensor

The strain-rate tensor can be written for two-dimensional flow in Cartesian coordinates as

$$S_V^2 \equiv \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) \cdot \bar{\bar{\mathbf{S}}}(\bar{\mathbf{V}}) = \left(\frac{\partial \bar{V}_x}{\partial x} \right)^2 + \left(\frac{\partial \bar{V}_y}{\partial y} \right)^2 + \frac{1}{2} \left(\frac{\partial \bar{V}_x}{\partial y} + \frac{\partial \bar{V}_y}{\partial x} \right)^2 \tag{F.23}$$

Applying Eqs. (F.10) and (F.11) gives

$$\begin{aligned}
S_V^2 &= \left(\frac{\partial \bar{V}_x}{\partial x} \right)^2 + \left(\frac{\partial \bar{V}_y}{\partial y} \right)^2 + \frac{1}{2} \left(\frac{\partial \bar{V}_x}{\partial y} + \frac{\partial \bar{V}_y}{\partial x} \right)^2 \\
&= \left(\xi_{,x} \frac{\partial \bar{V}_x}{\partial \xi} \right)^2 + \left(\eta_{,y} \frac{\partial \bar{V}_y}{\partial \eta} \right)^2 + \frac{1}{2} \left(\eta_{,y} \frac{\partial \bar{V}_x}{\partial \eta} + \xi_{,x} \frac{\partial \bar{V}_y}{\partial \xi} \right)^2 \\
&= \left(J_{y,\eta} \frac{\partial \bar{V}_x}{\partial \xi} \right)^2 + \left(J_{x,\xi} \frac{\partial \bar{V}_y}{\partial \eta} \right)^2 + \frac{1}{2} \left(J_{x,\xi} \frac{\partial \bar{V}_x}{\partial \eta} + J_{y,\eta} \frac{\partial \bar{V}_y}{\partial \xi} \right)^2
\end{aligned} \tag{F.24}$$

5. Rotation Tensor

The rotation tensor can be written for two-dimensional flow in Cartesian coordinates as

$$\Omega_V^2 \equiv \bar{\bar{\Omega}}(\mathbf{V}) \cdot \bar{\bar{\Omega}}(\mathbf{V}) = \frac{1}{2} \left(\frac{\partial \bar{V}_y}{\partial x} - \frac{\partial \bar{V}_x}{\partial y} \right)^2 \tag{F.25}$$

Applying Eqs. (F.10) and (F.11) gives

$$\begin{aligned}
\Omega_V^2 &= \frac{1}{2} \left(\frac{\partial \bar{V}_y}{\partial x} - \frac{\partial \bar{V}_x}{\partial y} \right)^2 \\
&= \frac{1}{2} \left(\xi_{,x} \frac{\partial \bar{V}_y}{\partial \xi} - \eta_{,y} \frac{\partial \bar{V}_x}{\partial \eta} \right)^2 \\
&= \frac{1}{2} \left(J_{y,\eta} \frac{\partial \bar{V}_y}{\partial \xi} - J_{x,\xi} \frac{\partial \bar{V}_x}{\partial \eta} \right)^2
\end{aligned} \tag{F.26}$$

APPENDIX G

COORDINATE TRANSFORMATIONS IN CYLINDRICAL COORDINATES

I. Transformation Definitions

Complex geometries are often modeled using a body-fitted coordinate system known as curvilinear coordinates. For axisymmetric flow in cylindrical coordinates, this coordinate system is created by assuming that a computational domain (ξ, η) can be defined by a transformation of the physical domain (z, r) where

$$\begin{aligned} z &= z(\xi, \eta) \\ r &= r(\xi, \eta) \end{aligned} \quad (\text{G.1})$$

The coordinate transformation function must be continuous over the entire domain. The determinant of the Jacobian transformation matrix, referred to here as simply the Jacobian and given the symbol J , is defined as

$$J \equiv \frac{\partial(\xi, \eta)}{\partial(z, r)} = \begin{vmatrix} \xi_{,z} & \xi_{,r} \\ \eta_{,z} & \eta_{,r} \end{vmatrix} = \xi_{,z}\eta_{,r} - \xi_{,r}\eta_{,z} \quad (\text{G.2})$$

The Jacobian can be evaluated from

$$J = \frac{1}{J^{-1}} = 1 \left/ \frac{\partial(z, r)}{\partial(\xi, \eta)} \right. = 1 \left/ \begin{vmatrix} z_{,\xi} & z_{,\eta} \\ r_{,\xi} & r_{,\eta} \end{vmatrix} \right. = \frac{1}{z_{,\xi}r_{,\eta} - z_{,\eta}r_{,\xi}} \quad (\text{G.3})$$

by using second-order central differences in the computational domain. Using the chain rule, partial derivatives can be expressed as

$$\begin{aligned} \frac{\partial}{\partial z} &= \xi_{,z} \frac{\partial}{\partial \xi} + \eta_{,z} \frac{\partial}{\partial \eta} \\ \frac{\partial}{\partial r} &= \xi_{,r} \frac{\partial}{\partial \xi} + \eta_{,r} \frac{\partial}{\partial \eta} \end{aligned} \quad (\text{G.4})$$

The metrics $\xi_{,z}$, $\xi_{,r}$, $\eta_{,z}$, and $\eta_{,r}$ can be evaluated by applying the relationship

$$\begin{bmatrix} \xi_{,z} & \xi_{,r} \\ \eta_{,z} & \eta_{,r} \end{bmatrix} = \begin{bmatrix} z_{,\xi} & z_{,\eta} \\ r_{,\xi} & r_{,\eta} \end{bmatrix}^{-1} = J \begin{bmatrix} r_{,\eta} & -z_{,\eta} \\ -r_{,\xi} & z_{,\xi} \end{bmatrix} \quad (\text{G.5})$$

This gives

$$\begin{aligned}
 \xi_{,z} &= Jr_{,\eta} \\
 \xi_{,r} &= -Jz_{,\eta} \\
 \eta_{,z} &= -Jr_{,\xi} \\
 \eta_{,r} &= Jz_{,\xi}
 \end{aligned} \tag{G.6}$$

Therefore, partial derivatives of any continuously differentiable scalar, ϕ , in the physical domain can be written as partial derivatives in the computational domain according to

$$\begin{aligned}
 \frac{\partial \phi}{\partial z} &= \phi_{,z} = J(r_{,\eta}\phi_{,\xi} - r_{,\xi}\phi_{,\eta}) \\
 \frac{\partial \phi}{\partial r} &= \phi_{,r} = J(z_{,\xi}\phi_{,\eta} - z_{,\eta}\phi_{,\xi})
 \end{aligned} \tag{G.7}$$

In this work, the grids are confined to orthogonal, rectilinear grids. Therefore, $z = z(\xi)$, $r = r(\eta)$, and $\xi_{,r} = \eta_{,z} = 0$. Equations (G.2) and (G.3) simplify to

$$J \equiv \frac{\partial(\xi, \eta)}{\partial(z, r)} = \begin{vmatrix} \xi_{,z} & 0 \\ 0 & \eta_{,r} \end{vmatrix} = \xi_{,z}\eta_{,r} \tag{G.8}$$

$$J = \frac{1}{J^{-1}} = 1 / \left/ \frac{\partial(z, r)}{\partial(\xi, \eta)} \right/ = 1 / \begin{vmatrix} z_{,\xi} & 0 \\ 0 & r_{,\eta} \end{vmatrix} = \frac{1}{z_{,\xi}r_{,\eta}} \tag{G.9}$$

Partial derivatives can be expressed as

$$\begin{aligned}
 \frac{\partial}{\partial z} &= \xi_{,z} \frac{\partial}{\partial \xi} \\
 \frac{\partial}{\partial r} &= \eta_{,r} \frac{\partial}{\partial \eta}
 \end{aligned} \tag{G.10}$$

where

$$\begin{aligned}
 \xi_{,z} &= Jr_{,\eta} \\
 \eta_{,r} &= Jz_{,\xi}
 \end{aligned} \tag{G.11}$$

In this work, the contravariant velocity components are defined as

$$\begin{aligned}
 \bar{V}_{\xi} &\equiv r_{,\eta} \bar{V}_z \\
 \bar{V}_{\eta} &\equiv z_{,\xi} \bar{V}_r
 \end{aligned} \tag{G.12}$$

II. Scalar Transport Coordinate Transformations

A. General Scalar Transport

The general steady-state scalar transport equation is written in vector form as

$$(\bar{\mathbf{V}} \cdot \nabla)\phi = \nabla \cdot (\Gamma \nabla \phi) + S \quad (\text{G.13})$$

where Γ is the diffusion coefficient and S represents the source terms. The steady-state transport equation for any scalar, ϕ , can be written in cylindrical coordinates as

$$\bar{V}_r \frac{\partial \phi}{\partial r} + \frac{\bar{V}_\theta}{r} \frac{\partial \phi}{\partial \theta} + \bar{V}_z \frac{\partial \phi}{\partial z} = \frac{1}{r} \frac{\partial}{\partial r} \left(r \Gamma \frac{\partial \phi}{\partial r} \right) + \frac{1}{r} \frac{\partial}{\partial \theta} \left(\Gamma \frac{\partial \phi}{\partial \theta} \right) + \frac{\partial}{\partial z} \left(\Gamma \frac{\partial \phi}{\partial z} \right) + S \quad (\text{G.14})$$

For axisymmetric flow, $\bar{V}_\theta = 0$, and Eq. (G.14) reduces to

$$\bar{V}_r \frac{\partial \phi}{\partial r} + \bar{V}_z \frac{\partial \phi}{\partial z} = \frac{1}{r} \frac{\partial}{\partial r} \left(r \Gamma \frac{\partial \phi}{\partial r} \right) + \frac{\partial}{\partial z} \left(\Gamma \frac{\partial \phi}{\partial z} \right) + S \quad (\text{G.15})$$

Applying Eq. (G.10) to Eq. (G.15) gives

$$\bar{V}_r \eta_{,r} \frac{\partial \phi}{\partial \eta} + \bar{V}_z \xi_{,z} \frac{\partial \phi}{\partial \xi} = \frac{1}{r} \eta_{,r} \frac{\partial}{\partial \eta} \left(r \Gamma \eta_{,r} \frac{\partial \phi}{\partial \eta} \right) + \xi_{,z} \frac{\partial}{\partial \xi} \left(\Gamma \xi_{,z} \frac{\partial \phi}{\partial \xi} \right) + S \quad (\text{G.16})$$

Applying Eq. (G.11) gives

$$\bar{V}_r J_{z,\xi} \frac{\partial \phi}{\partial \eta} + \bar{V}_z J_{r,\eta} \frac{\partial \phi}{\partial \xi} = \frac{1}{r} J_{z,\xi} \frac{\partial}{\partial \eta} \left(r \Gamma J_{z,\xi} \frac{\partial \phi}{\partial \eta} \right) + J_{r,\eta} \frac{\partial}{\partial \xi} \left(\Gamma J_{r,\eta} \frac{\partial \phi}{\partial \xi} \right) + S \quad (\text{G.17})$$

Simplifying gives the two-dimensional steady-state transport equation in the transformed coordinate system

$$\bar{V}_\eta \frac{\partial \phi}{\partial \eta} + \bar{V}_\xi \frac{\partial \phi}{\partial \xi} = z_{,\xi}^2 \frac{1}{r} \frac{\partial}{\partial \eta} \left(r \Gamma J \frac{\partial \phi}{\partial \eta} \right) + r_{,\eta}^2 \frac{\partial}{\partial \xi} \left(\Gamma J \frac{\partial \phi}{\partial \xi} \right) + \frac{S}{J} \quad (\text{G.18})$$

Note that the left-hand side of Eq. (G.13) can be written according to the identity

$$(\bar{\mathbf{V}} \cdot \nabla)\phi = \nabla \cdot (\phi \bar{\mathbf{V}}) - \phi (\nabla \cdot \bar{\mathbf{V}}) \quad (\text{G.13})$$

Applying continuity to this expression gives an alternate form for the right-hand side of Eq. (G.13)

$$\nabla \cdot (\phi \bar{\mathbf{V}}) - \phi (\nabla \cdot \bar{\mathbf{V}}) = \frac{1}{r} \frac{\partial (r \bar{V}_r \phi)}{\partial r} + \frac{1}{r} \frac{\partial (\bar{V}_\theta \phi)}{\partial \theta} + \frac{\partial (\bar{V}_z \phi)}{\partial z} - \phi \frac{1}{r} \frac{\partial (r \bar{V}_r)}{\partial r} - \phi \frac{1}{r} \frac{\partial \bar{V}_\theta}{\partial \theta} - \phi \frac{\partial \bar{V}_z}{\partial z} \quad (\text{G.13})$$

Assuming axisymmetric flow

$$\nabla \cdot (\phi \bar{\mathbf{V}}) - \phi (\nabla \cdot \bar{\mathbf{V}}) = \frac{1}{r} \frac{\partial (r \bar{V}_r \phi)}{\partial r} + \frac{\partial (\bar{V}_z \phi)}{\partial z} - \phi \frac{1}{r} \frac{\partial (r \bar{V}_r)}{\partial r} - \phi \frac{\partial \bar{V}_z}{\partial z} \quad (\text{G.13})$$

Using the chain rule for the first two terms on the left-hand side gives

$$\nabla \cdot (\phi \bar{\mathbf{V}}) - \phi (\nabla \cdot \bar{\mathbf{V}}) = \phi \frac{1}{r} \frac{\partial (r \bar{V}_r)}{\partial r} + \phi \frac{\partial (\bar{V}_z)}{\partial z} + r \bar{V}_r \frac{1}{r} \frac{\partial \phi}{\partial r} + \bar{V}_z \frac{\partial \phi}{\partial z} - \phi \frac{1}{r} \frac{\partial (r \bar{V}_r)}{\partial r} - \phi \frac{\partial \bar{V}_z}{\partial z} \quad (\text{G.13})$$

Simplifying gives

$$\nabla \cdot (\phi \bar{\mathbf{V}}) - \phi (\nabla \cdot \bar{\mathbf{V}}) = \bar{V}_r \frac{\partial \phi}{\partial r} + \bar{V}_z \frac{\partial \phi}{\partial z} \quad (\text{G.13})$$

B. Source Terms

1. Continuity

The continuity equation can be written in cylindrical coordinates as

$$\frac{1}{r} \frac{\partial (r \bar{V}_r)}{\partial r} + \frac{1}{r} \frac{\partial \bar{V}_\theta}{\partial \theta} + \frac{\partial \bar{V}_z}{\partial z} = 0 \quad (\text{G.19})$$

For axisymmetric flow, this reduces to

$$\frac{1}{r} \frac{\partial (r \bar{V}_r)}{\partial r} + \frac{\partial \bar{V}_z}{\partial z} = 0 \quad (\text{G.20})$$

Applying Eqs. (G.10), (G.11), and (G.12) gives

$$\begin{aligned} \frac{1}{r} \eta_{,r} \frac{\partial (r \bar{V}_r)}{\partial \eta} + \xi_{,z} \frac{\partial \bar{V}_z}{\partial \xi} &= 0 \\ \frac{1}{r} J_{z,\xi} \frac{\partial (r \bar{V}_r)}{\partial \eta} + J_{r,\eta} \frac{\partial \bar{V}_z}{\partial \xi} &= 0 \\ \frac{1}{r} J \frac{\partial (z_{,\xi} r \bar{V}_r)}{\partial \eta} + J \frac{\partial (r_{,\eta} \bar{V}_z)}{\partial \xi} &= 0 \\ \frac{1}{r} \frac{\partial (r \bar{V}_\eta)}{\partial \eta} + \frac{\partial \bar{V}_\xi}{\partial \xi} &= 0 \end{aligned} \quad (\text{G.21})$$

2. Momentum

The steady-state RANS equation is a vector equation and can be written in cylindrical coordinates as

$$\begin{aligned}
\bar{V}_r \frac{\partial \bar{V}_r}{\partial r} + \frac{\bar{V}_\theta}{r} \frac{\partial \bar{V}_r}{\partial \theta} + \bar{V}_z \frac{\partial \bar{V}_r}{\partial z} - \frac{\bar{V}_\theta^2}{r} = & -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial r} + \frac{1}{r} \frac{\partial}{\partial r} \left[2(v + v_t) r \frac{\partial \bar{V}_r}{\partial r} \right] \\
& + \frac{1}{r^2} \frac{\partial}{\partial \theta} \left[(v + v_t) \left(r^2 \frac{\partial (\bar{V}_\theta/r)}{\partial r} + \frac{\partial \bar{V}_r}{\partial \theta} \right) \right] \\
& + \frac{\partial}{\partial z} \left[(v + v_t) \left(\frac{\partial \bar{V}_z}{\partial r} + \frac{\partial \bar{V}_r}{\partial z} \right) \right] - \frac{2(v + v_t)}{r^2} \left(\frac{\partial \bar{V}_\theta}{\partial \theta} + \bar{V}_r \right)
\end{aligned} \tag{G.22}$$

$$\begin{aligned}
\bar{V}_r \frac{\partial \bar{V}_\theta}{\partial r} + \frac{\bar{V}_\theta}{r} \frac{\partial \bar{V}_\theta}{\partial \theta} + \bar{V}_z \frac{\partial \bar{V}_\theta}{\partial z} + \frac{\bar{V}_r \bar{V}_\theta}{r} = & -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial \theta} + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t) r \left(\frac{1}{r} \frac{\partial \bar{V}_r}{\partial \theta} + r \frac{\partial (\bar{V}_\theta/r)}{\partial r} \right) \right] \\
& + \frac{1}{r} \frac{\partial}{\partial \theta} \left[2(v + v_t) \left(\frac{1}{r} \frac{\partial \bar{V}_\theta}{\partial \theta} + \frac{\bar{V}_r}{r} \right) \right] \\
& + \frac{\partial}{\partial z} \left[(v + v_t) \left(\frac{1}{r} \frac{\partial \bar{V}_z}{\partial \theta} + \frac{\partial \bar{V}_\theta}{\partial z} \right) \right] + \frac{(v + v_t)}{r^2} \left(\frac{\partial \bar{V}_r}{\partial \theta} + r^2 \frac{\partial (\bar{V}_\theta/r)}{\partial r} \right)
\end{aligned} \tag{G.23}$$

$$\begin{aligned}
\bar{V}_r \frac{\partial \bar{V}_z}{\partial r} + \frac{\bar{V}_\theta}{r} \frac{\partial \bar{V}_z}{\partial \theta} + \bar{V}_z \frac{\partial \bar{V}_z}{\partial z} = & -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial z} + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t) r \left(\frac{\partial \bar{V}_r}{\partial z} + \frac{\partial \bar{V}_z}{\partial r} \right) \right] \\
& + \frac{1}{r} \frac{\partial}{\partial \theta} \left[(v + v_t) \left(\frac{\partial \bar{V}_\theta}{\partial z} + \frac{1}{r} \frac{\partial \bar{V}_z}{\partial \theta} \right) \right] + \frac{\partial}{\partial z} \left[2(v + v_t) \left(\frac{\partial \bar{V}_z}{\partial z} \right) \right]
\end{aligned} \tag{G.24}$$

For axisymmetric flow, these equations reduce to only two equations

$$\bar{V}_r \frac{\partial \bar{V}_r}{\partial r} + \bar{V}_z \frac{\partial \bar{V}_r}{\partial z} = -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial r} + \frac{1}{r} \frac{\partial}{\partial r} \left[2(v + v_t) r \frac{\partial \bar{V}_r}{\partial r} \right] + \frac{\partial}{\partial z} \left[(v + v_t) \left(\frac{\partial \bar{V}_z}{\partial r} + \frac{\partial \bar{V}_r}{\partial z} \right) \right] - \frac{2(v + v_t)}{r^2} (\bar{V}_r) \tag{G.25}$$

$$\bar{V}_r \frac{\partial \bar{V}_z}{\partial r} + \bar{V}_z \frac{\partial \bar{V}_z}{\partial z} = -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial z} + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t) r \left(\frac{\partial \bar{V}_r}{\partial z} + \frac{\partial \bar{V}_z}{\partial r} \right) \right] + \frac{\partial}{\partial z} \left[2(v + v_t) \left(\frac{\partial \bar{V}_z}{\partial z} \right) \right] \tag{G.26}$$

These can be written in the form of Eq. (G.15) to yield

$$\bar{V}_r \frac{\partial \bar{V}_r}{\partial r} + \bar{V}_z \frac{\partial \bar{V}_r}{\partial z} = \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t) r \frac{\partial \bar{V}_r}{\partial r} \right] + \frac{\partial}{\partial z} \left[(v + v_t) \frac{\partial \bar{V}_r}{\partial z} \right] + S_r \tag{G.27}$$

$$\bar{V}_r \frac{\partial \bar{V}_z}{\partial r} + \bar{V}_z \frac{\partial \bar{V}_z}{\partial z} = \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t) r \frac{\partial \bar{V}_z}{\partial r} \right] + \frac{\partial}{\partial z} \left[(v + v_t) \frac{\partial \bar{V}_z}{\partial z} \right] + S_z \tag{G.28}$$

where

$$\begin{aligned}
S_r = & -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial r} + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t) r \frac{\partial \bar{V}_r}{\partial r} \right] + \frac{\partial}{\partial z} \left[(v + v_t) \frac{\partial \bar{V}_z}{\partial r} \right] - \frac{2(v + v_t)}{r^2} (\bar{V}_r) \\
S_z = & -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial z} + \frac{1}{r} \frac{\partial}{\partial r} \left[(v + v_t) r \frac{\partial \bar{V}_r}{\partial z} \right] + \frac{\partial}{\partial z} \left[(v + v_t) \frac{\partial \bar{V}_z}{\partial z} \right]
\end{aligned} \tag{G.29}$$

Applying Eqs. (G.10) and (G.11) to the momentum source terms gives

$$\begin{aligned}
S_r &= -\frac{1}{\rho} J_{z,\xi} \frac{\partial \hat{p}}{\partial \eta} + \frac{1}{r} J_{z,\xi} \frac{\partial}{\partial \eta} \left[(v + v_t) r J_{z,\xi} \frac{\partial \bar{V}_r}{\partial \eta} \right] + J_{r,\eta} \frac{\partial}{\partial \xi} \left[(v + v_t) J_{z,\xi} \frac{\partial \bar{V}_z}{\partial \eta} \right] - \frac{2(v + v_t)}{r^2} (\bar{V}_r) \\
&= -J_{z,\xi} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \eta} + \frac{1}{r} J_{z,\xi}^2 \frac{\partial}{\partial \eta} \left[(v + v_t) r J \frac{\partial \bar{V}_r}{\partial \eta} \right] + J_{r,\eta} \frac{\partial}{\partial \xi} \left[(v + v_t) \eta_{,r} \frac{\partial \bar{V}_z}{\partial \eta} \right] - \frac{2(v + v_t)}{r^2} (\bar{V}_r) \quad (G.30) \\
&= -J_{z,\xi} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \eta} + \frac{1}{r} J_{z,\xi}^2 \frac{\partial}{\partial \eta} \left[(v + v_t) r J \frac{\partial \bar{V}_r}{\partial \eta} \right] + J \frac{\partial}{\partial \xi} \left[(v + v_t) \frac{\partial \bar{V}_z}{\partial \eta} \right] - \frac{2(v + v_t)}{r^2} (\bar{V}_r)
\end{aligned}$$

$$\begin{aligned}
S_z &= -\frac{1}{\rho} J_{r,\eta} \frac{\partial \hat{p}}{\partial \xi} + \frac{1}{r} J_{z,\xi} \frac{\partial}{\partial \eta} \left[(v + v_t) r J_{r,\eta} \frac{\partial \bar{V}_r}{\partial \xi} \right] + J_{r,\eta} \frac{\partial}{\partial \xi} \left[(v + v_t) J_{r,\eta} \frac{\partial \bar{V}_z}{\partial \xi} \right] \\
&= -J_{r,\eta} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \xi} + \frac{1}{r} J_{z,\xi} \frac{\partial}{\partial \eta} \left[(v + v_t) r \xi_{,z} \frac{\partial \bar{V}_r}{\partial \xi} \right] + J_{r,\eta}^2 \frac{\partial}{\partial \xi} \left[(v + v_t) J \frac{\partial \bar{V}_z}{\partial \xi} \right] \quad (G.31) \\
&= -J_{r,\eta} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \xi} + \frac{1}{r} J \frac{\partial}{\partial \eta} \left[(v + v_t) r \frac{\partial \bar{V}_r}{\partial \xi} \right] + J_{r,\eta}^2 \frac{\partial}{\partial \xi} \left[(v + v_t) J \frac{\partial \bar{V}_z}{\partial \xi} \right]
\end{aligned}$$

3. Strain-Rate Tensor

The squared magnitude of the strain-rate tensor is given the symbol $S_{\bar{V}}^2$ and is defined as

$$S_{\bar{V}}^2 \equiv \bar{\mathbf{S}}(\bar{\mathbf{V}}) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}}) \quad (G.32)$$

The squared magnitude of the strain-rate tensor can be written in cylindrical coordinates as

$$\begin{aligned}
S_{\bar{V}}^2 &= \left(\frac{\partial \bar{V}_r}{\partial r} \right)^2 + \left(\frac{1}{r} \frac{\partial \bar{V}_\theta}{\partial \theta} + \frac{\bar{V}_r}{r} \right)^2 + \left(\frac{\partial \bar{V}_z}{\partial z} \right)^2 \\
&\quad + \frac{1}{2} \left(r \frac{\partial (\bar{V}_\theta / r)}{\partial r} + \frac{1}{r} \frac{\partial \bar{V}_r}{\partial \theta} \right)^2 + \frac{1}{2} \left(\frac{1}{r} \frac{\partial \bar{V}_z}{\partial \theta} + \frac{\partial \bar{V}_\theta}{\partial z} \right)^2 + \frac{1}{2} \left(\frac{\partial \bar{V}_r}{\partial z} + \frac{\partial \bar{V}_z}{\partial r} \right)^2 \quad (G.33)
\end{aligned}$$

For axisymmetric flow, this can be written as

$$S_{\bar{V}}^2 = \left(\frac{\partial \bar{V}_r}{\partial r} \right)^2 + \left(\frac{\bar{V}_r}{r} \right)^2 + \left(\frac{\partial \bar{V}_z}{\partial z} \right)^2 + \frac{1}{2} \left(\frac{\partial \bar{V}_r}{\partial z} + \frac{\partial \bar{V}_z}{\partial r} \right)^2 \quad (G.34)$$

Applying Eqs. (G.10) and (G.11) gives

$$\begin{aligned}
S_{\bar{V}}^2 &= \left(\eta_{,r} \frac{\partial \bar{V}_r}{\partial \eta} \right)^2 + \left(\frac{\bar{V}_r}{r} \right)^2 + \left(\xi_{,z} \frac{\partial \bar{V}_z}{\partial \xi} \right)^2 + \frac{1}{2} \left(\xi_{,z} \frac{\partial \bar{V}_r}{\partial \xi} + \eta_{,r} \frac{\partial \bar{V}_z}{\partial \eta} \right)^2 \\
&= \left(J_{z,\xi} \frac{\partial \bar{V}_r}{\partial \eta} \right)^2 + \left(\frac{\bar{V}_r}{r} \right)^2 + \left(J_{r,\eta} \frac{\partial \bar{V}_z}{\partial \xi} \right)^2 + \frac{1}{2} \left(J_{r,\eta} \frac{\partial \bar{V}_r}{\partial \xi} + J_{z,\xi} \frac{\partial \bar{V}_z}{\partial \eta} \right)^2 \quad (G.35)
\end{aligned}$$

APPENDIX H

FINITE-VOLUME DISCRETIZATION

I. Domain Discretization

We wish to solve transport equations written in the computational coordinate system using the finite-volume method. This method requires the domain to be discretized into a finite number of control volumes which will be referred to as cells. These cells may vary in size in the physical domain, but the reader is reminded that the definition of the coordinate transformation requires the transformation function to be continuous. In order to facilitate the second-order central difference scheme, uniform cell spacing in the computational domain is chosen. Because the cell size is arbitrary, for simplicity it is set to $\Delta\xi = \Delta\eta = 1.0$. Each cell has a node P placed at the cell center in the computational domain at which the values of the dependent variables will be evaluated. Additionally, each cell has a north, south, east, and west faces denoted by points n , s , e , and w . Neighboring cell centers are denoted here as nodes N , S , E , and W respectively. Figures H.1 and H.2 show the relationship between the physical domain and computational domain as well as the cell nomenclature. Figures H.3 and H.4 show the physical and computational domains of an arbitrary rectilinear domain with logarithmic grid spacing. The points in the figure represent the cell nodes and are centered in the computational domain. Note that the centers of the cells in the computational domain do not correspond to centers of the cells in the physical domain.

Nodes are also placed along the boundaries of the domain at which boundary conditions are applied. The boundary nodes can be thought of as cells with a cell center directly on the boundary and a cell area of zero. Therefore, for cells adjacent to a boundary, the value at the boundary node is equal to the value at the face of the cell. For example, for a cell adjacent to the north boundary, $\phi_N = \phi_n$. Thus the nomenclature N and n can be used interchangeably for a cell adjacent to the north boundary.

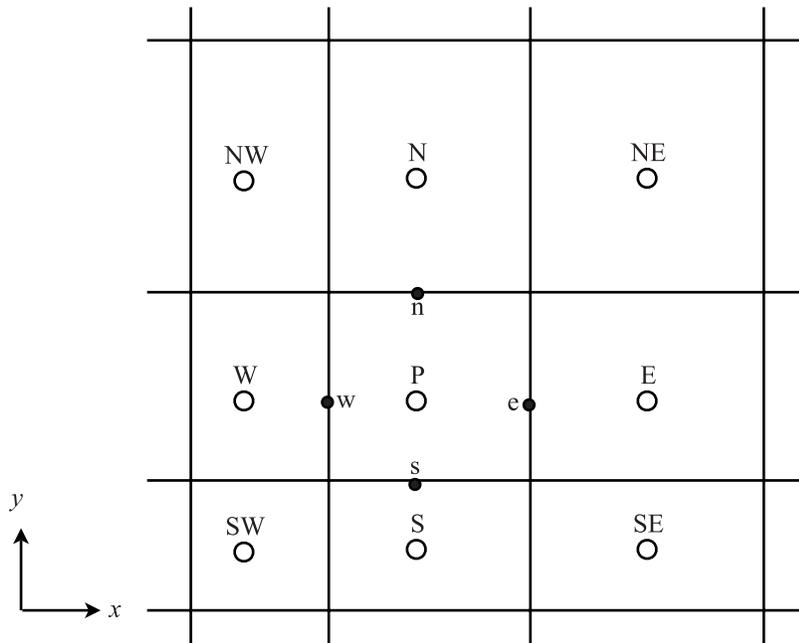


Fig. H.1 Physical domain discretization and nomenclature.

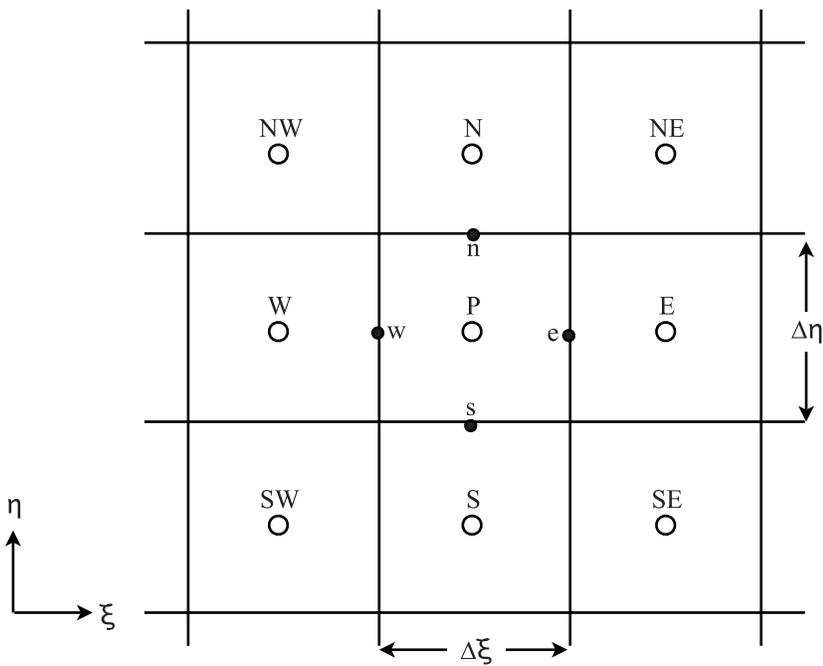


Fig. H.2 Computational domain discretization and nomenclature.

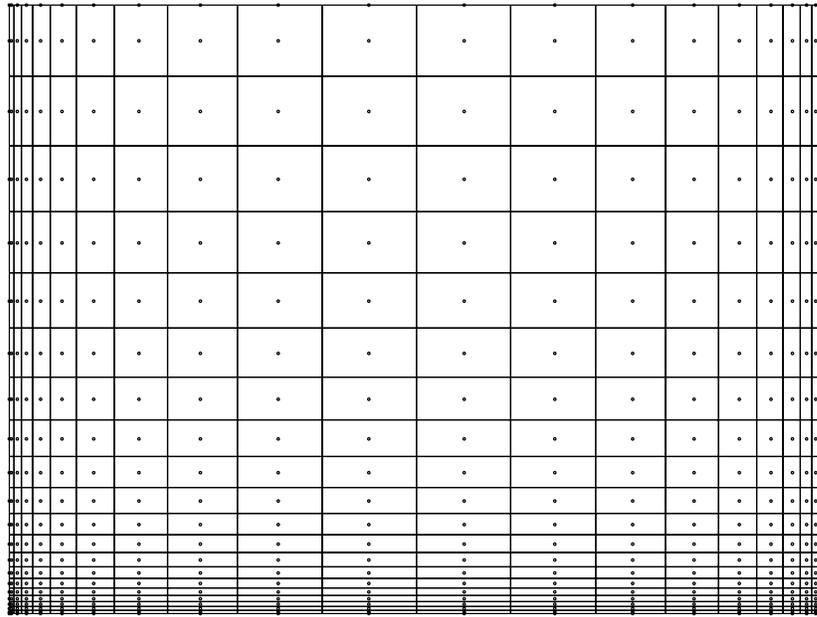


Fig. H.3 Physical domain of an example rectilinear grid with logarithmic spacing.

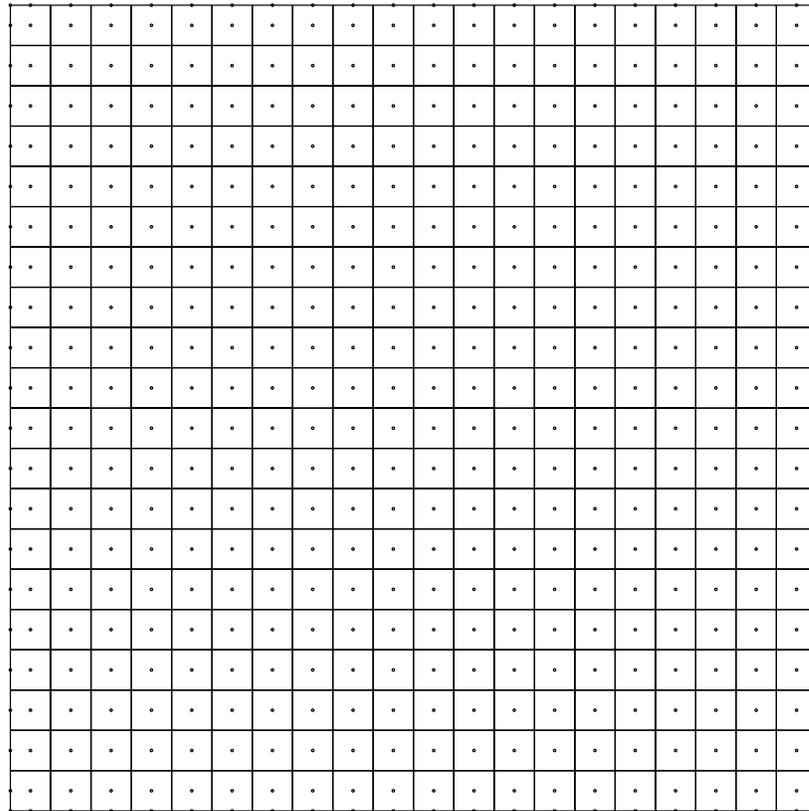


Fig. H.4 Computational domain of an example rectilinear grid with logarithmic spacing.

II. Discretization Schemes

A. Differencing

The Taylor series expansion for a function, $\phi(y)$ about a point y for the value at $y + \Delta y$ can be written

$$\phi(y + \Delta y) = \phi(y) + \left(\frac{\partial \phi}{\partial y}\right)_y \Delta y + \left(\frac{\partial^2 \phi}{\partial y^2}\right)_y \frac{\Delta y^2}{2} + \left(\frac{\partial^3 \phi}{\partial y^3}\right)_y \frac{\Delta y^3}{6} + O(\Delta y^4) + \dots \quad (\text{H.1})$$

This equation can be written in several discrete forms in order to yield useful approximations for values and derivatives near a point. Writing an equation for the Taylor series expansion at points a and b about point j gives

$$\phi_a = \phi(y_j + \Delta y_a) = \phi_j + \left(\frac{\partial \phi}{\partial y}\right)_j \Delta y_a + \left(\frac{\partial^2 \phi}{\partial y^2}\right)_j \frac{\Delta y_a^2}{2} + \left(\frac{\partial^3 \phi}{\partial y^3}\right)_j \frac{\Delta y_a^3}{6} + O(\Delta y_a^4) + \dots \quad (\text{H.2})$$

$$\phi_b = \phi(y_j + \Delta y_b) = \phi_j + \left(\frac{\partial \phi}{\partial y}\right)_j \Delta y_b + \left(\frac{\partial^2 \phi}{\partial y^2}\right)_j \frac{\Delta y_b^2}{2} + \left(\frac{\partial^3 \phi}{\partial y^3}\right)_j \frac{\Delta y_b^3}{6} + O(\Delta y_b^4) + \dots \quad (\text{H.3})$$

Subtracting the product of Eq. (H.3) and Δy_a from the product of Eq. (H.2) and Δy_b gives

$$\phi_a \Delta y_b - \phi_b \Delta y_a = \phi_j (\Delta y_b - \Delta y_a) + \left(\frac{\partial^2 \phi}{\partial y^2}\right)_j \frac{(\Delta y_b \Delta y_a^2 - \Delta y_a \Delta y_b^2)}{2} + O(\Delta y^4) + \dots \quad (\text{H.4})$$

Solving for ϕ_j gives

$$\phi_j = \frac{\phi_a \Delta y_b - \phi_b \Delta y_a}{(\Delta y_b - \Delta y_a)} - \left(\frac{\partial^2 \phi}{\partial y^2}\right)_j \frac{(\Delta y_b \Delta y_a^2 - \Delta y_a \Delta y_b^2)}{2(\Delta y_b - \Delta y_a)} + \dots \quad (\text{H.5})$$

Because the second term on the right-hand side of Eq. (H.5) is of order Δy^2 , dropping it yields the **second-order approximation for the value at a point**.

$$\phi_j = \frac{\phi_a \Delta y_b - \phi_b \Delta y_a}{(\Delta y_b - \Delta y_a)} \quad (\text{H.6})$$

This relationship is called linear interpolation if the point j lies between points a and b . Otherwise it is called linear extrapolation.

Subtracting the product of Eq. (H.3) and Δy_a^2 from the product of Eq. (H.2) and Δy_b^2 gives

$$\phi_a \Delta y_b^2 - \phi_b \Delta y_a^2 = \phi_j (\Delta y_b^2 - \Delta y_a^2) + \left(\frac{\partial \phi}{\partial y} \right)_j (\Delta y_b^2 \Delta y_a - \Delta y_a^2 \Delta y_b) + \left(\frac{\partial^3 \phi}{\partial y^3} \right)_j \frac{\Delta y_b^2 \Delta y_a^3 - \Delta y_a^2 \Delta y_b^3}{6} + \dots \quad (\text{H.7})$$

Solving for the first derivative at j gives

$$\left(\frac{\partial \phi}{\partial y} \right)_j = \frac{\phi_a \Delta y_b^2 - \phi_b \Delta y_a^2 - \phi_j (\Delta y_b^2 - \Delta y_a^2)}{(\Delta y_b^2 \Delta y_a - \Delta y_a^2 \Delta y_b)} - \left(\frac{\partial^3 \phi}{\partial y^3} \right)_j \frac{\Delta y_b^2 \Delta y_a^3 - \Delta y_a^2 \Delta y_b^3}{6(\Delta y_b^2 \Delta y_a - \Delta y_a^2 \Delta y_b)} + O(\Delta y^3) + \dots \quad (\text{H.8})$$

Because the second term on the right-hand side of Eq. (H.8) is of order Δy^2 , dropping it yields the **second-order approximation for the first derivative at a point**

$$\left(\frac{\partial \phi}{\partial y} \right)_j = \frac{\phi_a \Delta y_b^2 - \phi_b \Delta y_a^2 - \phi_j (\Delta y_b^2 - \Delta y_a^2)}{(\Delta y_b^2 \Delta y_a - \Delta y_a^2 \Delta y_b)} \quad (\text{H.9})$$

In order to develop a second-order approximation for the second derivative at point j , an additional equation must be added to Eqs. (H.2) and (H.3) for the Taylor series expansion at point c

$$\phi_c = \phi(y_j + \Delta y_c) = \phi_j + \left(\frac{\partial \phi}{\partial y} \right)_j \Delta y_c + \left(\frac{\partial^2 \phi}{\partial y^2} \right)_j \frac{\Delta y_c^2}{2} + \left(\frac{\partial^3 \phi}{\partial y^3} \right)_j \frac{\Delta y_c^3}{6} + O(\Delta y_c^4) + \dots \quad (\text{H.10})$$

After a considerable amount of algebra, Eqs. (H.2) (H.3) and (H.10) can be used to give the following **second-order approximation for the second derivative at a point**

$$\begin{aligned} \left(\frac{\partial^2 \phi}{\partial y^2} \right)_j &= \frac{2(\Delta y_a + \Delta y_b + \Delta y_c)}{\Delta y_a \Delta y_b \Delta y_c} \phi_j + \frac{2(\Delta y_b + \Delta y_c)}{\Delta y_a (\Delta y_a - \Delta y_b)(\Delta y_c - \Delta y_a)} \phi_a \\ &+ \frac{2(\Delta y_c + \Delta y_a)}{\Delta y_b (\Delta y_b - \Delta y_c)(\Delta y_a - \Delta y_b)} \phi_b + \frac{2(\Delta y_a + \Delta y_b)}{\Delta y_c (\Delta y_b - \Delta y_c)(\Delta y_c - \Delta y_a)} \phi_c + O(\Delta y^2) + \dots \end{aligned} \quad (\text{H.11})$$

The applications of the differencing approximations presented above will be discussed in detail in the following section for the case of one-dimensional approximations. The two-dimensional approximations are straight-forward and are included in the following section with less detail.

1. Cell Centers

In the notation implemented here, discrete values are used to represent an internal point, P , and its north (N) and south (S) neighboring points on a uniform grid with spacing equal to $\Delta \eta$. Using this notation, the equations from the previous section can be used with the uniform grid definitions $\Delta y_a = -\Delta \eta$,

$\Delta y_b = \Delta \eta$, $\phi_j = \phi_P$, $\phi_a = \phi_S$, and $\phi_b = \phi_N$. Using these definitions in Eqs. (H.6), (H.9), and (H.11) gives the second-order central difference approximations at an internal node P

$$\phi_P = \frac{\phi_N + \phi_S}{2} \quad (\text{H.12})$$

$$\left(\frac{\partial \phi}{\partial \eta} \right)_P = \frac{\phi_N - \phi_S}{2\Delta \eta} \quad (\text{H.13})$$

$$\left(\frac{\partial^2 \phi}{\partial \eta^2} \right)_P = \frac{\phi_N - 2\phi_P + \phi_S}{\Delta \eta^2} \quad (\text{H.14})$$

For a cell adjacent to a boundary, the uniform grid definitions cannot be applied. Instead we define the points such that ϕ_0 is the value at the wall, ϕ_1 is the value at the center of the first cell, ϕ_2 is the value at the center of the second cell, and ϕ_3 is the value at the center of the third cell. The near-wall definitions can be applied $\phi_j = \phi_1$, $\phi_a = \phi_0$, $\phi_b = \phi_2$, and $\phi_c = \phi_3$ with grid spacing $\Delta y_a = -\Delta \eta/2$, $\Delta y_b = \Delta \eta$, and $\Delta y_c = 2\Delta \eta$. Using these near-wall definitions in Eqs. (H.6), (H.9), and (H.11) gives the second-order forward difference approximations for a cell adjacent to a boundary

$$\phi_1 = \frac{2\phi_0 + \phi_2}{3} \quad (\text{H.15})$$

$$\left(\frac{\partial \phi}{\partial \eta} \right)_1 = \frac{-4\phi_0 + 3\phi_1 + \phi_2}{3\Delta \eta} \quad (\text{H.16})$$

$$\left(\frac{\partial^2 \phi}{\partial \eta^2} \right)_1 = \frac{48\phi_0 - 75\phi_1 + 30\phi_2 - 3\phi_3}{15\Delta \eta^2} \quad (\text{H.17})$$

2. Cell Faces

Consider an internal face, n , which is the north face of cell P and the south face of cell N . With uniform grid spacing equal to $\Delta \eta$, point n is the midpoint of nodes N and P . Using this notation, the equations from the previous section can be used with the uniform grid definitions $\Delta y_a = -\Delta \eta/2$, $\Delta y_b = \Delta \eta/2$, $\phi_j = \phi_n$, $\phi_a = \phi_P$, and $\phi_b = \phi_N$. Using these definitions in Eqs. (H.6), (H.9), and (H.11) gives the second-order central difference approximations for an internal face

$$\phi_n = \frac{\phi_N + \phi_P}{2} \quad (\text{H.18})$$

$$\left(\frac{\partial \phi}{\partial \eta} \right)_n = \frac{\phi_N - \phi_P}{\Delta \eta} \quad (\text{H.19})$$

$$\left(\frac{\partial^2 \phi}{\partial \eta^2} \right)_n = \frac{4\phi_N - 8\phi_n + 4\phi_P}{\Delta \eta^2} \quad (\text{H.20})$$

Boundary faces must be treated differently. Consider a cell adjacent to a boundary. Here we define the points such that ϕ_0 is the value at the wall, ϕ_1 is the value at the center of the first cell, ϕ_2 is the value at the center of the second cell, and ϕ_3 is the value at the center of the third cell. The near-wall definitions can be applied $\phi_j = \phi_0$, $\phi_a = \phi_1$, $\phi_b = \phi_2$, and $\phi_c = \phi_3$ with grid spacing $\Delta y_a = \Delta \eta/2$, $\Delta y_b = 3\Delta \eta/2$, and $\Delta y_c = 5\Delta \eta/2$. Using these near-wall definitions in Eqs. (H.6), (H.9), and (H.11) gives the second-order forward difference approximations for a boundary face where the transport property, ϕ , is known at cell centers

$$\phi_0 = \frac{3\phi_1 - \phi_2}{2} \quad (\text{H.21})$$

$$\left(\frac{\partial \phi}{\partial \eta} \right)_0 = \frac{-8\phi_0 + 9\phi_1 - \phi_2}{3\Delta \eta} \quad (\text{H.22})$$

$$\left(\frac{\partial^2 \phi}{\partial \eta^2} \right)_0 = \frac{72\phi_0 - 120\phi_1 + 60\phi_2 - 12\phi_3}{15\Delta \eta^2} \quad (\text{H.23})$$

In the case where ϕ is known at the cell faces on a uniform grid and not at the cell centers (as may be the case with estimating the Jacobian), the near-wall definitions can be applied $\phi_j = \phi_0$, $\phi_a = \phi_1$, $\phi_b = \phi_2$, and $\phi_c = \phi_3$ with grid spacing $\Delta y_a = \Delta \eta$, $\Delta y_b = 2\Delta \eta$, and $\Delta y_c = 3\Delta \eta$. Using these near-wall definitions in Eqs. (H.6), (H.9), and (H.11) gives the second-order forward difference approximations for a boundary face where the transport property, ϕ , is known at cell faces

$$\phi_0 = 2\phi_1 - \phi_2 \quad (\text{H.24})$$

$$\left(\frac{\partial \phi}{\partial \eta} \right)_0 = \frac{-3\phi_0 + 4\phi_1 - \phi_2}{2\Delta \eta} \quad (\text{H.25})$$

$$\left(\frac{\partial^2 \phi}{\partial \eta^2}\right)_0 = \frac{2\phi_0 - 5\phi_1 + 4\phi_2 - \phi_3}{\Delta \eta^2} \quad (\text{H.26})$$

B. Integration

The integral of a Taylor series expansion for a function, $\phi(y)$ about a point j from the point y_a to y_b can be written

$$\int_{y_a}^{y_b} \phi(y) dy = \int_{y_j + \Delta y_a}^{y_j + \Delta y_b} \left[\phi_j + \left(\frac{\partial \phi}{\partial y}\right)_j (y - y_j) + \left(\frac{\partial^2 \phi}{\partial y^2}\right)_j \frac{(y - y_j)^2}{2} + \left(\frac{\partial^3 \phi}{\partial y^3}\right)_j \frac{(y - y_j)^3}{6} + \dots \right] dy \quad (\text{H.27})$$

Integrating Eq. (H.12) gives

$$\begin{aligned} \int_{y_a}^{y_b} \phi(y) dy &= \phi_j (\Delta y_b - \Delta y_a) + \left(\frac{\partial \phi}{\partial y}\right)_j \frac{(\Delta y_b^2 - \Delta y_a^2)}{2} \\ &+ \left(\frac{\partial^2 \phi}{\partial y^2}\right)_j \frac{(\Delta y_b^3 - \Delta y_a^3)}{6} + \left(\frac{\partial^3 \phi}{\partial y^3}\right)_j \frac{(\Delta y_b^4 - \Delta y_a^4)}{24} + O(\Delta y^5) + \dots \end{aligned} \quad (\text{H.28})$$

Global error for integration is one order less than local truncation error. Therefore, for uniform grid spacing, only the first term on the right-hand side of the Eq. (H.28) must be retained for second-order accuracy. For a cell of size $\Delta \eta$ with the value specified at the midpoint, P , the integral over the cell volume can be evaluated by applying the definitions $\phi_j = \phi_P$, $\Delta y_b = \Delta \eta/2$, and $\Delta y_a = -\Delta \eta/2$. Using these definitions in Eq. (H.28) gives the second-order midpoint rule for an integral over a cell

$$\int_{-\Delta \eta/2}^{\Delta \eta/2} \phi(\eta) d\eta = \phi_P \Delta \eta \quad (\text{H.29})$$

C. Approximations in Two Dimensions

1. Cell Centers

Applying Eq. (H.13) at a node, P , that is not adjacent to a boundary gives

$$\begin{aligned} \left(\frac{\partial \phi}{\partial \xi}\right)_P &= \frac{\phi_E - \phi_W}{2\Delta \xi} \\ \left(\frac{\partial \phi}{\partial \eta}\right)_P &= \frac{\phi_N - \phi_S}{2\Delta \eta} \end{aligned} \quad (\text{H.30})$$

For a cell adjacent to the north boundary, applying Eq. (H.16) gives

$$\left(\frac{\partial\phi}{\partial\eta}\right)_P = \frac{4\phi_N - 3\phi_P - \phi_S}{3\Delta\eta} \quad (\text{H.31})$$

For a cell adjacent to the south boundary, applying Eq. (H.16) gives

$$\left(\frac{\partial\phi}{\partial\eta}\right)_P = \frac{-4\phi_S + 3\phi_P + \phi_N}{3\Delta\eta} \quad (\text{H.32})$$

For a cell adjacent to the east boundary, applying Eq. (H.16) gives

$$\left(\frac{\partial\phi}{\partial\xi}\right)_P = \frac{4\phi_E - 3\phi_P - \phi_W}{3\Delta\xi} \quad (\text{H.33})$$

For a cell adjacent to the west boundary, applying Eq. (H.16) gives

$$\left(\frac{\partial\phi}{\partial\xi}\right)_P = \frac{-4\phi_W + 3\phi_P + \phi_E}{3\Delta\xi} \quad (\text{H.34})$$

2. Cell Faces

Applying Eq. (H.18) at a node, P , that is not adjacent to a boundary gives

$$\begin{aligned} \phi_n &= \frac{\phi_N + \phi_P}{2} \\ \phi_s &= \frac{\phi_P + \phi_S}{2} \\ \phi_e &= \frac{\phi_E + \phi_P}{2} \\ \phi_w &= \frac{\phi_P + \phi_W}{2} \end{aligned} \quad (\text{H.35})$$

For a cell adjacent to the north boundary, applying Eq. (H.21) gives

$$\phi_n = \frac{3\phi_P - \phi_S}{2} \quad (\text{H.36})$$

For a cell adjacent to the south boundary, applying Eq. (H.21) gives

$$\phi_s = \frac{3\phi_P - \phi_N}{2} \quad (\text{H.37})$$

For a cell adjacent to the east boundary, applying Eq. (H.21) gives

$$\phi_e = \frac{3\phi_P - \phi_W}{2} \quad (\text{H.38})$$

For a cell adjacent to the west boundary, applying Eq. (H.21) gives

$$\phi_w = \frac{3\phi_P - \phi_E}{2} \quad (\text{H.39})$$

Applying Eq. (H.19) at a node, P , that is not adjacent to a boundary gives

$$\begin{aligned} \left(\frac{\partial \phi}{\partial \eta} \right)_n &= \frac{\phi_N - \phi_P}{\Delta \eta} \\ \left(\frac{\partial \phi}{\partial \eta} \right)_s &= \frac{\phi_P - \phi_S}{\Delta \eta} \\ \left(\frac{\partial \phi}{\partial \xi} \right)_e &= \frac{\phi_E - \phi_P}{\Delta \xi} \\ \left(\frac{\partial \phi}{\partial \xi} \right)_w &= \frac{\phi_P - \phi_W}{\Delta \xi} \end{aligned} \quad (\text{H.40})$$

For a cell adjacent to the north boundary, applying Eq. (H.22) gives

$$\left(\frac{\partial \phi}{\partial \eta} \right)_n = \frac{8\phi_N - 9\phi_P + \phi_S}{3\Delta \eta} \quad (\text{H.41})$$

For a cell adjacent to the south boundary, applying Eq. (H.22) gives

$$\left(\frac{\partial \phi}{\partial \eta} \right)_s = \frac{-8\phi_S + 9\phi_P - \phi_N}{3\Delta \eta} \quad (\text{H.42})$$

For a cell adjacent to the east boundary, applying Eq. (H.22) gives

$$\left(\frac{\partial \phi}{\partial \xi} \right)_e = \frac{8\phi_E - 9\phi_P + \phi_W}{3\Delta \xi} \quad (\text{H.43})$$

For a cell adjacent to the west boundary, applying Eq. (H.22) gives

$$\left(\frac{\partial \phi}{\partial \xi} \right)_w = \frac{-8\phi_W + 9\phi_P - \phi_E}{3\Delta \xi} \quad (\text{H.44})$$

Values at cell corners are simply the average of all neighboring cells. Therefore, cross-derivative terms are calculated for cell faces not on or adjacent to a boundary by applying Eq. (H.13)

$$\begin{aligned}
\left. \frac{\partial \phi}{\partial \xi} \right|_n &= \frac{\phi_{NE} - \phi_{NW} + \phi_E - \phi_W}{4\Delta\xi} \\
\left. \frac{\partial \phi}{\partial \xi} \right|_s &= \frac{\phi_{SE} - \phi_{SW} + \phi_E - \phi_W}{4\Delta\xi} \\
\left. \frac{\partial \phi}{\partial \eta} \right|_e &= \frac{\phi_{NE} - \phi_{SE} + \phi_N - \phi_S}{4\Delta\eta} \\
\left. \frac{\partial \phi}{\partial \eta} \right|_w &= \frac{\phi_{NW} - \phi_{SW} + \phi_N - \phi_S}{4\Delta\eta}
\end{aligned} \tag{H.45}$$

For a cell faces on or adjacent to the north boundary, Eqs. (H.13) and (H.16) give

$$\begin{aligned}
\left. \frac{\partial \phi}{\partial \xi} \right|_n &= \frac{\phi_{NE} - \phi_{NW}}{2\Delta\xi} \\
\left. \frac{\partial \phi}{\partial \eta} \right|_e &= \frac{4(\phi_N + \phi_{NE}) - 3(\phi_P + \phi_E) - (\phi_S + \phi_{SE})}{6\Delta\eta} \\
\left. \frac{\partial \phi}{\partial \eta} \right|_w &= \frac{4(\phi_N + \phi_{NW}) - 3(\phi_P + \phi_W) - (\phi_S + \phi_{SW})}{6\Delta\eta}
\end{aligned} \tag{H.46}$$

For a cell faces on or adjacent to the south boundary, Eqs. (H.13) and (H.16) give

$$\begin{aligned}
\left. \frac{\partial \phi}{\partial \xi} \right|_s &= \frac{\phi_{SE} - \phi_{SW}}{2\Delta\xi} \\
\left. \frac{\partial \phi}{\partial \eta} \right|_e &= \frac{-4(\phi_S + \phi_{SE}) + 3(\phi_P + \phi_E) + (\phi_N + \phi_{NE})}{6\Delta\eta} \\
\left. \frac{\partial \phi}{\partial \eta} \right|_w &= \frac{-4(\phi_S + \phi_{SW}) + 3(\phi_P + \phi_W) + (\phi_N + \phi_{NW})}{6\Delta\eta}
\end{aligned} \tag{H.47}$$

For a cell faces on or adjacent to the east boundary, Eqs. (H.13) and (H.16) give

$$\begin{aligned}
\left. \frac{\partial \phi}{\partial \eta} \right|_e &= \frac{\phi_{NE} - \phi_{SE}}{2\Delta\eta} \\
\left. \frac{\partial \phi}{\partial \xi} \right|_n &= \frac{4(\phi_E + \phi_{NE}) - 3(\phi_P + \phi_N) - (\phi_W + \phi_{NW})}{6\Delta\xi} \\
\left. \frac{\partial \phi}{\partial \xi} \right|_s &= \frac{4(\phi_E + \phi_{SE}) - 3(\phi_P + \phi_S) - (\phi_W + \phi_{SW})}{6\Delta\xi}
\end{aligned} \tag{H.48}$$

For a cell faces on or adjacent to the west boundary, Eqs. (H.13) and (H.16) give

$$\begin{aligned}
\left. \frac{\partial \phi}{\partial \eta} \right|_w &= \frac{\phi_{NW} - \phi_{SW}}{2\Delta\eta} \\
\left. \frac{\partial \phi}{\partial \xi} \right|_n &= \frac{-4(\phi_W + \phi_{NW}) + 3(\phi_P + \phi_N) + (\phi_E + \phi_{NE})}{6\Delta\eta} \\
\left. \frac{\partial \phi}{\partial \xi} \right|_s &= \frac{-4(\phi_W + \phi_{SW}) + 3(\phi_P + \phi_S) + (\phi_E + \phi_{SE})}{6\Delta\eta}
\end{aligned} \tag{H.49}$$

Corners of the grid require yet another treatment. At the north-east corner of the grid, Eq. (H.16) gives

$$\begin{aligned}
\left. \frac{\partial \phi}{\partial \eta} \right|_e &= \frac{4\phi_{NE} - 3\phi_E - \phi_{SE}}{3\Delta\eta} \\
\left. \frac{\partial \phi}{\partial \xi} \right|_n &= \frac{4\phi_{NE} - 3\phi_N - \phi_{NW}}{3\Delta\eta}
\end{aligned} \tag{H.50}$$

At the north-west corner of the grid, Eq. (H.16) gives

$$\begin{aligned}
\left. \frac{\partial \phi}{\partial \eta} \right|_w &= \frac{4\phi_{NW} - 3\phi_W - \phi_{SW}}{3\Delta\eta} \\
\left. \frac{\partial \phi}{\partial \xi} \right|_n &= \frac{-4\phi_{NW} + 3\phi_N + \phi_{NE}}{3\Delta\eta}
\end{aligned} \tag{H.51}$$

At the south-east corner of the grid, Eq. (H.16) gives

$$\begin{aligned}
\left. \frac{\partial \phi}{\partial \eta} \right|_e &= \frac{-4\phi_{SE} + 3\phi_E + \phi_{NE}}{3\Delta\eta} \\
\left. \frac{\partial \phi}{\partial \xi} \right|_s &= \frac{4\phi_{SE} - 3\phi_S - \phi_{SW}}{3\Delta\eta}
\end{aligned} \tag{H.52}$$

At the south-west corner of the grid, Eq. (H.16) gives

$$\begin{aligned}
\left. \frac{\partial \phi}{\partial \eta} \right|_w &= \frac{-4\phi_{SW} + 3\phi_W + \phi_{NW}}{3\Delta\eta} \\
\left. \frac{\partial \phi}{\partial \xi} \right|_s &= \frac{-4\phi_{SW} + 3\phi_S + \phi_{SE}}{3\Delta\eta}
\end{aligned} \tag{H.53}$$

3. Integration

In two dimensions Eq. (H.29) becomes

$$\int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \phi(\xi, \eta) d\xi d\eta = \phi_P \Delta\xi \Delta\eta \tag{H.54}$$

APPENDIX I

THE FINITE-VOLUME METHOD IN CARTESIAN COORDINATES

I. General Scalar Transport

The transport equation for any scalar, ϕ , can be written in curvilinear coordinates as

$$\frac{\partial(\bar{V}_\xi\phi)}{\partial\xi} + \frac{\partial(\bar{V}_\eta\phi)}{\partial\eta} = y_{,\eta}^2 \frac{\partial}{\partial\xi} \left(\Gamma J \frac{\partial\phi}{\partial\xi} \right) + x_{,\xi}^2 \frac{\partial}{\partial\eta} \left(\Gamma J \frac{\partial\phi}{\partial\eta} \right) + \frac{S}{J} \quad (1.1)$$

The finite volume method is used in this work to solve the governing equations. This method is applied by integrating about a rectangular control volume in the ξ and η directions

$$\int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left\{ \frac{\partial(\bar{V}_\xi\phi)}{\partial\xi} + \frac{\partial(\bar{V}_\eta\phi)}{\partial\eta} = y_{,\eta}^2 \frac{\partial}{\partial\xi} \left(\Gamma J \frac{\partial\phi}{\partial\xi} \right) + x_{,\xi}^2 \frac{\partial}{\partial\eta} \left(\Gamma J \frac{\partial\phi}{\partial\eta} \right) + \frac{S}{J} \right\} d\xi d\eta \quad (1.2)$$

This gives

$$(\bar{V}_\xi\phi)|_w^e \Delta\eta + (\bar{V}_\eta\phi)|_s^n \Delta\xi = y_{,\eta}^2 \left(\Gamma J \frac{\partial\phi}{\partial\xi} \right)|_w^e \Delta\eta + x_{,\xi}^2 \left(\Gamma J \frac{\partial\phi}{\partial\eta} \right)|_s^n \Delta\xi + \hat{S} \quad (1.3)$$

where

$$\hat{S} \equiv \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left(\frac{S}{J} \right) d\xi d\eta \quad (1.4)$$

The convection terms can be rewritten by defining the average volume flux terms on each face of the control volume

$$\begin{aligned} \dot{m}_e &\equiv \bar{V}_{\xi_e} \Delta\eta \\ \dot{m}_w &\equiv \bar{V}_{\xi_w} \Delta\eta \\ \dot{m}_n &\equiv \bar{V}_{\eta_n} \Delta\xi \\ \dot{m}_s &\equiv \bar{V}_{\eta_s} \Delta\xi \end{aligned} \quad (1.5)$$

Using these definitions for the volume flux and applying the second-order central differencing scheme to the diffusion terms gives **the discretized form of the governing equation for any cell, P , not adjacent to a boundary**

$$\begin{aligned}
\dot{m}_e\phi_e - \dot{m}_w\phi_w + \dot{m}_n\phi_n - \dot{m}_s\phi_s = & (y_{,\eta}^2)_P \left[\Gamma_e J_e \frac{\phi_E - \phi_P}{\Delta\xi} - \Gamma_w J_w \frac{\phi_P - \phi_W}{\Delta\xi} \right] \Delta\eta \\
& + (x_{,\xi}^2)_P \left[\Gamma_n J_n \frac{\phi_N - \phi_P}{\Delta\eta} - \Gamma_s J_s \frac{\phi_P - \phi_S}{\Delta\eta} \right] \Delta\xi + \hat{S}
\end{aligned} \tag{I.6}$$

The convection terms are calculated using deferred correction.

$$\dot{m}_e\phi_e - \dot{m}_w\phi_w + \dot{m}_n\phi_n - \dot{m}_s\phi_s = F_L + \beta(F_H^o - F_L^o) \tag{I.7}$$

The subscripts H and L stand for higher- and lower-order approximations and β is the blending factor. Several higher- and lower-order approximations could be used in this formulation. In this work, first-order upwinding has been implemented for the lower-order approximation, and second-order upwinding has been implemented for the higher-order approximation. The first-order upwinding approximation is calculated by retaining only the first term in the Taylor Series expansion about the upwind cell center

$$\begin{aligned}
F_L \equiv & \max[\dot{m}_e, 0]\phi_P - \max[-\dot{m}_e, 0]\phi_E \\
& + \max[-\dot{m}_w, 0]\phi_P - \max[\dot{m}_w, 0]\phi_W \\
& + \max[\dot{m}_n, 0]\phi_P - \max[-\dot{m}_n, 0]\phi_N \\
& + \max[-\dot{m}_s, 0]\phi_P - \max[\dot{m}_s, 0]\phi_S
\end{aligned} \tag{I.8}$$

The second-order upwinding approximation is calculated by retaining the first two terms in a Taylor Series expansion about the upwind cell center to approximate the value at the face

$$\begin{aligned}
F_H \equiv & \max[\dot{m}_e, 0] \left(\phi_P + \frac{\partial\phi}{\partial\xi} \Big|_P \frac{\Delta\xi}{2} \right) - \max[-\dot{m}_e, 0] \left(\phi_E - \frac{\partial\phi}{\partial\xi} \Big|_E \frac{\Delta\xi}{2} \right) \\
& + \max[-\dot{m}_w, 0] \left(\phi_P - \frac{\partial\phi}{\partial\xi} \Big|_P \frac{\Delta\xi}{2} \right) - \max[\dot{m}_w, 0] \left(\phi_W + \frac{\partial\phi}{\partial\xi} \Big|_W \frac{\Delta\xi}{2} \right) \\
& + \max[\dot{m}_n, 0] \left(\phi_P + \frac{\partial\phi}{\partial\eta} \Big|_P \frac{\Delta\eta}{2} \right) - \max[-\dot{m}_n, 0] \left(\phi_N - \frac{\partial\phi}{\partial\eta} \Big|_N \frac{\Delta\eta}{2} \right) \\
& + \max[-\dot{m}_s, 0] \left(\phi_P - \frac{\partial\phi}{\partial\eta} \Big|_P \frac{\Delta\eta}{2} \right) - \max[\dot{m}_s, 0] \left(\phi_S + \frac{\partial\phi}{\partial\eta} \Big|_S \frac{\Delta\eta}{2} \right)
\end{aligned} \tag{I.9}$$

The values in Eq. (I.9) contain derivatives estimated at cell centers. These are calculated according to the methods presented in Appendix H. The terms in the parentheses of Eq. (I.7) are calculated from the previous iteration as denoted by the superscript o and can be included as source terms.

$$S_c \equiv \beta(F_H^o - F_L^o) \tag{I.10}$$

Using Eqs. (I.7) – (I.10) in Eq. (I.6) gives

$$A_P \phi_P = A_E \phi_E + A_W \phi_W + A_N \phi_N + A_S \phi_S - S_c + \hat{S} \quad (\text{I.11})$$

where

$$\begin{aligned} A_E &= \max[-\dot{m}_e, 0] + (y_{,\eta}^2)_P \frac{\Delta \eta}{\Delta \xi} \Gamma_e J_e \\ A_W &= \max[\dot{m}_w, 0] + (y_{,\eta}^2)_P \frac{\Delta \eta}{\Delta \xi} \Gamma_w J_w \\ A_N &= \max[-\dot{m}_n, 0] + (x_{,\xi}^2)_P \frac{\Delta \xi}{\Delta \eta} \Gamma_n J_n \\ A_S &= \max[\dot{m}_s, 0] + (x_{,\xi}^2)_P \frac{\Delta \xi}{\Delta \eta} \Gamma_s J_s \\ A_P &= A_E + A_W + A_N + A_S \end{aligned} \quad (\text{I.12})$$

Solving Eq. (I.11) for ϕ_P gives

$$\phi_P = \frac{1}{A_P} (A_E \phi_E + A_W \phi_W + A_N \phi_N + A_S \phi_S - S_c + \hat{S}) \quad (\text{I.13})$$

Note that in the convection terms, the ϕ_P coefficients have been modified according to the equality

$$\begin{aligned} &(\max[\dot{m}_e, 0] + \max[-\dot{m}_w, 0] + \max[\dot{m}_n, 0] + \max[-\dot{m}_s, 0]) \phi_P \\ &= (\max[-\dot{m}_e, 0] + \max[\dot{m}_w, 0] + \max[-\dot{m}_n, 0] + \max[\dot{m}_s, 0]) \phi_P \\ &+ (\dot{m}_e - \dot{m}_w + \dot{m}_n - \dot{m}_s) \phi_P \end{aligned} \quad (\text{I.14})$$

in order to facilitate the summation in Eq. (I.12). The second-half of the right-hand side of Eq. (I.14) contains the continuity equation and is equal to zero for incompressible flows. Applying an under-relaxation variable, α , and the previous solution, ϕ_P^o , gives an under-relaxation form of Eq. (I.13),

$$\phi_P = (1 - \alpha) \phi_P^o + \frac{\alpha}{A_P} (A_E \phi_E + A_W \phi_W + A_N \phi_N + A_S \phi_S - S_c + \hat{S}) \quad (\text{I.15})$$

II. Boundary Condition Implementation

Two types of boundary conditions can be applied. A Dirichlet boundary condition exists when the value of the transport property is known at the boundary. A Neumann boundary condition exists when the gradient of the transport property is known at the boundary. The implementation of each of these conditions will be discussed here.

A. The Dirichlet Boundary Condition

For a cell adjacent to a boundary at which a Dirichlet boundary condition should be applied, the discretization method presented above is used, but the second-order differencing scheme for the boundary is employed. For example, using the approximation given in Eq. (H.42), the discretized form of Eq. (I.3) for a cell adjacent to the south boundary is

$$\begin{aligned} \dot{m}_e \phi_e - \dot{m}_w \phi_w + \dot{m}_n \phi_n - \dot{m}_s \phi_s = (y_{,\eta}^2)_P \left[\Gamma_e J_e \frac{\phi_E - \phi_P}{\Delta \xi} - \Gamma_w J_w \frac{\phi_P - \phi_W}{\Delta \xi} \right] \Delta \eta \\ + (x_{,\xi}^2)_P \left[\Gamma_n J_n \frac{\phi_N - \phi_P}{\Delta \eta} - \Gamma_s J_s \frac{-8\phi_S + 9\phi_P - \phi_N}{3\Delta \eta} \right] \Delta \xi + \hat{S} \end{aligned} \quad (\text{I.16})$$

where ϕ_S is the value of the transport property at the south wall. Using the same methods as presented above for the convection terms, Eq. (I.16) can be rewritten to match Eq. (I.11) where

$$\begin{aligned} A_E &= \max[-\dot{m}_e, 0] + (y_{,\eta}^2)_P \Gamma_e J_e \frac{\Delta \eta}{\Delta \xi} \\ A_W &= \max[\dot{m}_w, 0] + (y_{,\eta}^2)_P \Gamma_w J_w \frac{\Delta \eta}{\Delta \xi} \\ A_N &= \max[-\dot{m}_n, 0] + (x_{,\xi}^2)_P \frac{\Delta \xi}{\Delta \eta} \left(\Gamma_n J_n + \frac{1}{3} \Gamma_s J_s \right) \\ A_S &= \max[\dot{m}_s, 0] + (x_{,\xi}^2)_P \frac{\Delta \xi}{\Delta \eta} \frac{8}{3} \Gamma_s J_s \\ A_P &= A_E + A_W + A_N + A_S \end{aligned} \quad (\text{I.17})$$

Likewise, for a cell adjacent to the north boundary,

$$\begin{aligned} A_E &= \max[-\dot{m}_e, 0] + (y_{,\eta}^2)_P \Gamma_e J_e \frac{\Delta \eta}{\Delta \xi} \\ A_W &= \max[\dot{m}_w, 0] + (y_{,\eta}^2)_P \Gamma_w J_w \frac{\Delta \eta}{\Delta \xi} \\ A_N &= \max[-\dot{m}_n, 0] + (x_{,\xi}^2)_P \frac{\Delta \xi}{\Delta \eta} \frac{8}{3} \Gamma_n J_n \\ A_S &= \max[\dot{m}_s, 0] + (x_{,\xi}^2)_P \frac{\Delta \xi}{\Delta \eta} \left(\Gamma_s J_s + \frac{1}{3} \Gamma_n J_n \right) \\ A_P &= A_E + A_W + A_N + A_S \end{aligned} \quad (\text{I.18})$$

For a cell adjacent to the east boundary,

$$\begin{aligned}
A_E &= \max[-\dot{m}_e, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \frac{8}{3} \Gamma_e J_e \\
A_W &= \max[\dot{m}_w, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \left(\Gamma_w J_w + \frac{1}{3} \Gamma_e J_e \right) \\
A_N &= \max[-\dot{m}_n, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_n J_n \\
A_S &= \max[\dot{m}_s, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_s J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{I.19}$$

For a cell adjacent to the west boundary

$$\begin{aligned}
A_E &= \max[-\dot{m}_e, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \left(\Gamma_e J_e + \frac{1}{3} \Gamma_w J_w \right) \\
A_W &= \max[\dot{m}_w, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \frac{8}{3} \Gamma_w J_w \\
A_N &= \max[-\dot{m}_n, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_n J_n \\
A_S &= \max[\dot{m}_s, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_s J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{I.20}$$

B. The Neumann Boundary Condition

For a cell adjacent to a boundary at which a Neumann boundary condition should be applied, the discretization method presented above is used, but the gradient along the boundary is directly applied. For example, if a Neumann condition is known at the south boundary, the discretized form of Eq. (I.3) for a cell adjacent to the south boundary is

$$\begin{aligned}
\dot{m}_e \phi_e - \dot{m}_w \phi_w + \dot{m}_n \phi_n - \dot{m}_s \phi_s &= (y_{,\eta}^2)_P \left[\Gamma_e J_e \frac{\phi_E - \phi_P}{\Delta\xi} - \Gamma_w J_w \frac{\phi_P - \phi_W}{\Delta\xi} \right] \Delta\eta \\
&+ (x_{,\xi}^2)_P \left[\Gamma_n J_n \frac{\phi_N - \phi_P}{\Delta\eta} - \Gamma_s J_s \left(\frac{\partial \phi}{\partial \eta} \right)_s \right] \Delta\xi + \hat{S}
\end{aligned} \tag{I.21}$$

A common Neumann boundary condition is the zero gradient boundary condition, for example, at a symmetry boundary. For a zero gradient boundary condition on the south boundary,

$$\begin{aligned}
\dot{m}_e\phi_e - \dot{m}_w\phi_w + \dot{m}_n\phi_n - \dot{m}_s\phi_s = & (y_{,\eta}^2)_P \left[\Gamma_e J_e \frac{\phi_E - \phi_P}{\Delta\xi} - \Gamma_w J_w \frac{\phi_P - \phi_W}{\Delta\xi} \right] \Delta\eta \\
& + (x_{,\xi}^2)_P \left[\Gamma_n J_n \frac{\phi_N - \phi_P}{\Delta\eta} - 0 \right] \Delta\xi + \hat{S}
\end{aligned} \tag{I.22}$$

Using the same methods as presented above for the convection terms, Eq. (I.22) can be rewritten to match Eq. (I.11) where

$$\begin{aligned}
A_E &= \max[-\dot{m}_e, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_e J_e \\
A_W &= \max[\dot{m}_w, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_w J_w \\
A_N &= \max[-\dot{m}_n, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_n J_n \\
A_S &= \max[\dot{m}_s, 0] \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{I.23}$$

Likewise, for a cell adjacent to the north boundary,

$$\begin{aligned}
A_E &= \max[-\dot{m}_e, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_e J_e \\
A_W &= \max[\dot{m}_w, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_w J_w \\
A_N &= \max[-\dot{m}_n, 0] \\
A_S &= \max[\dot{m}_s, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_s J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{I.24}$$

For a cell adjacent to the east boundary,

$$\begin{aligned}
A_E &= \max[-\dot{m}_e, 0] \\
A_W &= \max[\dot{m}_w, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_w J_w \\
A_N &= \max[-\dot{m}_n, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_n J_n \\
A_S &= \max[\dot{m}_s, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_s J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{I.25}$$

For a cell adjacent to the west boundary,

$$\begin{aligned}
A_E &= \max[-\dot{m}_e, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_e J_e \\
A_W &= \max[\dot{m}_w, 0] \\
A_N &= \max[-\dot{m}_n, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_n J_n \\
A_S &= \max[\dot{m}_s, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_s J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{I.26}$$

Another common Neumann boundary condition is the constant gradient boundary condition where the gradient of the transport property normal to the boundary is assumed constant over the cell. This condition is sometimes used at a pressure boundary. For a constant gradient boundary condition on the north boundary of the domain, the gradient on the north face of each cell along the boundary is equal to the gradient on the south face of the cell.

$$\begin{aligned}
\left. \frac{\partial\phi}{\partial y} \right|_n &= \left. \frac{\partial\phi}{\partial y} \right|_s \\
\left(J_{x,\xi} \frac{\partial\phi}{\partial\eta} \right)_n &= \left(J_{x,\xi} \frac{\partial\phi}{\partial\eta} \right)_s
\end{aligned} \tag{I.27}$$

This can be applied to give the discretized form of Eq. (I.3) for a cell adjacent to a north boundary

$$\begin{aligned}
\dot{m}_e\phi_e - \dot{m}_w\phi_w + \dot{m}_n\phi_n - \dot{m}_s\phi_s &= (y_{,\eta}^2)_P \left[\Gamma_e J_e \frac{\phi_E - \phi_P}{\Delta\xi} - \Gamma_w J_w \frac{\phi_P - \phi_W}{\Delta\xi} \right] \Delta\eta \\
&+ (x_{,\xi}^2)_P \left[(\Gamma_n - \Gamma_s) J_s \frac{\phi_P - \phi_S}{\Delta\eta} \right] \Delta\xi + \hat{S}
\end{aligned} \tag{I.28}$$

Using the same methods as presented above for the convection terms, Eq. (I.28) can be rewritten to match Eq. (I.11) where

$$\begin{aligned}
A_E &= \max[-\dot{m}_e, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_e J_e \\
A_W &= \max[\dot{m}_w, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_w J_w \\
A_N &= \max[-\dot{m}_n, 0] \\
A_S &= \max[\dot{m}_s, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} (\Gamma_s - \Gamma_n) J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{I.29}$$

For a cell adjacent to a south boundary with a constant gradient boundary condition,

$$\begin{aligned}
A_E &= \max[-\dot{m}_e, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_e J_e \\
A_W &= \max[\dot{m}_w, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_w J_w \\
A_N &= \max[-\dot{m}_n, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} (\Gamma_n - \Gamma_s) J_n \\
A_S &= \max[\dot{m}_s, 0] \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{I.31}$$

For a cell adjacent to an east constant gradient boundary,

$$\begin{aligned}
A_E &= \max[-\dot{m}_e, 0] \\
A_W &= \max[\dot{m}_w, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} (\Gamma_w - \Gamma_e) J_w \\
A_N &= \max[-\dot{m}_n, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_n J_n \\
A_S &= \max[\dot{m}_s, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_s J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{I.32}$$

For a cell adjacent to a west constant gradient boundary,

$$\begin{aligned}
A_E &= \max[-\dot{m}_e, 0] + (y_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} (\Gamma_e - \Gamma_w) J_e \\
A_W &= \max[\dot{m}_w, 0] \\
A_N &= \max[-\dot{m}_n, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_n J_n \\
A_S &= \max[\dot{m}_s, 0] + (x_{,\xi}^2)_P \frac{\Delta\xi}{\Delta\eta} \Gamma_s J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{I.33}$$

C. Velocity Inlet

At a velocity inlet, the properties of the flow must be known. Therefore, a velocity inlet boundary condition is modeled using Dirichlet boundary conditions on all transport properties of the flow.

D. Outlet

At an outlet, the flow is assumed to be in a fully developed state. In other words, it is assumed that there are no gradients in the flow normal to the boundary at an outlet. Therefore, all flow properties are modeled using Neumann boundary conditions where the gradient normal to the boundary is equal to zero.

E. Smooth, No-Slip Wall

At a smooth wall, the velocity and turbulent velocity fluctuations must go to zero. Therefore, at a wall, the Dirichlet boundary conditions

$$\bar{V}_x = \bar{V}_y = 0 \quad (\text{I.34})$$

are applied for the x - and y -momentum transport equations.

Smooth-wall boundary conditions for the turbulent kinetic energy require that both k and its first derivative go to zero at the wall. Therefore, the value of k at a cell adjacent to a wall boundary is specified simply from the value of k at the second cell off of the wall. If k_0 is the value of k at the wall, k_1 is the value of k at the first cell off of the wall, and k_2 is the value of k at the second cell off of the wall, Eq. (H.22) can be applied to develop the following two relationships which must be enforced to correctly model the no-slip boundary condition

$$\begin{aligned} k_0 &= 0 \\ -8k_0 + 9k_1 - k_2 &= 0 \end{aligned} \quad (\text{I.35})$$

A Dirichlet boundary condition is often applied for the second transport equation at a wall. However, it is incorrect to refer to the value of the second transport property (ε , ω , ζ , $\tilde{\omega}$, etc.) at a wall as a boundary condition. Because of the order of the system of equations and because two boundary conditions are applied to the turbulent kinetic energy, k , it is mathematically incorrect to apply a boundary condition to the second transport property at a wall. However, examining the model equations often gives a near-wall approximation for the second transport property. This near-wall function is dependent on the model and is applied in the same way a Dirichlet boundary condition is applied. In this work the value of ε at a wall will be given the symbol $f_{\varepsilon 0}$ and the value of ω at a wall will be given the symbol $f_{\omega 0}$.

F. Symmetry

A symmetry boundary conditions assumes that gradients in the flow normal to the boundary are zero, and that no mass crosses the boundary. Because the x - and y -momentum equations combine to form a vector equation, the process for implementing a symmetry boundary condition includes both a Neumann

condition and a Dirichlet condition. For example, at a line of symmetry on the north or south boundary, the boundary conditions are

$$\begin{aligned} \text{Neumann } \frac{d\bar{V}_x}{d\eta} &= 0 \\ \text{Dirichlet } \bar{V}_y &= 0 \end{aligned} \tag{I.36}$$

At a line of symmetry on the east or west boundary, the boundary conditions are

$$\begin{aligned} \text{Neumann } \frac{d\bar{V}_y}{d\eta} &= 0 \\ \text{Dirichlet } \bar{V}_x &= 0 \end{aligned} \tag{I.37}$$

For any other transport property, a Neumann condition is applied with the gradient normal to the boundary equal to zero.

G. Pressure

Generally, a pressure boundary condition is used to represent a boundary across which the pressure is known, and across which the flow is allowed to enter or exit. In this work, it was helpful to specify two different types of pressure boundary conditions. A pressure inlet is a constant-pressure boundary across which flow is entering. A pressure outlet is a constant-pressure boundary across which flow is exiting. This differentiation between the two types of pressure boundary conditions helped to simplify the implementation of a pressure boundary condition.

At a pressure inlet, the properties of the flow must be known. In this work, it is assumed that at a pressure inlet, the flow enters the domain normal to the boundary. Applying mass conservation, this requires that the gradient of the normal velocity component must be zero at the boundary. For example, if the north side of a domain is specified as a pressure inlet boundary, flow must enter the domain without any x -velocity component. Continuity requires that

$$\frac{d\bar{V}_x}{dx} = -\frac{d\bar{V}_y}{dy} \tag{I.38}$$

Since it is assumed that the flow enters the domain in a uniform fashion along the boundary, this means that the left-hand side of Eq. (I.38) must be zero. Therefore, the y -velocity component must have a zero gradient

at the pressure boundary. This same procedure could be applied to any side of the flow domain to show that the velocity normal to the boundary must have a zero gradient normal to the boundary across the inlet pressure boundary if the flow enters normal to the boundary. Therefore, a pressure inlet boundary requires that a Dirichlet condition be applied to one of the flow velocity components, while a Neumann condition of zero gradient be applied to the other flow velocity component. All other transport properties must be specified at a pressure inlet boundary, and a Dirichlet condition is applied for these transport properties.

A pressure outlet boundary is specified across a side of the domain where the pressure is known and it is assumed that flow is exiting but the gradients of the flow properties are unknown. In this work, a Neumann condition is applied at a pressure outlet and the gradient is assumed constant across the cell adjacent to the boundary. Without knowing the exact solution to the flowfield, it is difficult to improve upon this type of implementation of a pressure outlet. Some codes assume a constant gradient of the properties, similar to how an outlet boundary condition is implemented here. However, such an implementation assumes that the flow has reached a fully developed state, which is not realistic for many outlet boundary scenarios.

III. Transport Equations

A. Momentum

The x -momentum equation is a special case of Eq. (I.1) where $\phi = \bar{V}_x$ and $\Gamma = \nu + \nu_t$. The y -momentum equation is a special case of Eq. (I.1) where $\phi = \bar{V}_y$ and $\Gamma = \nu + \nu_t$. The x - and y -momentum source terms include pressure source terms and viscous source terms and can be written in the transformed coordinate system as

$$\begin{aligned} S_x &= -J_{y,\eta} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \xi} + J_{y,\eta}^2 \frac{\partial}{\partial \xi} \left(\Gamma J \frac{\partial \bar{V}_x}{\partial \xi} \right) + J \frac{\partial}{\partial \eta} \left(\Gamma \frac{\partial \bar{V}_y}{\partial \xi} \right) \\ S_y &= -J_{x,\xi} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \eta} + J_{x,\xi}^2 \frac{\partial}{\partial \eta} \left(\Gamma J \frac{\partial \bar{V}_y}{\partial \eta} \right) + J \frac{\partial}{\partial \xi} \left(\Gamma \frac{\partial \bar{V}_x}{\partial \eta} \right) \end{aligned} \quad (I.39)$$

Integrating about a rectangular control volume in the ξ and η directions gives

$$\begin{aligned}
\hat{S}_x &= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left(\frac{S_x}{J} \right) d\xi d\eta \\
&= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left\{ -y_{,\eta} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \xi} + y_{,\eta}^2 \frac{\partial}{\partial \xi} \left(\Gamma J \frac{\partial \bar{V}_x}{\partial \xi} \right) + \frac{\partial}{\partial \eta} \left(\Gamma \frac{\partial \bar{V}_y}{\partial \xi} \right) \right\} d\xi d\eta \\
\hat{S}_y &= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left(\frac{S_y}{J} \right) d\xi d\eta \\
&= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left\{ -x_{,\xi} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \eta} + x_{,\xi}^2 \frac{\partial}{\partial \eta} \left(\Gamma J \frac{\partial \bar{V}_y}{\partial \eta} \right) + \frac{\partial}{\partial \xi} \left(\Gamma \frac{\partial \bar{V}_x}{\partial \eta} \right) \right\} d\xi d\eta
\end{aligned} \tag{I.40}$$

which can be approximated as

$$\begin{aligned}
\hat{S}_x &= -y_{,\eta} \frac{1}{\rho} (\hat{p})_w^e \Delta\eta + y_{,\eta}^2 \left(\Gamma J \frac{\partial \bar{V}_x}{\partial \xi} \right)_w^n \Delta\eta + \left(\Gamma \frac{\partial \bar{V}_y}{\partial \xi} \right)_s^n \Delta\xi \\
\hat{S}_y &= -x_{,\xi} \frac{1}{\rho} (\hat{p})_s^n \Delta\xi + x_{,\xi}^2 \left(\Gamma J \frac{\partial \bar{V}_y}{\partial \eta} \right)_s^n \Delta\xi + \left(\Gamma \frac{\partial \bar{V}_x}{\partial \eta} \right)_w^e \Delta\eta
\end{aligned} \tag{I.41}$$

and rearranged to give

$$\begin{aligned}
\hat{S}_x &= y_{,\eta} \frac{1}{\rho} (\hat{p}_w - \hat{p}_e) \Delta\eta + y_{,\eta}^2 \left[\Gamma_e J_e \left(\frac{\partial \bar{V}_x}{\partial \xi} \right)_e - \Gamma_w J_w \left(\frac{\partial \bar{V}_x}{\partial \xi} \right)_w \right] \Delta\eta \\
&\quad + \left[\Gamma_n \left(\frac{\partial \bar{V}_y}{\partial \xi} \right)_n - \Gamma_s \left(\frac{\partial \bar{V}_y}{\partial \xi} \right)_s \right] \Delta\xi \\
\hat{S}_y &= x_{,\xi} \frac{1}{\rho} (\hat{p}_s - \hat{p}_n) \Delta\xi + x_{,\xi}^2 \left[\Gamma_n J_n \left(\frac{\partial \bar{V}_y}{\partial \eta} \right)_n - \Gamma_s J_s \left(\frac{\partial \bar{V}_y}{\partial \eta} \right)_s \right] \Delta\xi \\
&\quad + \left[\Gamma_e \left(\frac{\partial \bar{V}_x}{\partial \eta} \right)_e - \Gamma_w \left(\frac{\partial \bar{V}_x}{\partial \eta} \right)_w \right] \Delta\eta
\end{aligned} \tag{I.42}$$

These derivatives are calculated according to the methods presented earlier in this appendix.

Once an estimate for the velocity field has been obtained, the shear stress along any grid boundary can be estimated. For example, the shear stress can be written in the computational domain along the south boundary as

$$\begin{aligned}
\tau_w &\equiv \left[\rho(v + v_t) \frac{\partial \bar{V}_x}{\partial y} \right]_0 \\
&= \left[\rho(v + v_t) \eta_{,y} \frac{\partial \bar{V}_x}{\partial \eta} \right]_0 \\
&= \left[\rho(v + v_t) J_{x,\xi} \frac{\partial \bar{V}_x}{\partial \eta} \right]_0
\end{aligned} \tag{I.43}$$

Applying Eq. (H.42) gives the discretized form

$$\tau_w = \rho(v + v_t)_s J_{s,x,\xi_s} \frac{-8\bar{V}_{x_s} + 9\bar{V}_{x_p} - \bar{V}_{x_N}}{3\Delta\eta} \tag{I.44}$$

Likewise, along a north boundary the shear stress can be calculated as

$$\tau_w = \rho(v + v_t)_n J_{n,x,\xi_n} \frac{8\bar{V}_{x_N} - 9\bar{V}_{x_p} + \bar{V}_{x_S}}{3\Delta\eta} \tag{I.45}$$

Along an east boundary the shear stress can be calculated as

$$\tau_w = \rho(v + v_t)_e J_{e,y,\eta_e} \frac{8\bar{V}_{y_E} - 9\bar{V}_{y_p} + \bar{V}_{y_W}}{3\Delta\xi} \tag{I.46}$$

Along a west boundary the shear stress can be calculated as

$$\tau_w = \rho(v + v_t)_w J_{w,y,\eta_w} \frac{-8\bar{V}_{y_W} + 9\bar{V}_{y_p} - \bar{V}_{y_E}}{3\Delta\xi} \tag{I.47}$$

B. Turbulent Kinetic Energy

The turbulent kinetic energy equation is a special case of Eq. (I.1) where $\phi = k$, $\Gamma = \nu + \nu_t/\sigma_k$, and the source term is

$$S_k = f_{k1} S_V^2 - f_{k2} \tag{I.48}$$

Integrating about a rectangular control volume in the ξ and η directions gives

$$\begin{aligned}
\hat{S}_k &= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left(\frac{S_k}{J} \right) d\xi d\eta \\
&= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left(\frac{f_{k1} S_V^2 - f_{k2}}{J} \right) d\xi d\eta
\end{aligned} \tag{I.49}$$

Applying Eq. (H.29) gives

$$\hat{S}_k = (f_{k1}S_V^2 - f_{k2}) \frac{\Delta\xi\Delta\eta}{J} \quad (\text{I.50})$$

C. Dissipation

The dissipation equation is a special case of Eq. (I.1) where $\phi = \varepsilon$, $\Gamma = \nu + \nu_t/\sigma_\varepsilon$, and the source term is

$$S_\varepsilon = f_{\varepsilon1}S_V^2 - f_{\varepsilon2} \quad (\text{I.51})$$

Integrating about a rectangular control volume in the ξ and η directions gives

$$\begin{aligned} \hat{S}_\varepsilon &= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left(\frac{S_\varepsilon}{J} \right) d\xi d\eta \\ &= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left(\frac{f_{\varepsilon1}S_V^2 - f_{\varepsilon2}}{J} \right) d\xi d\eta \end{aligned} \quad (\text{I.52})$$

Applying Eq. (H.29) gives

$$\hat{S}_\varepsilon = (f_{\varepsilon1}S_V^2 - f_{\varepsilon2}) \frac{\Delta\xi\Delta\eta}{J} \quad (\text{I.53})$$

D. Dissipation Frequency

The dissipation frequency equation is a special case of Eq. (I.1) where $\phi = \omega$, $\Gamma = \nu + \nu_t/\sigma_\omega$, and the source term is

$$S_\omega = f_{\omega1}S_V^2 - f_{\omega2} \quad (\text{I.54})$$

Integrating about a rectangular control volume in the ξ and η directions gives

$$\begin{aligned} \hat{S}_\omega &= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left(\frac{S_\omega}{J} \right) d\xi d\eta \\ &= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left(\frac{f_{\omega1}S_V^2 - f_{\omega2}}{J} \right) d\xi d\eta \end{aligned} \quad (\text{I.55})$$

Applying Eq. (H.29) gives

$$\hat{S}_\omega = (f_{\omega1}S_V^2 - f_{\omega2}) \frac{\Delta\xi\Delta\eta}{J} \quad (\text{I.56})$$

APPENDIX J

FINITE-VOLUME METHOD IN CYLINDRICAL COORDINATES

I. General Scalar Transport

The transport equation for any scalar, ϕ , can be written in curvilinear coordinates as

$$\bar{V}_\xi \frac{\partial \phi}{\partial \xi} + \bar{V}_\eta \frac{\partial \phi}{\partial \eta} = r_{,\eta}^2 \frac{\partial}{\partial \xi} \left(\Gamma J \frac{\partial \phi}{\partial \xi} \right) + z_{,\xi}^2 \frac{1}{r} \frac{\partial}{\partial \eta} \left(r \Gamma J \frac{\partial \phi}{\partial \eta} \right) + \frac{S}{J} \quad (\text{J.1})$$

The finite volume method is used in this work to solve the governing equations. This method is applied by integrating about a rectangular control volume in the ξ and η directions

$$\int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left\{ \bar{V}_\xi \frac{\partial \phi}{\partial \xi} + \bar{V}_\eta \frac{\partial \phi}{\partial \eta} = r_{,\eta}^2 \frac{\partial}{\partial \xi} \left(\Gamma J \frac{\partial \phi}{\partial \xi} \right) + z_{,\xi}^2 \frac{1}{r} \frac{\partial}{\partial \eta} \left(r \Gamma J \frac{\partial \phi}{\partial \eta} \right) + \frac{S}{J} \right\} d\xi d\eta \quad (\text{J.2})$$

This gives

$$\bar{V}_\xi(\phi)|_w^e \Delta\eta + \bar{V}_\eta(\phi)|_s^n \Delta\xi = r_{,\eta}^2 \left(\Gamma J \frac{\partial \phi}{\partial \xi} \right) \Big|_w^e \Delta\eta + z_{,\xi}^2 \frac{1}{r} \left(r \Gamma J \frac{\partial \phi}{\partial \eta} \right) \Big|_s^n \Delta\xi + \hat{S} \quad (\text{J.3})$$

where

$$\hat{S} \equiv \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left(\frac{S}{J} \right) d\xi d\eta \quad (\text{J.4})$$

The convection terms can be rewritten by defining the average volume flux terms on each face of the control volume

$$\begin{aligned} \dot{m}_\xi &\equiv \bar{V}_\xi \Delta\eta \\ \dot{m}_\eta &\equiv \bar{V}_\eta \Delta\xi \end{aligned} \quad (\text{J.5})$$

Using these definitions for the volume flux and applying the second-order central differencing scheme to the diffusion terms gives **the discretized form of the governing equation for any cell, P , not adjacent to a boundary**

$$\begin{aligned} \dot{m}_\xi(\phi_e - \phi_w) + \dot{m}_\eta(\phi_n - \phi_s) = (r_\eta^2)_P \left[\Gamma_e J_e \frac{\phi_E - \phi_P}{\Delta\xi} - \Gamma_w J_w \frac{\phi_P - \phi_W}{\Delta\xi} \right] \Delta\eta \\ + \left(\frac{z_\xi^2}{r} \right)_P \left[r_n \Gamma_n J_n \frac{\phi_N - \phi_P}{\Delta\eta} - r_s \Gamma_s J_s \frac{\phi_P - \phi_S}{\Delta\eta} \right] \Delta\xi + \hat{S} \end{aligned} \quad (\text{J.6})$$

The convection terms are calculated using deferred correction.

$$\dot{m}_\xi\phi_e - \dot{m}_\xi\phi_w + \dot{m}_\eta\phi_n - \dot{m}_\eta\phi_s = F_L + \beta(F_H^o - F_L^o) \quad (\text{J.7})$$

The subscripts H and L stand for higher- and lower-order approximations and β is the blending factor. Several higher- and lower-order approximations could be used in this formulation. In this work, first-order upwinding has been implemented for the lower-order approximation, and second-order upwinding has been implemented for the higher-order approximation. The first-order upwinding approximation is calculated by retaining only the first term in the Taylor Series expansion about the upwind cell center

$$\begin{aligned} F_L \equiv & \max[\dot{m}_\xi, 0]\phi_P - \max[-\dot{m}_\xi, 0]\phi_E \\ & + \max[-\dot{m}_\xi, 0]\phi_P - \max[\dot{m}_\xi, 0]\phi_W \\ & + \max[\dot{m}_\eta, 0]\phi_P - \max[-\dot{m}_\eta, 0]\phi_N \\ & + \max[-\dot{m}_\eta, 0]\phi_P - \max[\dot{m}_\eta, 0]\phi_S \end{aligned} \quad (\text{J.8})$$

The second-order upwinding approximation is calculated by retaining the first two terms in a Taylor Series expansion about the upwind cell center to approximate the value at the face

$$\begin{aligned} F_H \equiv & \max[\dot{m}_\xi, 0] \left(\phi_P + \frac{\partial\phi}{\partial\xi} \Big|_P \frac{\Delta\xi}{2} \right) - \max[-\dot{m}_\xi, 0] \left(\phi_E - \frac{\partial\phi}{\partial\xi} \Big|_E \frac{\Delta\xi}{2} \right) \\ & + \max[-\dot{m}_\xi, 0] \left(\phi_P - \frac{\partial\phi}{\partial\xi} \Big|_P \frac{\Delta\xi}{2} \right) - \max[\dot{m}_\xi, 0] \left(\phi_W + \frac{\partial\phi}{\partial\xi} \Big|_W \frac{\Delta\xi}{2} \right) \\ & + \max[\dot{m}_\eta, 0] \left(\phi_P + \frac{\partial\phi}{\partial\eta} \Big|_P \frac{\Delta\eta}{2} \right) - \max[-\dot{m}_\eta, 0] \left(\phi_N - \frac{\partial\phi}{\partial\eta} \Big|_N \frac{\Delta\eta}{2} \right) \\ & + \max[-\dot{m}_\eta, 0] \left(\phi_P - \frac{\partial\phi}{\partial\eta} \Big|_P \frac{\Delta\eta}{2} \right) - \max[\dot{m}_\eta, 0] \left(\phi_S + \frac{\partial\phi}{\partial\eta} \Big|_S \frac{\Delta\eta}{2} \right) \end{aligned} \quad (\text{J.9})$$

The values in Eq. (J.9) contain derivatives estimated at cell centers. These are calculated according to the methods presented earlier in this appendix. The terms in the parentheses of Eq. (J.7) are calculated from the previous iteration as denoted by the superscript o and can be included as source terms.

$$S_c \equiv \beta(F_H^o - F_L^o) \quad (\text{J.10})$$

Using Eqs. (J.7) – (J.10) in Eq. (J.6) gives

$$A_P \phi_P = A_E \phi_E + A_W \phi_W + A_N \phi_N + A_S \phi_S - S_c + \hat{S} \quad (\text{J.11})$$

where

$$\begin{aligned} A_E &= \max[-\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta \eta}{\Delta \xi} \Gamma_e J_e \\ A_W &= \max[\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta \eta}{\Delta \xi} \Gamma_w J_w \\ A_N &= \max[-\dot{m}_\eta, 0] + \left(\frac{z_{\xi}^2}{r} \right)_P \frac{\Delta \xi}{\Delta \eta} r_n \Gamma_n J_n \\ A_S &= \max[\dot{m}_\eta, 0] + \left(\frac{z_{\xi}^2}{r} \right)_P \frac{\Delta \xi}{\Delta \eta} r_s \Gamma_s J_s \\ A_P &= A_E + A_W + A_N + A_S \end{aligned} \quad (\text{J.12})$$

Solving Eq. (J.11) for ϕ_P gives

$$\phi_P = \frac{1}{A_P} (A_E \phi_E + A_W \phi_W + A_N \phi_N + A_S \phi_S - S_c + \hat{S}) \quad (\text{J.13})$$

Note that in the convection terms, the ϕ_P coefficients have been modified according to the equality

$$\begin{aligned} &(\max[\dot{m}_\xi, 0] + \max[-\dot{m}_\xi, 0] + \max[\dot{m}_\eta, 0] + \max[-\dot{m}_\eta, 0]) \phi_P \\ &= (\max[-\dot{m}_\xi, 0] + \max[\dot{m}_\xi, 0] + \max[-\dot{m}_\eta, 0] + \max[\dot{m}_\eta, 0]) \phi_P \end{aligned} \quad (\text{J.14})$$

in order to facilitate the summation in Eq. (J.12). Applying an under-relaxation variable, α , and the previous solution, ϕ_P^o , gives an under-relaxation form of Eq. (J.13),

$$\phi_P = (1 - \alpha) \phi_P^o + \frac{\alpha}{A_P} (A_E \phi_E + A_W \phi_W + A_N \phi_N + A_S \phi_S - S_c + \hat{S}) \quad (\text{J.15})$$

II. Boundary Condition Implementation

Two types of boundary conditions can be applied. A Dirichlet boundary condition exists when the value of the transport property is known at the boundary. A Neumann boundary condition exists when the gradient of the transport property is known at the boundary. The implementation of each of these conditions will be discussed here.

A. The Dirichlet Boundary Condition

For a cell adjacent to a boundary at which a Dirichlet boundary condition should be applied, the discretization method presented above is used, but the second-order differencing scheme for the boundary is employed. For example, using the approximation given in Eq. (H.42), the discretized form of Eq. (J.3) for a cell adjacent to the south boundary is

$$\begin{aligned} \dot{m}_\xi(\phi_e - \phi_w) + \dot{m}_\eta(\phi_n - \phi_s) = & (r_\eta^2)_P \left[\Gamma_e J_e \frac{\phi_E - \phi_P}{\Delta \xi} - \Gamma_w J_w \frac{\phi_P - \phi_W}{\Delta \xi} \right] \Delta \eta \\ & + \left(\frac{z_\xi^2}{r} \right)_P \left[r_n \Gamma_n J_n \frac{\phi_N - \phi_P}{\Delta \eta} - r_s \Gamma_s J_s \frac{-8\phi_S + 9\phi_P - \phi_N}{3\Delta \eta} \right] \Delta \xi + \hat{S} \end{aligned} \quad (J.16)$$

where ϕ_S is the value of the transport property at the south wall. Using the same methods as presented above for the convection terms, Eq. (J.16) can be rewritten to match Eq. (J.11) where

$$\begin{aligned} A_E &= \max[-\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta \eta}{\Delta \xi} \Gamma_e J_e \\ A_W &= \max[\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta \eta}{\Delta \xi} \Gamma_w J_w \\ A_N &= \max[-\dot{m}_\eta, 0] + \left(\frac{z_\xi^2}{r} \right)_P \frac{\Delta \xi}{\Delta \eta} \left(r_n \Gamma_n J_n + \frac{1}{3} r_s \Gamma_s J_s \right) \\ A_S &= \max[\dot{m}_\eta, 0] + \left(\frac{z_\xi^2}{r} \right)_P \frac{\Delta \xi}{\Delta \eta} \frac{8}{3} r_s \Gamma_s J_s \\ A_P &= A_E + A_W + A_N + A_S \end{aligned} \quad (J.17)$$

Likewise, for a cell adjacent to the north boundary,

$$\begin{aligned} A_E &= \max[-\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta \eta}{\Delta \xi} \Gamma_e J_e \\ A_W &= \max[\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta \eta}{\Delta \xi} \Gamma_w J_w \\ A_N &= \max[-\dot{m}_\eta, 0] + \left(\frac{z_\xi^2}{r} \right)_P \frac{\Delta \xi}{\Delta \eta} \frac{8}{3} r_n \Gamma_n J_n \\ A_S &= \max[\dot{m}_\eta, 0] + \left(\frac{z_\xi^2}{r} \right)_P \frac{\Delta \xi}{\Delta \eta} \left(r_s \Gamma_s J_s + \frac{1}{3} r_n \Gamma_n J_n \right) \\ A_P &= A_E + A_W + A_N + A_S \end{aligned} \quad (J.18)$$

For a cell adjacent to the east boundary,

$$\begin{aligned}
A_E &= \max[-\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta\eta}{\Delta\xi} \frac{8}{3} \Gamma_e J_e \\
A_W &= \max[\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta\eta}{\Delta\xi} \left(\Gamma_w J_w + \frac{1}{3} \Gamma_e J_e \right) \\
A_N &= \max[-\dot{m}_\eta, 0] + \left(\frac{z_{,\xi}^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} r_n \Gamma_n J_n \\
A_S &= \max[\dot{m}_\eta, 0] + \left(\frac{z_{,\xi}^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} r_s \Gamma_s J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{J.19}$$

For a cell adjacent to the west boundary

$$\begin{aligned}
A_E &= \max[-\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta\eta}{\Delta\xi} \left(\Gamma_e J_e + \frac{1}{3} \Gamma_w J_w \right) \\
A_W &= \max[\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta\eta}{\Delta\xi} \frac{8}{3} \Gamma_w J_w \\
A_N &= \max[-\dot{m}_\eta, 0] + \left(\frac{z_{,\xi}^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} r_n \Gamma_n J_n \\
A_S &= \max[\dot{m}_\eta, 0] + \left(\frac{z_{,\xi}^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} r_s \Gamma_s J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{J.20}$$

B. The Neumann Boundary Condition

For a cell adjacent to a boundary at which a Neumann boundary condition should be applied, the discretization method presented above is used, but the gradient along the boundary is directly applied. For example, if a Neumann condition is known at the south boundary, the discretized form of Eq. (J.3) for a cell adjacent to the south boundary is

$$\begin{aligned}
\dot{m}_\xi(\phi_e - \phi_w) + \dot{m}_\eta(\phi_n - \phi_s) &= (r_\eta^2)_P \left[\Gamma_e J_e \frac{\phi_E - \phi_P}{\Delta\xi} - \Gamma_w J_w \frac{\phi_P - \phi_W}{\Delta\xi} \right] \Delta\eta \\
&\quad + \left(\frac{z_{,\xi}^2}{r} \right)_P \left[r_n \Gamma_n J_n \frac{\phi_N - \phi_P}{\Delta\eta} - r_s \Gamma_s J_s \left(\frac{\partial\phi}{\partial\eta} \right)_s \right] \Delta\xi + \hat{S}
\end{aligned} \tag{J.21}$$

A common Neumann boundary condition is the zero gradient boundary condition, for example, at a symmetry boundary. For a zero gradient boundary condition on the south boundary,

$$\begin{aligned} \dot{m}_\xi(\phi_e - \phi_w) + \dot{m}_\eta(\phi_n - \phi_s) = & (r_\eta^2)_P \left[\Gamma_e J_e \frac{\phi_E - \phi_P}{\Delta\xi} - \Gamma_w J_w \frac{\phi_P - \phi_W}{\Delta\xi} \right] \Delta\eta \\ & + \left(\frac{z_\xi^2}{r} \right)_P \left[r_n \Gamma_n J_n \frac{\phi_N - \phi_P}{\Delta\eta} - 0 \right] \Delta\xi + \hat{S} \end{aligned} \quad (\text{J.22})$$

Using the same methods as presented above for the convection terms, Eq. (J.22) can be rewritten to match Eq. (J.11) where

$$\begin{aligned} A_E &= \max[-\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_e J_e \\ A_W &= \max[\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_w J_w \\ A_N &= \max[-\dot{m}_\eta, 0] + \left(\frac{z_\xi^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} r_n \Gamma_n J_n \\ A_S &= \max[\dot{m}_\eta, 0] \\ A_P &= A_E + A_W + A_N + A_S \end{aligned} \quad (\text{J.23})$$

Likewise, for a cell adjacent to the north boundary,

$$\begin{aligned} A_E &= \max[-\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_e J_e \\ A_W &= \max[\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_w J_w \\ A_N &= \max[-\dot{m}_\eta, 0] \\ A_S &= \max[\dot{m}_\eta, 0] + \left(\frac{z_\xi^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} r_s \Gamma_s J_s \\ A_P &= A_E + A_W + A_N + A_S \end{aligned} \quad (\text{J.24})$$

For a cell adjacent to the east boundary,

$$\begin{aligned} A_E &= \max[-\dot{m}_\xi, 0] \\ A_W &= \max[\dot{m}_\xi, 0] + (r_\eta^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_w J_w \\ A_N &= \max[-\dot{m}_\eta, 0] + \left(\frac{z_\xi^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} r_n \Gamma_n J_n \\ A_S &= \max[\dot{m}_\eta, 0] + \left(\frac{z_\xi^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} r_s \Gamma_s J_s \\ A_P &= A_E + A_W + A_N + A_S \end{aligned} \quad (\text{J.25})$$

For a cell adjacent to the west boundary,

$$\begin{aligned}
A_E &= \max[-\dot{m}_\xi, 0] + (r_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_e J_e \\
A_W &= \max[\dot{m}_\xi, 0] \\
A_N &= \max[-\dot{m}_\eta, 0] + \left(\frac{z_{,\xi}^2}{r}\right)_P \frac{\Delta\xi}{\Delta\eta} r_n \Gamma_n J_n \\
A_S &= \max[\dot{m}_\eta, 0] + \left(\frac{z_{,\xi}^2}{r}\right)_P \frac{\Delta\xi}{\Delta\eta} r_s \Gamma_s J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{J.26}$$

Another common Neumann boundary condition is the constant gradient boundary condition where the gradient of the transport property normal to the boundary is assumed constant over the cell. This condition is sometimes used at a pressure boundary. For a constant gradient boundary condition on the north boundary of the domain, the gradient on the north face of each cell along the boundary is equal to the gradient on the south face of the cell.

$$\begin{aligned}
\left. \frac{\partial\phi}{\partial r} \right|_n &= \left. \frac{\partial\phi}{\partial r} \right|_s \\
\left(J_{z,\xi} \frac{\partial\phi}{\partial\eta} \right)_n &= \left(J_{z,\xi} \frac{\partial\phi}{\partial\eta} \right)_s
\end{aligned} \tag{J.27}$$

This can be applied to give the discretized form of Eq. (J.3) for a cell adjacent to a north boundary

$$\begin{aligned}
\dot{m}_\xi(\phi_e - \phi_w) + \dot{m}_\eta(\phi_n - \phi_s) &= (r_{,\eta}^2)_P \left[\Gamma_e J_e \frac{\phi_e - \phi_P}{\Delta\xi} - \Gamma_w J_w \frac{\phi_P - \phi_w}{\Delta\xi} \right] \Delta\eta \\
&+ \left(\frac{z_{,\xi}^2}{r}\right)_P \left[(r_n \Gamma_n - r_s \Gamma_s) J_s \frac{\phi_P - \phi_S}{\Delta\eta} \right] \Delta\xi + \hat{S}
\end{aligned} \tag{J.28}$$

Using the same methods as presented above for the convection terms, Eq. (J.28) can be rewritten to match Eq. (J.11) where

$$\begin{aligned}
A_E &= \max[-\dot{m}_\xi, 0] + (r_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_e J_e \\
A_W &= \max[\dot{m}_\xi, 0] + (r_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_w J_w \\
A_N &= \max[-\dot{m}_\eta, 0] \\
A_S &= \max[\dot{m}_\eta, 0] + \left(\frac{z_{,\xi}^2}{r}\right)_P \frac{\Delta\xi}{\Delta\eta} (r_s \Gamma_s - r_n \Gamma_n) J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{J.29}$$

For a cell adjacent to a south boundary with a constant gradient boundary condition,

$$\begin{aligned}
A_E &= \max[-\dot{m}_\xi, 0] + (r_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_e J_e \\
A_W &= \max[\dot{m}_\xi, 0] + (r_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} \Gamma_w J_w \\
A_N &= \max[-\dot{m}_\eta, 0] + \left(\frac{z_{,\xi}^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} (r_n \Gamma_n - r_s \Gamma_s) J_n \\
A_S &= \max[\dot{m}_\eta, 0] \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{J.30}$$

For a cell adjacent to an east constant gradient boundary,

$$\begin{aligned}
A_E &= \max[-\dot{m}_\xi, 0] \\
A_W &= \max[\dot{m}_\xi, 0] + (r_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} (\Gamma_w - \Gamma_e) J_w \\
A_N &= \max[-\dot{m}_\eta, 0] + \left(\frac{z_{,\xi}^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} r_n \Gamma_n J_n \\
A_S &= \max[\dot{m}_\eta, 0] + \left(\frac{z_{,\xi}^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} r_s \Gamma_s J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{J.31}$$

For a cell adjacent to a west constant gradient boundary,

$$\begin{aligned}
A_E &= \max[-\dot{m}_\xi, 0] + (r_{,\eta}^2)_P \frac{\Delta\eta}{\Delta\xi} (\Gamma_e - \Gamma_w) J_e \\
A_W &= \max[\dot{m}_\xi, 0] \\
A_N &= \max[-\dot{m}_\eta, 0] + \left(\frac{z_{,\xi}^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} r_n \Gamma_n J_n \\
A_S &= \max[\dot{m}_\eta, 0] + \left(\frac{z_{,\xi}^2}{r} \right)_P \frac{\Delta\xi}{\Delta\eta} r_s \Gamma_s J_s \\
A_P &= A_E + A_W + A_N + A_S
\end{aligned} \tag{J.32}$$

For the cases of interest in this work, the boundary conditions for a velocity inlet, outlet, smooth-wall, symmetry, and pressure outlet are the same as for the Cartesian formulation. However, the pressure inlet condition across the north side of the domain requires special attention.

It is assumed that at a pressure inlet, the flow enters the domain normal to the boundary and the properties of the flow are known across the boundary. If the north side of a domain is specified as a

pressure inlet boundary, flow must enter the domain without any z-velocity component. Continuity requires that

$$\frac{d\bar{V}_r}{dr} = -\frac{\bar{V}_r}{r} \quad (\text{J.33})$$

Applying Eq. (G.10) gives this relationship in the transformed coordinate system

$$\frac{\partial \bar{V}_r}{\partial r} = J_{z,\xi} \frac{\partial \bar{V}_r}{\partial \eta} = -\frac{\bar{V}_r}{r} \quad (\text{J.34})$$

Applying Eq. (H.22) across the north side of the domain gives

$$\begin{aligned} J_{z,\xi} \frac{\partial \bar{V}_r}{\partial \eta} &= -\frac{\bar{V}_r}{r} \\ J_{n^z,\xi_n} \frac{8\bar{V}_{r_N} - 9\bar{V}_{r_P} + \bar{V}_{r_S}}{3\Delta\eta} &= \frac{\bar{V}_{r_N}}{r_N} \end{aligned} \quad (\text{J.35})$$

Rearranging gives

$$\bar{V}_{r_N} = \frac{J_{n^z,\xi_n} (9\bar{V}_{r_P} - \bar{V}_{r_S})}{\left(\frac{J_{n^z,\xi_n}}{3\Delta\eta} 8 - \frac{1}{r} \right)} \quad (\text{J.36})$$

Using that in this:

$$\begin{aligned} \dot{m}_\xi(\phi_e - \phi_w) + \dot{m}_\eta(\phi_n - \phi_s) &= (r,\eta)_P \left[\Gamma_e J_e \frac{\phi_E - \phi_P}{\Delta\xi} - \Gamma_w J_w \frac{\phi_P - \phi_W}{\Delta\xi} \right] \Delta\eta \\ &+ \left(\frac{z,\xi}{r} \right)_P \left[r_n \Gamma_n J_n \frac{8\phi_N - 9\phi_P + \phi_S}{3\Delta\eta} - r_s \Gamma_s J_s \frac{\phi_P - \phi_S}{\Delta\eta} \right] \Delta\xi + \hat{S} \end{aligned} \quad (\text{J.37})$$

gives

$$\begin{aligned} \dot{m}_\xi(\phi_e - \phi_w) + \dot{m}_\eta(\phi_n - \phi_s) &= (r,\eta)_P \left[\Gamma_e J_e \frac{\phi_E - \phi_P}{\Delta\xi} - \Gamma_w J_w \frac{\phi_P - \phi_W}{\Delta\xi} \right] \Delta\eta \\ &+ \left(\frac{z,\xi}{r} \right)_P \left[r_n \Gamma_n J_n (1/C - 1) \frac{9\phi_P - \phi_S}{3\Delta\eta} - r_s \Gamma_s J_s \frac{\phi_P - \phi_S}{\Delta\eta} \right] \Delta\xi + \hat{S} \end{aligned} \quad (\text{J.38})$$

where

$$C = \left(1 - \frac{1}{8r_n \frac{J_n^{z,\xi_n}}{3\Delta\eta}} \right) \quad (\text{J.39})$$

It is recommended that the developer use a Dirichlet boundary condition for both velocity components on the north face and update it every iteration according to

$$\begin{aligned} \bar{V}_{r_n} &= \frac{\frac{J_n^{z,\xi_n}}{3\Delta\eta} (9\bar{V}_{r_p} - \bar{V}_{r_s})}{\left(\frac{J_n^{z,\xi_n}}{3\Delta\eta} 8 - \frac{1}{r_n} \right)} \\ \bar{V}_{z_n} &= 0 \end{aligned} \quad (\text{J.40})$$

All other transport properties must be specified at a pressure inlet boundary, and a Dirichlet condition is applied for these transport properties.

III. Transport Equations

The z -momentum equation is a special case of Eq. (J.1) where $\phi = \bar{V}_z$ and $\Gamma = \nu + \nu_r$. The r -momentum equation is a special case of Eq. (J.1) where $\phi = \bar{V}_r$ and $\Gamma = \nu + \nu_r$. The z - and r -momentum source terms include pressure source terms and viscous source terms and can be written in the transformed coordinate system as

$$\begin{aligned} S_z &= -Jr_{,\eta} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \xi} + \frac{1}{r} J \frac{\partial}{\partial \eta} \left(\Gamma r \frac{\partial \bar{V}_r}{\partial \xi} \right) + Jr_{,\eta}^2 \frac{\partial}{\partial \xi} \left(\Gamma J \frac{\partial \bar{V}_z}{\partial \xi} \right) \\ S_r &= -Jz_{,\xi} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \eta} + \frac{1}{r} Jz_{,\xi}^2 \frac{\partial}{\partial \eta} \left(\Gamma r J \frac{\partial \bar{V}_r}{\partial \eta} \right) + J \frac{\partial}{\partial \xi} \left(\Gamma \frac{\partial \bar{V}_z}{\partial \eta} \right) - \frac{2\Gamma \bar{V}_r}{r^2} \end{aligned} \quad (\text{J.41})$$

Integrating about a rectangular control volume in the ξ and η directions gives

$$\begin{aligned} \hat{S}_z &= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left(\frac{S_z}{J} \right) d\xi d\eta \\ &= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left\{ -r_{,\eta} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \xi} + \frac{1}{r} \frac{\partial}{\partial \eta} \left(\Gamma r \frac{\partial \bar{V}_r}{\partial \xi} \right) + r_{,\eta}^2 \frac{\partial}{\partial \xi} \left(\Gamma J \frac{\partial \bar{V}_z}{\partial \xi} \right) \right\} d\xi d\eta \\ \hat{S}_r &= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left(\frac{S_r}{J} \right) d\xi d\eta \\ &= \int_{-\Delta\eta/2}^{\Delta\eta/2} \int_{-\Delta\xi/2}^{\Delta\xi/2} \left\{ -z_{,\xi} \frac{1}{\rho} \frac{\partial \hat{p}}{\partial \eta} + \frac{1}{r} z_{,\xi}^2 \frac{\partial}{\partial \eta} \left(\Gamma r J \frac{\partial \bar{V}_r}{\partial \eta} \right) + \frac{\partial}{\partial \xi} \left(\Gamma \frac{\partial \bar{V}_z}{\partial \eta} \right) - \frac{2\Gamma \bar{V}_r}{Jr^2} \right\} d\xi d\eta \end{aligned} \quad (\text{J.42})$$

which can be approximated as

$$\begin{aligned}\hat{S}_z &= -r_{,\eta} \frac{1}{\rho} (\hat{p}) \Big|_w^e \Delta\eta + r_{,\eta}^2 \left(\Gamma J \frac{\partial \bar{V}_z}{\partial \xi} \right) \Big|_w^e \Delta\eta + \frac{1}{r} \left(\Gamma r \frac{\partial \bar{V}_r}{\partial \xi} \right) \Big|_s^n \Delta\xi \\ \hat{S}_r &= -z_{,\xi} \frac{1}{\rho} (\hat{p}) \Big|_s^n \Delta\xi + \frac{1}{r} z_{,\xi}^2 \left(\Gamma r J \frac{\partial \bar{V}_r}{\partial \eta} \right) \Big|_s^n \Delta\xi + \left(\Gamma \frac{\partial \bar{V}_z}{\partial \eta} \right) \Big|_w^e \Delta\eta - \frac{2\Gamma \bar{V}_r}{Jr^2} \Delta\xi \Delta\eta\end{aligned}\quad (\text{J.43})$$

and rearranged to give

$$\begin{aligned}\hat{S}_z &= r_{,\eta} \frac{1}{\rho} (\hat{p}_w - \hat{p}_e) \Delta\eta + r_{,\eta}^2 \left[\Gamma_e J_e \left(\frac{\partial \bar{V}_z}{\partial \xi} \right)_e - \Gamma_w J_w \left(\frac{\partial \bar{V}_z}{\partial \xi} \right)_w \right] \Delta\eta \\ &\quad + \frac{1}{r} \left[\Gamma_n r_n \left(\frac{\partial \bar{V}_r}{\partial \xi} \right)_n - \Gamma_s r_s \left(\frac{\partial \bar{V}_r}{\partial \xi} \right)_s \right] \Delta\xi \\ \hat{S}_r &= z_{,\xi} \frac{1}{\rho} (\hat{p}_s - \hat{p}_n) \Delta\xi + \frac{1}{r} z_{,\xi}^2 \left[\Gamma_n r_n J_n \left(\frac{\partial \bar{V}_r}{\partial \eta} \right)_n - \Gamma_s r_s J_s \left(\frac{\partial \bar{V}_r}{\partial \eta} \right)_s \right] \Delta\xi \\ &\quad + \left[\Gamma_e \left(\frac{\partial \bar{V}_z}{\partial \eta} \right)_e - \Gamma_w \left(\frac{\partial \bar{V}_z}{\partial \eta} \right)_w \right] \Delta\eta - \frac{2\Gamma \bar{V}_r}{Jr^2} \Delta\xi \Delta\eta\end{aligned}\quad (\text{J.44})$$

These derivatives are discretized according to the discretization methods presented in Appendix H.

The same no-slip wall boundary conditions apply as in Cartesian coordinates, and the wall shear stress is estimated the same way. For example, the shear stress can be written in the computational domain along the south boundary as

$$\begin{aligned}\tau_w &\equiv \left[\rho(v + v_t) \frac{\partial \bar{V}_z}{\partial r} \right]_{r=0} \\ &= \left[\rho(v + v_t) \eta_{,r} \frac{\partial \bar{V}_z}{\partial \eta} \right]_{r=0} \\ &= \left[\rho(v + v_t) J_{z,\xi} \frac{\partial \bar{V}_z}{\partial \eta} \right]_{r=0}\end{aligned}\quad (\text{J.45})$$

Applying Eq. (H.42) gives the discretized form

$$\tau_w = \rho(v + v_t)_s J_{s,z,\xi_s} \frac{-8\bar{V}_{z_s} + 9\bar{V}_{z_p} - \bar{V}_{z_N}}{3\Delta\eta}\quad (\text{J.46})$$

Likewise, along a north boundary the shear stress can be calculated as

$$\tau_w = \rho(v + v_t)_n J_{n,z,\xi_n} \frac{8\bar{V}_{z_N} - 9\bar{V}_{z_p} + \bar{V}_{z_s}}{3\Delta\eta}\quad (\text{J.47})$$

Along an east boundary the shear stress can be calculated as

$$\tau_w = \rho(v + v_t)_e J_e r_{,\eta_e} \frac{8\bar{V}_{r_E} - 9\bar{V}_{r_P} + \bar{V}_{r_W}}{3\Delta\xi} \quad (\text{J.48})$$

Along a west boundary the shear stress can be calculated as

$$\tau_w = \rho(v + v_t)_w J_w r_{,\eta_w} \frac{-8\bar{V}_{r_W} + 9\bar{V}_{r_P} - \bar{V}_{r_E}}{3\Delta\xi} \quad (\text{J.49})$$

The source terms and boundary conditions for the other transport equations in cylindrical coordinates are identical to those in Cartesian coordinates.

APPENDIX K

THE SIMPLE ALGORITHM IN CARTESIAN COORDINATES

I. Development of the SIMPLE Algorithm

The conservation of momentum equations can be written for any cell as

$$\begin{aligned} A_p \bar{V}_{x_p} &= A_E \bar{V}_{x_E} + A_W \bar{V}_{x_W} + A_N \bar{V}_{x_N} + A_S \bar{V}_{x_S} + \hat{S}_x - S_{c_x} \\ A_p \bar{V}_{y_p} &= A_E \bar{V}_{y_E} + A_W \bar{V}_{y_W} + A_N \bar{V}_{y_N} + A_S \bar{V}_{y_S} + \hat{S}_y - S_{c_y} \end{aligned} \quad (\text{K.1})$$

Neglecting all source terms other than pressure, these equations can be rewritten as

$$\begin{aligned} A_p \bar{V}_{x_p} &= \sum_{nb} A_{nb} \bar{V}_{x_{nb}} + \frac{y_{, \eta}}{\rho} (\hat{p}_w - \hat{p}_e) \Delta \eta \\ A_p \bar{V}_{y_p} &= \sum_{nb} A_{nb} \bar{V}_{y_{nb}} + \frac{x_{, \xi}}{\rho} (\hat{p}_s - \hat{p}_n) \Delta \xi \end{aligned} \quad (\text{K.2})$$

where the subscript, nb , stands for the neighboring boundaries, N, S, E, and W. At some point during the solution procedure, approximate values for the pressure and velocities are known. The actual values of the pressure and velocities can be written as

$$\begin{aligned} \bar{V}_x &= \bar{V}_x^* + \bar{V}'_x \\ \bar{V}_y &= \bar{V}_y^* + \bar{V}'_y \\ \hat{p} &= \hat{p}^* + \hat{p}' \end{aligned} \quad (\text{K.3})$$

where the starred values indicate the present approximate or “guessed” values and the prime values indicate correction factors. Therefore, at any given time during the solution procedure, instead of Eq. (K.2) we actually have

$$\begin{aligned} A_p \bar{V}_{x_p}^* &= \sum_{nb} A_{nb} \bar{V}_{x_{nb}}^* + \frac{y_{, \eta}}{\rho} (\hat{p}_w^* - \hat{p}_e^*) \Delta \eta \\ A_p \bar{V}_{y_p}^* &= \sum_{nb} A_{nb} \bar{V}_{y_{nb}}^* + \frac{x_{, \xi}}{\rho} (\hat{p}_s^* - \hat{p}_n^*) \Delta \xi \end{aligned} \quad (\text{K.4})$$

Subtracting Eq. (K.4) from Eq. (K.2) gives the momentum equations in terms of the correction factors

$$\begin{aligned}
A_P \bar{V}'_{x_P} &= \sum_{nb} A_{nb} \bar{V}'_{x_{nb}} + \frac{y_{,\eta}}{\rho} (\hat{p}'_w - \hat{p}'_e) \Delta \eta \\
A_P \bar{V}'_{y_P} &= \sum_{nb} A_{nb} \bar{V}'_{y_{nb}} + \frac{x_{,\xi}}{\rho} (\hat{p}'_s - \hat{p}'_n) \Delta \xi
\end{aligned} \tag{K.5}$$

Neglecting the summation terms gives expressions for the velocity correction terms

$$\begin{aligned}
\bar{V}'_{x_P} &= \frac{y_{,\eta} \Delta \eta}{A_P \rho} (\hat{p}'_w - \hat{p}'_e) \\
\bar{V}'_{y_P} &= \frac{x_{,\xi} \Delta \xi}{A_P \rho} (\hat{p}'_s - \hat{p}'_n)
\end{aligned} \tag{K.6}$$

From the definition of the contravariant velocity components given in Eq. (F.12), the correction to the contravariant velocity components can be written in terms of the correction to the physical velocity components.

$$\begin{aligned}
\bar{V}'_{\xi_P} &= y_{,\eta} \bar{V}'_{x_P} \\
\bar{V}'_{\eta_P} &= x_{,\xi} \bar{V}'_{y_P}
\end{aligned} \tag{K.7}$$

Using Eq. (K.6) in Eq. (K.7) gives

$$\begin{aligned}
\bar{V}'_{\xi_P} &= \frac{y_{,\eta}^2 \Delta \eta}{A_P \rho} (\hat{p}'_w - \hat{p}'_e) \\
\bar{V}'_{\eta_P} &= \frac{x_{,\xi}^2 \Delta \xi}{A_P \rho} (\hat{p}'_s - \hat{p}'_n)
\end{aligned} \tag{K.8}$$

Equation (K.8) is a velocity correction equation for the cell center in terms of pressure values at the faces.

Using a similar method to that presented here in Eqs. (K.1) through (K.8), expressions can be written for the velocity correction on the cell faces in terms of pressure values at cell centers for any cell face that is not a boundary.

$$\begin{aligned}
\bar{V}'_{\xi_e} &= \frac{y_{,\eta}^2 \Delta \eta}{A_e \rho} (\hat{p}'_P - \hat{p}'_E) \\
\bar{V}'_{\xi_w} &= \frac{y_{,\eta}^2 \Delta \eta}{A_w \rho} (\hat{p}'_W - \hat{p}'_P) \\
\bar{V}'_{\eta_n} &= \frac{x_{,\xi}^2 \Delta \xi}{A_n \rho} (\hat{p}'_P - \hat{p}'_N) \\
\bar{V}'_{\eta_s} &= \frac{x_{,\xi}^2 \Delta \xi}{A_s \rho} (\hat{p}'_S - \hat{p}'_P)
\end{aligned} \tag{K.9}$$

Now consider the average continuity equation for incompressible flow

$$\nabla \cdot \bar{\mathbf{V}} = 0 \quad (\text{K.10})$$

For two-dimensional flow, this can be written in curvilinear coordinates as

$$\frac{\partial \bar{V}_\xi}{\partial \xi} + \frac{\partial \bar{V}_\eta}{\partial \eta} = 0 \quad (\text{K.11})$$

Integrating around the boundary of a cell gives

$$\iint \left[\frac{\partial \bar{V}_\xi}{\partial \xi} + \frac{\partial \bar{V}_\eta}{\partial \eta} = 0 \right] d\xi d\eta \quad (\text{K.12})$$

$$\Delta \eta (\bar{V}_{\xi_e} - \bar{V}_{\xi_w}) + \Delta \xi (\bar{V}_{\eta_n} - \bar{V}_{\eta_s}) = 0$$

Using Eq. (K.3) in Eq. (K.12) gives

$$\Delta \eta (\bar{V}_{\xi_e}^* + \bar{V}_{\xi_e}' - \bar{V}_{\xi_w}^* - \bar{V}_{\xi_w}') + \Delta \xi (\bar{V}_{\eta_n}^* + \bar{V}_{\eta_n}' - \bar{V}_{\eta_s}^* - \bar{V}_{\eta_s}') = 0 \quad (\text{K.13})$$

Rearranging yields

$$\Delta \eta (\bar{V}_{\xi_e}' - \bar{V}_{\xi_w}') + \Delta \xi (\bar{V}_{\eta_n}' - \bar{V}_{\eta_s}') + S_p = 0 \quad (\text{K.14})$$

where

$$S_p = \Delta \eta (\bar{V}_{\xi_e}^* - \bar{V}_{\xi_w}^*) + \Delta \xi (\bar{V}_{\eta_n}^* - \bar{V}_{\eta_s}^*) \quad (\text{K.15})$$

is the imbalance of mass in the cell and approaches zero as the solution converges. Notice that this mass imbalance equation (the source term in the pressure solver) is dependent on knowing the contravariant velocity components at the faces of the cell. Obtaining these velocities through interpolations from the cell center can result in pressure oscillations for collocated grid arrangements. Employing a staggered grid arrangement eliminates this problem. To remedy the pressure oscillation problem on collocated grids, Rhie and Chow [80] suggest calculating the velocities at the cell faces by adding in some additional factors. This method has been widely adopted for collocated grids, and will be used here. The velocities at the cell faces are calculated from

$$\begin{aligned}
\bar{V}_{\xi_e}^* &= \bar{\tilde{V}}_{\xi_e}^* - \frac{y_{,\eta}^2}{A_e \rho} \Delta \xi \Delta \eta \left(\frac{p_E^* - p_P^*}{\delta \xi} - \bar{\tilde{p}}_{,\xi}^* \right) \\
\bar{V}_{\xi_w}^* &= \bar{\tilde{V}}_{\xi_w}^* - \frac{y_{,\eta}^2}{A_w \rho} \Delta \xi \Delta \eta \left(\frac{p_P^* - p_W^*}{\delta \xi} - \bar{\tilde{p}}_{,\xi}^* \right) \\
\bar{V}_{\eta_n}^* &= \bar{\tilde{V}}_{\eta_n}^* - \frac{x_{,\xi}^2}{A_n \rho} \Delta \xi \Delta \eta \left(\frac{p_N^* - p_P^*}{\delta \eta} - \bar{\tilde{p}}_{,\eta}^* \right) \\
\bar{V}_{\eta_s}^* &= \bar{\tilde{V}}_{\eta_s}^* - \frac{x_{,\xi}^2}{A_s \rho} \Delta \xi \Delta \eta \left(\frac{p_P^* - p_S^*}{\delta \eta} - \bar{\tilde{p}}_{,\eta}^* \right)
\end{aligned} \tag{K.16}$$

where the curved bar represents an interpolated value to the cell face.

Using Eq. (K.9) in Eq. (K.14) gives

$$\begin{aligned}
&\frac{y_{,\eta}^2 \Delta \eta^2}{A_e \rho} (\hat{p}_P - \hat{p}_E) - \frac{y_{,\eta}^2 \Delta \eta^2}{A_w \rho} (\hat{p}_W - \hat{p}_P) \\
&+ \frac{x_{,\xi}^2 \Delta \xi^2}{A_n \rho} (\hat{p}_P - \hat{p}_N) - \frac{x_{,\xi}^2 \Delta \xi^2}{A_s \rho} (\hat{p}_S - \hat{p}_P) + S_p = 0
\end{aligned} \tag{K.17}$$

Collecting terms and rearranging gives

$$D_P \hat{p}_P = D_E \hat{p}_E + D_W \hat{p}_W + D_N \hat{p}_N + D_S \hat{p}_S - S_p \tag{K.18}$$

where

$$\begin{aligned}
D_E &\equiv \frac{y_{,\eta}^2 \Delta \eta^2}{A_e \rho} \\
D_W &\equiv \frac{y_{,\eta}^2 \Delta \eta^2}{A_w \rho} \\
D_N &\equiv \frac{x_{,\xi}^2 \Delta \xi^2}{A_n \rho} \\
D_S &\equiv \frac{x_{,\xi}^2 \Delta \xi^2}{A_s \rho} \\
D_P &\equiv D_E + D_W + D_N + D_S
\end{aligned} \tag{K.19}$$

Therefore,

$$\hat{p}_P = \frac{1}{D_P} (D_E \hat{p}_E + D_W \hat{p}_W + D_N \hat{p}_N + D_S \hat{p}_S - S_p) \tag{K.20}$$

Note that Eq. (K.20) is linear and therefore requires no relaxation factor. Once the pressure correction terms are converged, they are added to the previous pressure terms using an under-relaxation factor, α_p

$$\hat{p}_P = \hat{p}_P^* + \alpha_p \hat{p}_P' \quad (\text{K.21})$$

Using Eq. (K.9), the contravariant velocity terms are then corrected without relaxation (the under-relaxation for the velocity terms takes place in the momentum solver).

$$\begin{aligned} \bar{V}_{\xi_e} &= \bar{V}_{\xi_e}^* + \frac{y_{,\eta}^2 \Delta \eta}{A_e \rho} (\hat{p}_P' - \hat{p}_E') \\ \bar{V}_{\xi_w} &= \bar{V}_{\xi_w}^* + \frac{y_{,\eta}^2 \Delta \eta}{A_w \rho} (\hat{p}_W' - \hat{p}_P') \\ \bar{V}_{\eta_n} &= \bar{V}_{\eta_n}^* + \frac{x_{,\xi}^2 \Delta \xi}{A_n \rho} (\hat{p}_P' - \hat{p}_N') \\ \bar{V}_{\eta_s} &= \bar{V}_{\eta_s}^* + \frac{x_{,\xi}^2 \Delta \xi}{A_s \rho} (\hat{p}_S' - \hat{p}_P') \end{aligned} \quad (\text{K.22})$$

These velocities are now mass-conserving velocities and can be averaged to find the contravariant velocity components at the center of the cell. The contravariant velocity components are then used to calculate the corrected velocities in rectilinear coordinates using Eq. (F.12).

II. Boundary Treatment

Several different types of boundary conditions exist for pressure, however, they all require similar treatment. If the velocity at the boundary is fixed, then the velocity need not be updated, and the corresponding last term on the right-hand side of Eq. (K.22) is zero. Likewise, if the pressure gradient normal to the boundary is zero, then the corresponding last term on the right-hand side of Eq. (K.22) is zero. For example, if the east boundary is an outlet, the pressure gradient is zero and Eq. (K.20) can be solved by setting $D_E = 0$. The same would be true if the east boundary were a velocity inlet (velocity specified), a wall (velocity = 0, and the normal pressure gradient is zero), or a line of symmetry (normal pressure gradient is zero).

APPENDIX L

THE SIMPLE ALGORITHM IN CYLINDRICAL COORDINATES

I. Development of the SIMPLE Algorithm

The conservation of momentum equations can be written for any cell as

$$\begin{aligned} A_P \bar{V}_{z_P} &= A_E \bar{V}_{z_E} + A_W \bar{V}_{z_W} + A_N \bar{V}_{z_N} + A_S \bar{V}_{z_S} + \hat{S}_z - S_{c_z} \\ A_P \bar{V}_{r_P} &= A_E \bar{V}_{r_E} + A_W \bar{V}_{r_W} + A_N \bar{V}_{r_N} + A_S \bar{V}_{r_S} + \hat{S}_r - S_{c_r} \end{aligned} \quad (\text{L.1})$$

Neglecting all source terms other than pressure, these equations can be rewritten as

$$\begin{aligned} A_P \bar{V}_{z_P} &= \sum_{nb} A_{nb} \bar{V}_{z_{nb}} + \frac{r_{, \eta}}{\rho} (\hat{p}_w - \hat{p}_e) \Delta \eta \\ A_P \bar{V}_{r_P} &= \sum_{nb} A_{nb} \bar{V}_{r_{nb}} + \frac{z_{, \xi}}{\rho} (\hat{p}_s - \hat{p}_n) \Delta \xi \end{aligned} \quad (\text{L.2})$$

where the subscript, nb , stands for the neighboring boundaries, N, S, E, and W. At some point during the solution procedure, approximate values for the pressure and velocities are known. The actual values of the pressure and velocities can be written as

$$\begin{aligned} \bar{V}_z &= \bar{V}_z^* + \bar{V}_z' \\ \bar{V}_r &= \bar{V}_r^* + \bar{V}_r' \\ \hat{p} &= \hat{p}^* + \hat{p}' \end{aligned} \quad (\text{L.3})$$

where the starred values indicate the present approximate or “guessed” values and the prime values indicate correction factors. Therefore, at any given time during the solution procedure, instead of Eq. (L.2) we actually have

$$\begin{aligned} A_P \bar{V}_{z_P}^* &= \sum_{nb} A_{nb} \bar{V}_{z_{nb}}^* + \frac{r_{, \eta}}{\rho} (\hat{p}_w^* - \hat{p}_e^*) \Delta \eta \\ A_P \bar{V}_{r_P}^* &= \sum_{nb} A_{nb} \bar{V}_{r_{nb}}^* + \frac{z_{, \xi}}{\rho} (\hat{p}_s^* - \hat{p}_n^*) \Delta \xi \end{aligned} \quad (\text{L.4})$$

Subtracting Eq. (L.4) from Eq. (L.2) gives the momentum equations in terms of the correction factors

$$\begin{aligned}
A_P \bar{V}'_{z_P} &= \sum_{nb} A_{nb} \bar{V}'_{z_{nb}} + \frac{r_{,\eta}}{\rho} (\hat{p}'_w - \hat{p}'_e) \Delta \eta \\
A_P \bar{V}'_{r_P} &= \sum_{nb} A_{nb} \bar{V}'_{r_{nb}} + \frac{z_{,\xi}}{\rho} (\hat{p}'_s - \hat{p}'_n) \Delta \xi
\end{aligned}
\tag{L.5}$$

Neglecting the summation terms gives expressions for the velocity correction terms

$$\begin{aligned}
\bar{V}'_{z_P} &= \frac{r_{,\eta} \Delta \eta}{A_P \rho} (\hat{p}'_w - \hat{p}'_e) \\
\bar{V}'_{r_P} &= \frac{z_{,\xi} \Delta \xi}{A_P \rho} (\hat{p}'_s - \hat{p}'_n)
\end{aligned}
\tag{L.6}$$

From the definition of the contravariant velocity components given in Eq. (G.12), the correction to the contravariant velocity components can be written in terms of the correction to the physical velocity components.

$$\begin{aligned}
\bar{V}'_{\xi_P} &= r_{,\eta} \bar{V}'_{z_P} \\
\bar{V}'_{\eta_P} &= z_{,\xi} \bar{V}'_{r_P}
\end{aligned}
\tag{L.7}$$

Using Eq. (L.6) in Eq. (L.7) gives

$$\begin{aligned}
\bar{V}'_{\xi_P} &= \frac{r_{,\eta}^2 \Delta \eta}{A_P \rho} (\hat{p}'_w - \hat{p}'_e) \\
\bar{V}'_{\eta_P} &= \frac{z_{,\xi}^2 \Delta \xi}{A_P \rho} (\hat{p}'_s - \hat{p}'_n)
\end{aligned}
\tag{L.8}$$

Equation (L.8) is a velocity correction equation for the cell center in terms of pressure values at the faces.

Using a similar method to that presented here in Eqs. (L.1) through (L.8), expressions can be written for the velocity correction on the cell faces in terms of pressure values at cell centers for any cell face that is not a boundary.

$$\begin{aligned}
\bar{V}'_{\xi_e} &= \frac{r_{,\eta}^2 \Delta \eta}{A_e \rho} (\hat{p}'_P - \hat{p}'_E) \\
\bar{V}'_{\xi_w} &= \frac{r_{,\eta}^2 \Delta \eta}{A_w \rho} (\hat{p}'_W - \hat{p}'_P) \\
\bar{V}'_{\eta_n} &= \frac{z_{,\xi}^2 \Delta \xi}{A_n \rho} (\hat{p}'_P - \hat{p}'_N) \\
\bar{V}'_{\eta_s} &= \frac{z_{,\xi}^2 \Delta \xi}{A_s \rho} (\hat{p}'_S - \hat{p}'_P)
\end{aligned}
\tag{L.9}$$

Now consider the average continuity equation for incompressible flow

$$\nabla \cdot \bar{\mathbf{V}} = 0 \quad (\text{L.10})$$

For two-dimensional axisymmetric flow, this can be written in curvilinear coordinates as

$$\frac{1}{r} \frac{\partial(r\bar{V}_\eta)}{\partial\eta} + \frac{\partial\bar{V}_\xi}{\partial\xi} = 0 \quad (\text{L.11})$$

Integrating around the boundary of a cell gives

$$\iint \left[\frac{1}{r} \frac{\partial(r\bar{V}_\eta)}{\partial\eta} + \frac{\partial\bar{V}_\xi}{\partial\xi} = 0 \right] d\xi d\eta \quad (\text{L.12})$$

$$\Delta\eta(\bar{V}_{\xi_e} - \bar{V}_{\xi_w}) + \frac{\Delta\xi}{r_p}(r_n\bar{V}_{\eta_n} - r_s\bar{V}_{\eta_s}) = 0$$

Using Eq. (L.3) in Eq. (L.12) gives

$$\Delta\eta(\bar{V}_{\xi_e}^* + \bar{V}_{\xi_e}' - \bar{V}_{\xi_w}^* - \bar{V}_{\xi_w}') + \frac{\Delta\xi}{r_p}(r_n\bar{V}_{\eta_n}^* + r_n\bar{V}_{\eta_n}' - r_s\bar{V}_{\eta_s}^* - r_s\bar{V}_{\eta_s}') = 0 \quad (\text{L.13})$$

Rearranging yields

$$\Delta\eta(\bar{V}_{\xi_e}' - \bar{V}_{\xi_w}') + \frac{\Delta\xi}{r_p}(r_n\bar{V}_{\eta_n}' - r_s\bar{V}_{\eta_s}') + S_p = 0 \quad (\text{L.14})$$

where

$$S_p = \Delta\eta(\bar{V}_{\xi_e}^* - \bar{V}_{\xi_w}^*) + \frac{\Delta\xi}{r_p}(r_n\bar{V}_{\eta_n}^* - r_s\bar{V}_{\eta_s}^*) \quad (\text{L.15})$$

is the imbalance of mass in the cell and approaches zero as the solution converges. Notice that this mass imbalance equation (the source term in the pressure solver) is dependent on knowing the contravariant velocity components at the faces of the cell. Obtaining these velocities through interpolations from the cell center can result in pressure oscillations for collocated grid arrangements. Employing a staggered grid arrangement eliminates this problem. To remedy the pressure oscillation problem on collocated grids, Rhie and Chow [80] suggest calculating the velocities at the cell faces by adding in some additional factors. This method has been widely adopted for collocated grids, and will be used here. The velocities at the cell faces are calculated from

$$\begin{aligned}
\bar{V}_{\xi_e}^* &= \bar{V}_{\xi_e}^* - \frac{r_{,\eta}^2}{A_e \rho} \Delta \xi \Delta \eta \left(\frac{p_E^* - p_P^*}{\delta \xi} - \bar{p}_{,\xi}^* \right) \\
\bar{V}_{\xi_w}^* &= \bar{V}_{\xi_w}^* - \frac{r_{,\eta}^2}{A_w \rho} \Delta \xi \Delta \eta \left(\frac{p_P^* - p_W^*}{\delta \xi} - \bar{p}_{,\xi}^* \right) \\
\bar{V}_{\eta_n}^* &= \bar{V}_{\eta_n}^* - \frac{z_{,\xi}^2}{A_n \rho} \Delta \xi \Delta \eta \left(\frac{p_N^* - p_P^*}{\delta \eta} - \bar{p}_{,\eta}^* \right) \\
\bar{V}_{\eta_s}^* &= \bar{V}_{\eta_s}^* - \frac{z_{,\xi}^2}{A_s \rho} \Delta \xi \Delta \eta \left(\frac{p_P^* - p_S^*}{\delta \eta} - \bar{p}_{,\eta}^* \right)
\end{aligned} \tag{L.16}$$

where the curved bar represents an interpolated value to the cell face.

Using Eq. (L.9) in Eq. (L.14) gives

$$\frac{r_{,\eta}^2 \Delta \eta^2}{A_e \rho} (\hat{p}_P' - \hat{p}_E') - \frac{r_{,\eta}^2 \Delta \eta^2}{A_w \rho} (\hat{p}_W' - \hat{p}_P') + \frac{z_{,\xi}^2 \Delta \xi^2 r_n}{r_P A_n \rho} (\hat{p}_P' - \hat{p}_N') - \frac{z_{,\xi}^2 \Delta \xi^2 r_s}{r_P A_s \rho} (\hat{p}_S' - \hat{p}_P') + S_p = 0 \tag{L.17}$$

Collecting terms and rearranging gives

$$D_P \hat{p}_P' = D_E \hat{p}_E' + D_W \hat{p}_W' + D_N \hat{p}_N' + D_S \hat{p}_S' - S_p \tag{L.18}$$

where

$$\begin{aligned}
D_E &\equiv \frac{r_{,\eta}^2 \Delta \eta^2}{A_e \rho} \\
D_W &\equiv \frac{r_{,\eta}^2 \Delta \eta^2}{A_w \rho} \\
D_N &\equiv \frac{z_{,\xi}^2 \Delta \xi^2 r_n}{r_P A_n \rho} \\
D_S &\equiv \frac{z_{,\xi}^2 \Delta \xi^2 r_s}{r_P A_s \rho} \\
D_P &\equiv D_E + D_W + D_N + D_S
\end{aligned} \tag{L.19}$$

Therefore,

$$\hat{p}_P' = \frac{1}{D_P} (D_E \hat{p}_E' + D_W \hat{p}_W' + D_N \hat{p}_N' + D_S \hat{p}_S' - S_p) \tag{L.20}$$

Note that Eq. (L.20) is linear and therefore requires no relaxation factor. Once the pressure correction terms are converged, they are added to the previous pressure terms using an under-relaxation factor, α_p

$$\hat{p}_P = \hat{p}_P^* + \alpha_p \hat{p}_P' \tag{L.21}$$

Using Eq. (L.9), the contravariant velocity terms are then corrected without relaxation (the under-relaxation for the velocity terms takes place in the momentum solver).

$$\begin{aligned}
 \bar{V}_{\xi_e} &= \bar{V}_{\xi_e}^* + \frac{r_{,\eta}^2 \Delta \eta}{A_e \rho} (\hat{p}'_P - \hat{p}'_E) \\
 \bar{V}_{\xi_w} &= \bar{V}_{\xi_w}^* + \frac{r_{,\eta}^2 \Delta \eta}{A_w \rho} (\hat{p}'_W - \hat{p}'_P) \\
 \bar{V}_{\eta_n} &= \bar{V}_{\eta_n}^* + \frac{z_{,\xi}^2 \Delta \xi}{A_n \rho} (\hat{p}'_P - \hat{p}'_N) \\
 \bar{V}_{\eta_s} &= \bar{V}_{\eta_s}^* + \frac{z_{,\xi}^2 \Delta \xi}{A_s \rho} (\hat{p}'_S - \hat{p}'_P)
 \end{aligned} \tag{L.22}$$

These velocities are now mass-conserving velocities and can be averaged to find the contravariant velocity components at the center of the cell. The contravariant velocity components are then used to calculate the corrected velocities in rectilinear coordinates using Eq. (G.12).

II. Boundary Treatment

Several different types of boundary conditions exist for pressure, however, they all require similar treatment. If the velocity at the boundary is fixed, then the velocity need not be updated, and the corresponding last term on the right-hand side of Eq. (L.22) is zero. Likewise, if the pressure gradient normal to the boundary is zero, then the corresponding last term on the right-hand side of Eq. (L.22) is zero. For example, if the east boundary is an outlet, the pressure gradient is zero and Eq. (L.20) can be solved by setting $D_E = 0$. The same would be true if the east boundary were a velocity inlet (velocity specified), a wall (velocity = 0, and the normal pressure gradient is zero), or a line of symmetry (normal pressure gradient is zero).

APPENDIX M

VORTICITY TRANSPORT

I. Laminar Flow Algorithm

The steady-state, incompressible continuity and Navier-Stokes equations can be written in vector format as

$$\nabla \cdot \mathbf{V} = 0 \quad (\text{M.1})$$

$$\frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla) \mathbf{V} = -\nabla \hat{p} / \rho + \nabla \cdot [2\nu \bar{\bar{\mathbf{S}}}(\mathbf{V})] \quad (\text{M.2})$$

Vorticity is defined as the curl of the velocity vector

$$\boldsymbol{\Omega} \equiv \nabla \times \mathbf{V} \quad (\text{M.3})$$

The vorticity transport equation for incompressible flow can be developed by taking the curl of the Navier-Stokes equations. After considerable algebra and the application of vector multiplication identities, the vorticity transport equation can be written as

$$\frac{\partial \boldsymbol{\Omega}}{\partial t} + (\mathbf{V} \cdot \nabla) \boldsymbol{\Omega} = (\boldsymbol{\Omega} \cdot \nabla) \mathbf{V} + \nu \nabla^2 \boldsymbol{\Omega} + \nabla \nu \times (\nabla^2 \mathbf{V}) + 2\nabla \times [\nabla \nu \cdot \bar{\bar{\mathbf{S}}}(\mathbf{V})] \quad (\text{M.4})$$

For two-dimensional, steady-state flow in Cartesian coordinates,

$$\boldsymbol{\Omega} = \begin{Bmatrix} 0 \\ 0 \\ \Omega \end{Bmatrix}, \quad \nabla = \begin{Bmatrix} \partial x \\ \partial y \\ 0 \end{Bmatrix}, \quad \frac{\partial}{\partial t} = 0 \quad (\text{M.5})$$

Assuming that the fluid properties are constant throughout the flow (ie. laminar flow, $\nabla \nu = 0$), and using Eq. (M.5) in Eq. (M.4) gives the scalar two-dimensional vorticity transport equation for steady-state flow

$$(\mathbf{V} \cdot \nabla) \Omega = \nu \nabla^2 \Omega \quad (\text{M.6})$$

This can be written in Cartesian coordinates as

$$V_x \frac{\partial \Omega}{\partial x} + V_y \frac{\partial \Omega}{\partial y} = \nu \left(\frac{\partial^2 \Omega}{\partial x^2} + \frac{\partial^2 \Omega}{\partial y^2} \right) \quad (\text{M.7})$$

By definition,

$$\nabla \times \boldsymbol{\Omega} = \nabla \times (\nabla \times \mathbf{V}) = \nabla(\nabla \cdot \mathbf{V}) - \nabla^2 \mathbf{V} \quad (\text{M.8})$$

Applying the continuity equation for incompressible flow to Eq. (M.8) gives

$$\nabla \times \boldsymbol{\Omega} = -\nabla^2 \mathbf{V} \quad (\text{M.9})$$

This vector equation can be written as two scalar equations in Cartesian coordinates as

$$-\frac{\partial \Omega}{\partial y} = \frac{\partial^2 V_x}{\partial x^2} + \frac{\partial^2 V_x}{\partial y^2} \quad (\text{M.10})$$

$$\frac{\partial \Omega}{\partial x} = \frac{\partial^2 V_y}{\partial x^2} + \frac{\partial^2 V_y}{\partial y^2} \quad (\text{M.11})$$

Equations (M.7), (M.10), and (M.11) can be discretized and rearranged to yield three equations that can be used to obtain better estimates for the three unknown values, Ω , V_x , and V_y . For example, given an estimate for the three variables, Eq. (M.7) can be used to calculate an improved estimate for Ω . Likewise, Eq. (M.10) can be used to calculate an improved estimate for V_x , and Eq. (M.11) can be used to calculate an improved estimate for V_y .

II. Turbulent Flow Algorithm

A. The Ensemble Average Vorticity Transport Equation

Taking the ensemble average of Eq. (M.4) gives

$$\frac{\partial \overline{\boldsymbol{\Omega}}}{\partial t} + (\overline{\mathbf{V}} \cdot \nabla) \overline{\boldsymbol{\Omega}} + \overline{(\tilde{\mathbf{V}} \cdot \nabla) \tilde{\boldsymbol{\Omega}}} = (\overline{\boldsymbol{\Omega}} \cdot \nabla) \overline{\mathbf{V}} + \overline{(\tilde{\boldsymbol{\Omega}} \cdot \nabla) \tilde{\mathbf{V}}} + \nu \nabla^2 \overline{\boldsymbol{\Omega}} + \nabla \nu \times (\nabla^2 \overline{\mathbf{V}}) + 2 \nabla \times [\nabla \nu \cdot \overline{\tilde{\mathbf{S}}}(\overline{\mathbf{V}})] \quad (\text{M.12})$$

Applying Eq. (M.5) gives the steady-state two-dimensional equation

$$(\overline{\mathbf{V}} \cdot \nabla) \overline{\boldsymbol{\Omega}} + \overline{(\tilde{\mathbf{V}} \cdot \nabla) \tilde{\boldsymbol{\Omega}}} = \nu \nabla^2 \overline{\boldsymbol{\Omega}} + \nabla \nu \times (\nabla^2 \overline{\mathbf{V}}) + 2 \nabla \times [\nabla \nu \cdot \overline{\tilde{\mathbf{S}}}(\overline{\mathbf{V}})] \quad (\text{M.13})$$

In order to use this equation in an algorithm to predict turbulent flow, the quantity $\overline{(\tilde{\mathbf{V}} \cdot \nabla) \tilde{\boldsymbol{\Omega}}}$ must somehow be modeled. An alternate approach is to develop the vorticity transport equation from the RANS equations directly as follows.

B. The RANS Ensemble Average Vorticity Transport Equation

The steady-state, incompressible continuity and Boussinesq-based RANS equations can be written in vector format as

$$\nabla \cdot \bar{\mathbf{V}} = 0 \quad (\text{M.14})$$

$$\frac{\partial \bar{\mathbf{V}}}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) \bar{\mathbf{V}} = -\nabla \hat{p} / \rho + \nabla \cdot [2(\nu + \nu_t) \bar{\mathbf{S}}(\bar{\mathbf{V}})] \quad (\text{M.15})$$

The ensemble average of the vorticity is defined as the curl of the ensemble average of the velocity vector

$$\bar{\boldsymbol{\Omega}} \equiv \nabla \times \bar{\mathbf{V}} \quad (\text{M.16})$$

The ensemble average vorticity transport equation for incompressible flow can be developed by taking the curl of the Boussinesq-based RANS equations. Following the development of Phillips, after considerable algebra and the application of vector multiplication identities, the ensemble average vorticity transport equation can be written as

$$\frac{\partial \bar{\boldsymbol{\Omega}}}{\partial t} + (\bar{\mathbf{V}} \cdot \nabla) \bar{\boldsymbol{\Omega}} = (\bar{\boldsymbol{\Omega}} \cdot \nabla) \bar{\mathbf{V}} + (\nu + \nu_t) \nabla^2 \bar{\boldsymbol{\Omega}} + \nabla(\nu + \nu_t) \times (\nabla^2 \bar{\mathbf{V}}) + 2\nabla \times [\nabla(\nu + \nu_t) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}})] \quad (\text{M.17})$$

For two-dimensional, steady-state flow in Cartesian coordinates,

$$\bar{\boldsymbol{\Omega}} = \begin{Bmatrix} 0 \\ 0 \\ \bar{\Omega} \end{Bmatrix}, \quad \nabla = \begin{Bmatrix} \partial x \\ \partial y \\ 0 \end{Bmatrix}, \quad \frac{\partial}{\partial t} = 0 \quad (\text{M.18})$$

Using Eq. (M.18) in Eq. (M.17) gives the scalar two-dimensional vorticity transport equation for steady-state flow

$$(\bar{\mathbf{V}} \cdot \nabla) \bar{\boldsymbol{\Omega}} = (\nu + \nu_t) \nabla^2 \bar{\boldsymbol{\Omega}} + \nabla(\nu + \nu_t) \times (\nabla^2 \bar{\mathbf{V}}) + 2\nabla \times [\nabla(\nu + \nu_t) \cdot \bar{\mathbf{S}}(\bar{\mathbf{V}})] \quad (\text{M.19})$$

This can be written in Cartesian coordinates as

$$\begin{aligned}
\bar{V}_x \frac{\partial \bar{\Omega}}{\partial x} + \bar{V}_y \frac{\partial \bar{\Omega}}{\partial y} &= (\nu + \nu_t) \left(\frac{\partial^2 \bar{\Omega}}{\partial x^2} + \frac{\partial^2 \bar{\Omega}}{\partial y^2} \right) + \frac{\partial(\nu + \nu_t)}{\partial x} \left(\frac{\partial^2 \bar{V}_y}{\partial x^2} + \frac{\partial^2 \bar{V}_y}{\partial y^2} \right) - \frac{\partial(\nu + \nu_t)}{\partial y} \left(\frac{\partial^2 \bar{V}_x}{\partial x^2} + \frac{\partial^2 \bar{V}_x}{\partial y^2} \right) \\
&+ \frac{\partial}{\partial x} \left[\frac{\partial(\nu + \nu_t)}{\partial x} \left(\frac{\partial \bar{V}_x}{\partial y} + \frac{\partial \bar{V}_y}{\partial x} \right) + 2 \frac{\partial(\nu + \nu_t)}{\partial y} \frac{\partial \bar{V}_y}{\partial y} \right] \\
&- \frac{\partial}{\partial y} \left[2 \frac{\partial(\nu + \nu_t)}{\partial x} \frac{\partial \bar{V}_x}{\partial x} + \frac{\partial(\nu + \nu_t)}{\partial y} \left(\frac{\partial \bar{V}_y}{\partial x} + \frac{\partial \bar{V}_x}{\partial y} \right) \right]
\end{aligned} \tag{M.20}$$

By definition,

$$\nabla \times \bar{\Omega} = \nabla \times (\nabla \times \bar{\mathbf{V}}) = \nabla(\nabla \cdot \bar{\mathbf{V}}) - \nabla^2 \bar{\mathbf{V}} \tag{M.21}$$

Applying the continuity equation for incompressible flow to Eq. (M.21) gives

$$\nabla \times \bar{\Omega} = -\nabla^2 \bar{\mathbf{V}} \tag{M.22}$$

This vector equation can be written as two scalar equations in Cartesian coordinates as

$$-\frac{\partial \bar{\Omega}}{\partial y} = \frac{\partial^2 \bar{V}_x}{\partial x^2} + \frac{\partial^2 \bar{V}_x}{\partial y^2} \tag{M.23}$$

$$\frac{\partial \bar{\Omega}}{\partial x} = \frac{\partial^2 \bar{V}_y}{\partial x^2} + \frac{\partial^2 \bar{V}_y}{\partial y^2} \tag{M.24}$$

Equations (M.20), (M.23), and (M.24) can be discretized and rearranged to yield three equations that can be used to obtain better estimates for the three unknown values, $\bar{\Omega}$, \bar{V}_x , and \bar{V}_y . For example, given an estimate for the three variables, Eq. (M.20) can be used to calculate an improved estimate for $\bar{\Omega}$. Likewise, Eq. (M.23) can be used to calculate an improved estimate for \bar{V}_x , and Eq. (M.24) can be used to calculate an improved estimate for \bar{V}_y .

APPENDIX N

ONE-DIMENSIONAL DIRECT INTEGRATION METHOD

I. The k - ε Model Equations

Combining Eq. (C.31) with Eqs. (B.11) and (B.12) gives one first-order equation and two second-order equations. This system of equations can be rewritten as a system of five first-order equations by introducing the change of variables

$$\begin{aligned} q^+ &\equiv -(1+\nu^+/\sigma_k) \frac{dk^+}{dy^+} \\ \theta^+ &\equiv -(1+\nu^+/\sigma_\varepsilon) \frac{d\varepsilon^+}{dy^+} \end{aligned} \quad (\text{N.1})$$

Directly substituting the definition for ν^+ , the five first-order equations can be written as

$$\begin{aligned} \frac{du^+}{dy^+} &= \frac{\varepsilon^+(R_\tau - y^+)}{R_\tau(\varepsilon^+ + C_\mu f_\mu k^{+2})} \\ \frac{dq^+}{dy^+} &= \frac{C_\mu f_\mu k^{+2} \varepsilon^+(R_\tau - y^+)^2}{R_\tau^2(\varepsilon^+ + C_\mu f_\mu k^{+2})^2} - \varepsilon^+ - \varepsilon_o^+ \\ \frac{dk^+}{dy^+} &= -\frac{\sigma_k \varepsilon^+ q^+}{\sigma_k \varepsilon^+ + C_\mu f_\mu k^{+2}} \\ \frac{d\theta^+}{dy^+} &= C_{\varepsilon 1} f_1 \frac{C_\mu f_\mu k^+ \varepsilon^{+2} (R_\tau - y^+)^2}{R_\tau^2(\varepsilon^+ + C_\mu f_\mu k^{+2})^2} - C_{\varepsilon 2} f_2 \frac{\varepsilon^{+2}}{k^+} + E^+ \\ \frac{d\varepsilon^+}{dy^+} &= -\frac{\sigma_\varepsilon \varepsilon^+ \theta^+}{\sigma_\varepsilon \varepsilon^+ + C_\mu f_\mu k^{+2}} \end{aligned} \quad (\text{N.2})$$

and the boundary conditions can be written as

$$u^+(0) = 0, \quad q^+(0) = 0, \quad k^+(0) = 0, \quad q^+(R_\tau) = 0, \quad \theta^+(R_\tau) = 0 \quad (\text{N.3})$$

Once the damping functions f_μ , f_1 , f_2 , E^+ , and ε_o^+ have been specified, the system of equations given by Eq. (N.2) along with the boundary conditions in Eq. (N.3) could likely be directly integrated using a numerical integration method such as the fourth-order Runge-Kutta algorithm. Various damping functions have been proposed and one is included in the following section.

The model proposed by Lam and Bremhorst [68] can be written in nondimensional form as

$$\begin{aligned} f_\mu &= [1 - \exp(-0.0165R_y)]^2 [1 + 20.5/R_t], & f_1 &= 1 + (0.05/f_\mu)^3, & f_2 &= 1 - \exp(-R_t^2), \\ R_t &\equiv k^{+2}/\varepsilon^+, & R_y &\equiv y^+ \sqrt{k^+}, & E^+ &= \varepsilon_o^+ = 0 \end{aligned} \quad (\text{N.4})$$

and the closure coefficients are

$$C_\mu = 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3, \quad (\text{N.5})$$

Lam and Bremhorst propose the boundary condition

$$\varepsilon^+(0) = \frac{d^2 k^+}{dy^{+2}}(0) \quad (\text{N.6})$$

However, this is not a boundary condition and can be derived directly by examining the transport equations as they approach the wall. This relation is satisfied by virtue of the equations, and cannot be used as a boundary condition. Equation (N.6) can be used as a near-wall approximation for ε to replace the governing equation for ε near the wall. However, it is not a boundary condition and should not be referred to as a boundary condition. Others have used the boundary condition

$$\frac{d\varepsilon^+}{dy^+}(0) = 0 \quad (\text{N.7})$$

This is also incorrect. It has no physical basis and is employed most often because it is easy to implement.

The Lam-Bremhorst model is indeterminate at the wall. Near the wall, approximations for the damping functions can be written as

$$f_\mu = \frac{20.5(0.0165)^2 \varepsilon^+ y^{+2}}{k^+}, \quad f_1 = 1 + \frac{(0.05)^3 k^{+3}}{(20.5)^3 (0.0165)^6 \varepsilon^{+3} y^{+6}}, \quad f_2 = \frac{k^{+4}}{\varepsilon^{+2}} \quad (\text{N.8})$$

Note that f_μ and f_1 are still indeterminate at the wall where both y^+ and k^+ approach zero. The Taylor Series expansion of k^+ near the wall can be written as

$$k^+(y^+) = k^+(0) + \frac{dk^+}{dy^+}(0)y^+ + \frac{1}{2} \frac{d^2 k^+}{dy^{+2}}(0)y^{+2} + \dots \quad (\text{N.9})$$

The first two terms in this series are zero from the boundary condition definitions. The third term can be written in terms of ε because the turbulent eddy viscosity approaches zero near a wall. Near a wall, Eq. (B.12) can be combined with the Taylor Series expansion of k^+ to yield

$$k^+(y^+) \cong \frac{1}{2} \frac{d^2 k^+}{dy^{+2}}(0) y^{+2} \cong \frac{y^{+2}}{2} \varepsilon^+(0) \quad (\text{N.10})$$

Using this in Eq. (N.8) gives the final near-wall approximations for the damping functions

$$f_\mu = 0.01116225, \quad f_1 = 1 + \frac{(0.05)^3}{(0.01116225)^3}, \quad f_2 = \frac{k^{+4}}{\varepsilon^{+2}} \quad (\text{N.11})$$

Note that f_μ and f_1 are simply constants and can be used in the five first-order system of equations. Substituting the near-wall approximation for f_2 into the fourth equation in Eq. (N.2) gives the near-wall approximation for that equation

$$\frac{d\theta^+}{dy^+} = C_{\varepsilon 1} f_1 \frac{C_\mu f_\mu k^+ \varepsilon^{+2} (R_\tau - y^+)^2}{R_\tau^2 (\varepsilon^+ + C_\mu f_\mu k^{+2})^2} - C_{\varepsilon 2} k^{+3} \quad (\text{N.12})$$

II. The k - ω Model Equations

Combining Eq. (C.31) with Eqs. (B.65) and (B.66) gives one first-order equation and two second-order equations. This system of equations can be rewritten as a system of five first-order equations by introducing the change of variables

$$\begin{aligned} q^+ &\equiv -(1 + \nu^+ / \sigma_k) \frac{dk^+}{dy^+} \\ \psi^+ &\equiv -(1 + \nu^+ / \sigma_\omega) \frac{d\omega^+}{dy^+} \end{aligned} \quad (\text{N.13})$$

The five first-order equations can be written as

$$\begin{aligned}
\frac{du^+}{dy^+} &= \frac{1-y^+/R_\tau}{1+\nu^+} \equiv u^+ \\
\frac{dq^+}{dy^+} &= \nu^+(u^+)^2 - C_\mu f_k k^+ \omega^+ \\
\frac{dk^+}{dy^+} &= -\frac{\sigma_k q^+}{\sigma_k + \nu^+} \equiv k^+ \\
\frac{\partial \psi^+}{\partial y^+} &= C_{\omega 1} f_1 f_\mu (u^+)^2 - C_{\omega 2} f_2 \omega^{+2} \\
\frac{d\omega^+}{dy^+} &= -\frac{\sigma_\omega \psi^+}{\sigma_\omega + \nu^+} \equiv \omega^+
\end{aligned} \tag{N.14}$$

Or, by substituting the definition for ν^+ directly,

$$\begin{aligned}
\frac{du^+}{dy^+} &= \frac{\omega^+(1-y^+/R_\tau)}{\omega^+ + f_\mu k^+} \equiv u^+ \\
\frac{dq^+}{dy^+} &= f_\mu k^+ \omega^+ \frac{(1-y^+/R_\tau)^2}{(\omega^+ + f_\mu k^+)^2} - C_\mu f_k k^+ \omega^+ \\
\frac{dk^+}{dy^+} &= -\frac{\omega^+ \sigma_k q^+}{\omega^+ \sigma_k + f_\mu k^+} \equiv k^+ \\
\frac{\partial \psi^+}{\partial y^+} &= C_{\omega 1} f_1 f_\mu (u^+)^2 - C_{\omega 2} f_2 \omega^{+2} \\
\frac{d\omega^+}{dy^+} &= -\frac{\omega^+ \sigma_\omega \psi^+}{\omega^+ \sigma_\omega + f_\mu k^+} \equiv \omega^+
\end{aligned} \tag{N.15}$$

The latter formulation is better suited, because ω approaches zero at the channel centerline. The boundary conditions are

$$u^+(0) = 0, \quad q^+(0) = 0, \quad k^+(0) = 0, \quad q^+(R_\tau) = 0, \quad \psi^+(R_\tau) = 0 \tag{N.16}$$

At the wall, ω^+ and $d\omega^+/dy^+$ are singular. Therefore, near-wall approximations for the model must be used until some specified distance away from the wall. The leading-order terms of the dependent variables are

$$\begin{aligned}
u^+(y^+) &= y^+ - \frac{y^{+2}}{2R_t} \\
q^+(y^+) &= -\frac{dk^+}{dy^+} = -k^{+''}(0)y^+ \\
k^+(y^+) &= \frac{k^{+''}(0)}{2}y^{+2} \\
\psi^+(y^+) &= -\frac{d\omega^+}{dy^+} = \frac{12}{C_{\omega 2}f_2(0)y^{+3}} \\
\omega^+(y^+) &= \frac{6}{C_{\omega 2}f_2(0)y^{+2}}
\end{aligned} \tag{N.17}$$

Once the damping functions f_μ , f_1 , f_2 , and f_k have been specified, the system of equations given by Eq. (N.17) along with the boundary conditions in Eq. (N.16) could likely be directly integrated using a numerical integration method such as the fourth-order Runge-Kutta algorithm. Various damping functions have been proposed and one is included in the following section.

The Wilcox 1998 model [45] can be written in nondimensional form as

$$\begin{aligned}
f_\mu &= \frac{0.024 + R_t/6}{1 + R_t/6}, \quad f_1 = \frac{1/9 + R_t/2.95}{f_\mu(1 + R_t/2.95)}, \quad f_2 = 1 \\
f_k &= \frac{4/15 + (R_t/8)^4}{1 + (R_t/8)^4} \left\{ \begin{array}{l} 1, \quad \psi_k \leq 0 \\ \frac{1 + 680\psi_k^2}{1 + 400\psi_k^2}, \quad \psi_k > 0 \end{array} \right\}, \\
\psi_k &= \frac{1}{\omega^{+3}} \frac{dk^+}{dy^+} \frac{d\omega^+}{dy^+}, \quad R_t = \frac{k^+}{\omega^+}
\end{aligned} \tag{N.18}$$

and the closure constants are

$$C_\mu = 0.09, \quad C_{\omega 1} = 0.52, \quad C_{\omega 2} = 0.072, \quad \sigma_k = 2.0, \quad \sigma_\omega = 2.0 \tag{N.19}$$

Near a wall, the damping functions approach

$$f_\mu = 0.024, \quad f_1 = 0.216, \quad f_2 = 1, \quad f_k = 4/15 \tag{N.20}$$

At the wall, ω^+ and $d\omega^+/dy^+$ are singular. Wilcox [86] suggests using the leading-order solution

$$\omega^+ \cong \frac{6}{C_{\omega 2}f_2}y^{+2} = \frac{6}{0.072}y^{+2} \tag{N.21}$$

for the first 7 to 10 grid points off the surface to avoid numerical errors near a wall boundary. Wilcox states that the approximation given in Eq. (N.21) is only good for grid points where $y^+ < 2.5$. Therefore, it is recommended that the grid be fine enough to ensure that at least 7 grid points are within this constraint.

APPENDIX O

CODE FOR SOLVING THE PHILLIPS k - λ MODEL

The following code can be used to solve the Phillips k - λ model for fully rough pipe flow. The code can also be used to give solutions for the eddy-viscosity models included as Eqs. (7.35), (7.36), (7.42), and (7.43).

```
!gfortran -fdefault-real-8 solver.f90 main_solve.f90
module solver
  IMPLICIT NONE
  integer :: m,nplot
  integer :: model
  integer :: modelcase

  real :: Rtau,kr,ksp,beta,nu_hat,ximax,nutmax
  real :: kappa, gamma,C10
  real :: sigmak,Clambda
  real :: kc,kw !kcenter, kwall
  real :: Re_bulk,Re_max,Re_core,R_c,Cf4
  real :: yp1
  real :: AA,BB
  real :: Cr1,Cr2,Cr3,ar1,ar2,ar3,ar4
  real :: A0,A1,a,rc,B0

  CHARACTER*(80):: rec,init,file_r
  CHARACTER(LEN=100)::fn

  real, allocatable, dimension(:)::r_hat,yplus
  real, allocatable, dimension(:)::k,kprime
  real, allocatable, dimension(:)::q,qprime
  real, allocatable, dimension(:)::u,uprime
  real, allocatable, dimension(:)::nut_hat
  real, allocatable, dimension(:)::omega,lambda

  real, allocatable, dimension(:)::vRe,vCF,vksp,vNi,vRebulk,vkr

contains

  subroutine solver_allocate()
    !Allocate Memory
    ALLOCATE(r_hat(m)); ALLOCATE(yplus(m))
    ALLOCATE(k(m)); ALLOCATE(kprime(m));
    ALLOCATE(q(m)); ALLOCATE(qprime(m));
    ALLOCATE(u(m)); ALLOCATE(uprime(m));
    ALLOCATE(nut_hat(m));
    ALLOCATE(omega(m)); ALLOCATE(lambda(m));
    ALLOCATE(vRe(nplot)); ALLOCATE(vCF(nplot));
    ALLOCATE(vksp(nplot)); ALLOCATE(vNi(nplot));
    ALLOCATE(vkr(nplot)); ALLOCATE(vRebulk(nplot))
  end subroutine solver_allocate

  subroutine solver_deallocate()
    !Deallocate Memory
    DEALLOCATE(r_hat); DEALLOCATE(yplus);
    DEALLOCATE(k); DEALLOCATE(kprime);
    DEALLOCATE(q); DEALLOCATE(qprime);
    DEALLOCATE(u); DEALLOCATE(uprime);
    DEALLOCATE(nut_hat);
    DEALLOCATE(omega); DEALLOCATE(lambda);
    DEALLOCATE(vRe); DEALLOCATE(vCF);
    DEALLOCATE(vksp); DEALLOCATE(vNi);
    DEALLOCATE(vkr); DEALLOCATE(vRebulk)
  end subroutine solver_deallocate

  subroutine create_grid()
```

```

integer :: j
real :: eta,cbeta,dzeta

dzeta = 1.0/real(m-1)
!Create Grid
do j=1,m,1
  eta = real(m-j)/real(m-1)
  cbeta = ((beta+1.0)/(beta-1.0))**(1.0-eta)
  if(beta .eq. 0.0) then
    r_hat(j) = 1.0 - real(j-1)*dzeta
  else
    r_hat(j) = 1.0 - (beta+1.0 - (beta-1.0)*cbeta)/(1.0 + cbeta)
  end if
end do
r_hat(1) = 0.0
end subroutine create_grid

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! Phillips Model !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
subroutine Phillips_kl(iprint,niter)
  real :: x1,x2,f1,f2,slope,xnew,Re_old,relaxation
  integer :: iter,niter,iprint,oiter

!   write(*,*) '----- Newton Solver -----'
  if(iprint.eq.1) then
    write(*,*) '   iter          q_wall          q_center
k_center          & Re_core          4CF'
    write(*,*) '----- & ----- & ----- & -----'
    write(*,*) '----- & ----- & ----- & -----'
  end if
  relaxation = 1.0
  niter = 100
  oiter = 0
  x2 = 0.0
do while(((niter.eq.100).or.(x2.ne.x2)).and.(oiter<5)) !under relax if solution not
found
  Rtau = 0.5*ksp/kr
  nu_hat = 1.0/Rtau
  Re_core = 21.26*Rtau**(1.078)

  nut_hat(1) = Cl0 !initialize so that near-centerline approx for k works
  k(1) = 0.205*Rtau**0.158

  kw = ((sqrt(1.0/ksp**2 + 4.0*(kappa*gamma)**2)-1.0/ksp)/(2.0*Cr2*ksp**ar2))**2

  x1 = 9.4*kr**(-0.152)
  x2 = x1*(1.0+0.000001)
  call p_integrate_k_fromwall(x1,f1)
  call p_integrate_k_fromwall(x1,f1) !Call twice so that f1 is based off of
practical Re_core
  Re_old = 0.0
  if(iprint.eq.1) write(*,*) "relax = ",relaxation
  iter = 1
  do while(((abs(f1).gt.1.0e-9).or.(abs(Re_core-Re_old)/Re_core.gt.1.0e-
13)).and.(iter.lt.100).and.(x2.eq.x2)) ! secant method for finding root
    Re_old = Re_core
    call p_integrate_k_fromwall(x2,f2)
    slope = (f2-f1)/(x2-x1)
    xnew = x2 - relaxation*f2/slope
    if(abs((x1-x2)/x1).lt.1.0e-15) xnew = x2
    x1 = x2; x2 = xnew; f1 = f2
    if(iprint.eq.1) write(*,*) iter,xnew,f2,k(1),Re_core,Cf4
    iter = iter + 1
  end do
  niter = iter
  relaxation = relaxation - 0.05
  oiter = oiter + 1
end do
!   call p_log_check()
end subroutine Phillips_kl

subroutine p_vksp(iprint,ngood,error)
  integer :: iprint,ngood,niter
  real :: error,power

  power = 3.0
  ksp = 10**power
  Re_bulk = 0.0

```

```

ngood = 1
error = 0.0
if(iprint.eq.1) write(*,*) '      ngood   kr                ksp
q_wall          q_center   &                Re_bulk          4CF
Colebrook       error      &                &                num_iter'
do while(((Re_bulk<1.0e8).and.(ngood<=nplot)).or.(ngood<5))
  call Phillips_kl(0,niter)
  vRebulk(ngood) = Re_bulk
  vCF(ngood) = Cf4
  if((Re_bulk<1.0e8).or.(ngood<4)) error = error + (abs(func_Colebrook(Re_bulk) -
vCF(ngood))/func_Colebrook(Re_bulk))**2 !new corrected
!   if((Re_bulk<1.0e8).or.(ngood<4)) error = error + (abs(func_Colebrook(Re_bulk) -
vCF(ngood))/vCF(ngood))**2 !old
  if(abs(q(1)).gt.1.0e-8) error = sqrt(-error) !puts a NAN in the output so that
the optimization will halt
  if(iprint.eq.1) write(*,*)
ngood,kr,ksp,q(m),q(1),Re_bulk,Cf4,func_Colebrook(Re_bulk),error,niter
  ngood = ngood + 1
  power = power + 0.25
  ksp = 10.0**power
end do
ngood = ngood - 2
end subroutine p_vksp

subroutine p_log_check()
  real ::
ypmin,logypmin,logrtau,dlog,yprealmin,dxi,xi,yp,dyp,ans0(4),ans1(4),du(4,4),dummy(2),ump,
nut
  integer :: i,lower
  ypmin = 0.1
  lower = (m-1)/100
  logypmin = log10(ypmin)
  logrtau = log10(Rtau)
  dlog = (logrtau-logypmin)/real(m-1-lower)
  yprealmin = logypmin-(lower-1)*dlog
  ans0(1) = k(m)
  ans0(2) = q(m)
  ans0(3) = 0.0 !wall value
  ans0(4) = 0.0 !wall value
  yp = 0.0
  xi = 1.0
  do i=m,2,-1
    dyp = 10.0**(yprealmin+real(m-i)*dlog) - yp
    dxi = 1.0-(yp+dyp)/Rtau - xi
    call rnkta4(func_rnkta_p,4,dummy,xi,ans0,dxi,du,ans1)
    ans0(:) = ans1(:)
    yp = 10.0**(yprealmin+real(m-i)*dlog)
    xi = 1.0-yp/Rtau
  end do
  ump = -ans1(4)
  Cf4 = 8.0/(ump**2)
  Re_bulk = 2.0*ump/nu_hat
  Re_max = 2.0*ans1(3)/nu_hat
  nut = func_p_nut(0.0,ans1(1))
  R_c = sqrt(2.0*(nu_hat+nut)*ans1(3))
  Re_core = R_c*Re_max
  write(*,*) 'Log Grid Check:'
  write(*,*) 1,q(m),ans1(2),ans1(1),Re_core,Cf4
end subroutine p_log_check

subroutine p_update()
  integer :: i
  real :: xi,nut
  do i=1,m
    xi = r_hat(i)
    lambda(i) = func_p_lambda(xi)
    nut = func_p_nut(xi,k(i))
    nut_hat(i) = nut
    yplus(i) = Rtau*(1.0-r_hat(i))
    uprime(i) = func_p_uprime(xi,k(i))
    kprime(i) = func_p_kprime(xi,k(i),q(i))
    qprime(i) = func_p_qprime(xi,k(i))
    omega(i) = sqrt(Clambda*k(i))/lambda(i)
  end do
  kprime(1) = 0.0
end subroutine p_update

```

```

subroutine p_integrate_k_fromwall(qwall,qcenter)
  real :: qwall, qcenter,dxi,xi,ans0(4),ans1(4),du(4,4),dummy(2),ump
  integer :: i

  k(m) = kw !k wall value
  q(m) = qwall
  u(m) = 0.0
  ans0(1) = k(m)
  ans0(2) = q(m)
  ans0(3) = 0.0 !wall value
  ans0(4) = 0.0 !wall value
  xi = 0.0
  do i=m,2,-1
    xi = r_hat(i)
    dxi = r_hat(i-1)-r_hat(i)
    call rnkta4(func_rnkta_p,4,dummy,xi,ans0,dxi,du,ans1)
    ans0(:) = ans1(:)
    k(i-1) = ans1(1)
    q(i-1) = ans1(2)
    u(i-1) = ans1(3)
  end do
  qcenter = q(1)
  call p_update()
  ump = -ans1(4)
  Cf4 = 8.0/(ump**2)
  Re_bulk = 2.0*ump/nu_hat
  Re_max = 2.0*u(1)/nu_hat
  R_c = sqrt(2.0*(nu_hat+nut_hat(1))*u(1))
  Re_core = R_c*Re_max
end subroutine p_integrate_k_fromwall

real function func_p_nut(xi,k_local)
  real :: xi,k_local
  func_p_nut = func_p_lambda(xi)*sqrt(k_local)
  if(k_local.lt.0.0) func_p_nut = 0.0
return; end function func_p_nut

real function func_p_lambda(xi)
  real :: xi
  func_p_lambda = min(Cr1*Re_core**ar1, 2.0*kr*Cr2*ksp**ar2 + Cr3*Re_core**ar3*(1.0-xi)**ar4)
return; end function func_p_lambda

real function func_rnkta_p(n,i,dummy,xi,curr)
  integer :: n,i
  real :: xi,curr(n),dummy(2)
  if(i.eq.1) then !k eqn
    func_rnkta_p = func_p_kprime(xi,curr(1),curr(2))
!   if(xi.eq.0.0) func_rnkta_p = 0.0
  elseif(i.eq.2) then !q eqn
    func_rnkta_p = func_p_qprime(xi,curr(1))
  elseif(i.eq.3) then !u eqn
    func_rnkta_p = func_p_uprime(xi,curr(1))
  else !Cf eqn
    func_rnkta_p = 2.0*curr(3)*xi
  end if
return; end function func_rnkta_p

real function func_p_kprime(xi,k_local,q_local)
  real :: xi,k_local,q_local,nut
  nut = func_p_nut(xi,k_local)
  func_p_kprime = -q_local/xi/(nu_hat/3.0 + 5.0/3.0*nut/sigmak)
  if(xi.lt.0.002) then
    AA = 3.0*sigmak*Clambda*nu_hat/(4.0*(sigmak*nu_hat+5.0*nut)*nut**2)
    BB = 3.0*sigmak*nut/(8.0*(sigmak*nu_hat+5.0*nut)*(nu_hat+nut)**2)
    func_p_kprime = 2.0*AA*k(1)**2*xi -2.0*(BB-AA**2*k(1)**3)*xi**3
    if(nut.eq.0.0) func_p_kprime = 0.0
  end if
return; end function func_p_kprime

real function func_p_qprime(xi,k_local)
  real :: xi,k_local,nut,lam
  lam = func_p_lambda(xi)
  nut = func_p_nut(xi,k_local)
  func_p_qprime = nut*xi**3/(nu_hat+nut)**2 - xi*nu_hat*Clambda*k_local/lam**2
return; end function func_p_qprime

real function func_p_uprime(xi,k_local)
  real :: xi,k_local,nut

```

```

    nut = func_p_nut(xi,k_local)
    func_p_uprime = -xi/(nu_hat+nut)
return; end function func_p_uprime

!!!!!!!!!!!!!!!!!!!!!!!!!!!! Viscosity Models !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

subroutine Viscosity_Model()
  real :: dxi,xi,ans0(2),ans1(2),du(2,4),dummy(2),ump
  integer :: i

  Rtau = 0.5*ksp/kr
  nu_hat = 1.0/Rtau

  if(model.eq.4) then !Kays and Crawford must find root
    call KaysCrawford_root()
  end if

  u(m) = 0.0
  ans0(1) = 0.0 !wall value
  ans0(2) = 0.0 !wall value
  xi = 0.0
  do i=m,2,-1
    xi = r_hat(i)
    dxi = r_hat(i-1)-r_hat(i)
    call rnkta4(func_rnkta_v,2,dummy,xi,ans0,dxi,du,ans1)
    ans0(:) = ans1(:)
    u(i-1) = ans1(1)
  end do
  call v_update()
  ump = -ans1(2)
  Cf4 = 8.0/(ump**2)
  Re_bulk = 2.0*ump/nu_hat
  Re_max = 2.0*u(1)/nu_hat
  R_c = sqrt(2.0*(nu_hat+nut_hat(1))*u(1))
  Re_core = R_c*Re_max
end subroutine Viscosity_Model

subroutine v_update()
  integer :: i
  real :: xi,nut
  do i=1,m
    xi = r_hat(i)
    nut = func_v_nut(xi)
    nut_hat(i) = nut
    yplus(i) = Rtau*(1.0-r_hat(i))
    uprime(i) = -xi/(nu_hat+nut)
  end do
end subroutine v_update

real function func_rnkta_v(n,i,dummy,xi,curr)
  integer :: n,i
  real :: xi,curr(n),dummy(2),nut
  if(i.eq.1) then !u eqn
    nut = func_v_nut(xi)
    func_rnkta_v = -xi/(nu_hat+nut)
  else !Cf eqn
    func_rnkta_v = 2.0*curr(1)*xi
  end if
return; end function func_rnkta_v

real function func_v_nut(xi)
  real :: xi
  if(model.eq.2) then !Nikuradse Eq. 7.35
    func_v_nut = kappa*(C10+2.0*gamma*kr-(2.0*C10-0.5)*xi**2-(0.5-
C10)*xi**4)*xi**(0.5)
  elseif(model.eq.3) then !Reichardt Eq. 7.36
    func_v_nut = kappa/6.0*(1.0-xi+2.0*gamma*kr)*(1.0+xi)*(1.0+2.0*xi**2)
  elseif(model.eq.4) then !Kays and Crawford Eq. 7.42
    func_v_nut = func_KC_nut(xi)
  elseif(model.eq.5) then !Log-Law Model Eq. 7.43
    func_v_nut = kappa*xi*(1.0-xi+2.0*gamma*kr)
  elseif(model.eq.6) then !Blending Function suggested by Phillips
    func_v_nut = func_v_nut_blend(xi)
  end if
return; end function func_v_nut

real function func_v_nut_blend(xi)
  real :: xi,rh,ksh,yh,yc,vc,vcp,A2,A3,B1,B2
  rh = xi

```

```

ksh = 2.0*kr
yh = 1.0-rh
yc = 1.0-rc
A2 = 2.0*A0 - 0.5*(1.0 + a*gamma*ksh)
A3 = 0.5*(1.0 + a*gamma*ksh) - A0
vc = kappa*(A0 + gamma*ksh + A1*yc**2 - A2*rc**2 - A3*rc**4)*rc**(0.5+a)
vcp = (0.5+a)*kappa*(A0+gamma*ksh+A1*yc**2-A2*rc**2-A3*rc**4)*rc**(a-0.5)-
2.0*kappa*(A1*yc + A2*rc + 2.0*A3*rc**3)*rc**(0.5+a)
B1 = 3.0*(vc-B0)/rc**2 - vcp/rc;
B2 = vcp/rc**2 - 2.0*(vc-B0)/rc**3
if(rh.ge.rc) then
  func_v_nut_blend = kappa*(A0 + gamma*ksh + A1*yh**2 - A2*rh**2 -
A3*rh**4)*rh**(0.5+a)
else
  func_v_nut_blend = B0 + B1*rh**2 + B2*rh**3
end if
end function func_v_nut_blend

subroutine KaysCrawford_root()
real :: x1,x2,xnew,f1,f2,slope

ximax = 0.0
x1 = 0.2
x2 = 0.4
f1 = func_KC_nut_prime(x1)
do while (abs(f1) > 1.0e-12) !secant method for finding root
  f2 = func_KC_nut_prime(x2)
  slope = (f2-f1)/(x2-x1)
  xnew = x2 - f2/slope
  x1 = x2; x2 = xnew; f1 = f2
end do
ximax = xnew; nutmax = func_KC_nut(ximax)
end subroutine KaysCrawford_root

real function func_KC_nut(xi)
real :: xi,F

if(xi.lt.ximax) then
  func_KC_nut = nutmax
else
  F = kappa*(1.0-xi+2.0*gamma*kr)
  func_KC_nut = sqrt((kr/ksp)**2 + F**2*xi) - kr/ksp
end if
if(func_KC_nut.gt.kappa/6.0) func_KC_nut = kappa/6.0
return; end function func_KC_nut

real function func_KC_nut_prime(xi)
real :: xi,F,Fp

F = kappa*(1.0-xi+2.0*gamma*kr)
Fp = -kappa
func_KC_nut_prime = (F**2 + 2.0*F*Fp*xi)/(2.0*sqrt((kr/ksp)**2 + F**2*xi))
return; end function func_KC_nut_prime

!!!!!!!!!!!!!!!!!!!!!!!!!!!! General Functions !!!!!!!!!!!!!!!!!!!!!!!!!!!!!

real function func_Colebrook(Re)
real :: Re,old,new
if(Re<100.0) then
  func_Colebrook = 64.0/Re
else
  new = (-1.8*log10((kr/3.7)**(1.11) + 6.9/Re))**(-2.0)
  old = 10.0
  do while(abs((new-old)/new) .gt. 1.0e-5)
    old = new
    new = (-2.0*log10(kr/3.7 + 2.51/(Re*sqrt(old))))**(-2.0)
  end do
  func_Colebrook = new
end if
return; end function func_Colebrook

real function maximum(n,vec)
integer :: i,n
real :: vec(n)
maximum = vec(1)
do i=1,n
  if(vec(i).gt.maximum) maximum = vec(i)
end do

```

```

return; end function maximum

real function minimum(n,vec)
  integer :: i,n
  real :: vec(n)
  minimum = vec(1)
  do i=1,n
    if(vec(i).lt.minimum) minimum = vec(i)
  end do
return; end function minimum

subroutine rkta4(func,n,a,t0,y0,dt,dy,y)
!   This single precision subroutine computes a value for the n component
!   vector y(t0+dt) from a known value of the vector y(t0)=y0. The solution
!   is based on a fourth order Runge-Kutta solution to the system of n
!   differential equations,
!
!           dy(i)/dt = f(i,a,t,y)           i = 1,2,3,...,n
!
!   where a is a coefficient array passed to the functuon f. The single
!   precision function subprogram f(i,a,t,y) must be provided by the user.
!   func is a function pointer to be called

  integer :: n,j,i
  real :: a(*),t0,y0(n),dt,dy(n,4),y(n),c(4)
  interface
    real function func(n,i,a,t,y)
      integer :: n,i
      real :: a(*),t,y(n)
    end function
  end interface
  c(1) = 1.0/6.0
  c(2) = 1.0/3.0
  c(3) = c(2)
  c(4) = c(1)
  do j=1,n
    dy(j,1)=func(n,j,a,t0,y0)*dt
    y(j)=y0(j)+dy(j,1)/2.
  end do
  do j=1,n
    dy(j,2)=func(n,j,a,t0+dt/2.,y)*dt
  end do
  do j=1,n
    y(j)=y0(j)+dy(j,2)/2.
  end do
  do j=1,n
    dy(j,3)=func(n,j,a,t0+dt/2.,y)*dt
  end do
  do j=1,n
    y(j)=y0(j)+dy(j,3)
  end do
  do j=1,n
    dy(j,4)=func(n,j,a,t0+dt,y)*dt
  end do
  do j=1,n
    y(j)=y0(j)
    do i=1,4
      y(j)=y(j)+c(i)*dy(j,i)
    end do
  end do
  return
end subroutine rkta4

end module solver

!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!
program main
  use solver
  implicit none
  integer :: ID,ierror,irun

  ID = 0
  irun = 3
  call set_defaults()

```

```

call solver_allocate()
call create_grid()

write(*,*)
write(*,*)
write(*,*) '|-----|'
write(*,*) '| Fully Rough Pipe Flow Solver |'
write(*,*) '| By: Doug Hunsaker July 2011 |'
write(*,*) '|-----|'
write(*,*)
write(*,*) 'Select model (' ,model,' ) : '
write(*,*) ' 1: Phillips k-lambda Model'
write(*,*) ' 2: Nikuradse (Eq. 7.35) -|'
write(*,*) ' 3: Reichardt (Eq. 7.36) -|'
write(*,*) ' 4: Kays/Crawford (Eq. 7.42) -| Eddy Viscosity Models'
write(*,*) ' 5: Log-Law (Eq. 7.43) -|'
! write(*,*) ' 6: Blending Function (Eq. P.75) -|' !This function was suggested by
Phillips
read(5,'(a)') rec
if(rec .ne. ' ') read(rec,*) model

if(model.eq.1) then
  kappa = 0.404
  gamma = 0.0341
  write(*,*) 'Enter kappa (' ,kappa,' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) kappa

  write(*,*) 'Enter gamma (' ,gamma,' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) gamma

  write(*,*) 'Enter sigmak (' ,sigmak,' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) sigmak

  write(*,*) 'Enter C_lambda (' ,Clambda,' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) Clambda

  write(*,*) 'Enter Cr1 (' ,Cr1,' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) Cr1

  write(*,*) 'Enter ar1 (' ,ar1,' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) ar1

  write(*,*) 'Enter Cr2 (' ,Cr2,' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) Cr2

  write(*,*) 'Enter ar2 (' ,ar2,' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) ar2

  write(*,*) 'Enter Cr3 (' ,Cr3,' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) Cr3

  write(*,*) 'Enter ar3 (' ,ar3,' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) ar3

  write(*,*) 'Enter ar4 (' ,ar4,' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) ar4
end if

if(model.eq.6) then
  write(*,*) 'Select case for constant values (' ,modelcase,' ) : '
  write(*,*) ' 0: Enter specific constants'
  write(*,*) ' 1: Empirical correlation for Nikuradse eddy viscosity (Eq. 7.35)'
  write(*,*) ' 2: Empirical fit to Nikuradse Velocity Profile'
  write(*,*) ' 3: Approximation for Reichardt eddy viscosity profile'
  write(*,*) ' 4: Fit to Nikuradse and Reichardt eddy viscosity data'
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) modelcase

```

```

if(modelcase.eq.0) then
  write(*,*) 'Enter kappa (' ,kappa,') : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) kappa
  write(*,*) 'Enter gamma (' ,gamma,') : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) gamma
  write(*,*) 'Enter A0 (' ,A0,') : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) A0
  write(*,*) 'Enter A1 (' ,A1,') : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) A1
  write(*,*) 'Enter a (' ,a,') : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) a
  write(*,*) 'Enter rc (' ,rc,') : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) rc
  write(*,*) 'Enter B0 (' ,B0,') : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) B0
elseif(modelcase.eq.1) then
  A0 = 0.345
  A1 = 0.0
  a = 0.0
  rc = 0.0
  B0 = 0.0
elseif(modelcase.eq.2) then
  A0 = 0.5+0.5*gamma*kr
  A1 = A0
  a = 0.5
  rc = 0.0
  B0 = 0.0
elseif(modelcase.eq.3) then
  A0 = 0.297
  A1 = 0.0
  a = 0.0
  rc = 0.57
  B0 = .0667
elseif(modelcase.eq.4) then
  A0 = 0.36
  A1 = 0.36
  a = 0.5
  rc = 0.7
  B0 = 0.057
end if
end if

write(*,*) 'Select what to run (' ,irun,') : '
write(*,*) ' 1: Single case for specific kr and ks+'
write(*,*) ' 2: Nikuradse kr values at constant ks+'
write(*,*) ' 3: Nikuradse, Shockling, and other kr values over a range of ks+>1000'
write(*,*) '      (#3 returns the % RMS error)'
read(5,'(a)') rec
if(rec .ne. ' ') read(rec,*) irun

if((irun.eq.1).or.(irun.eq.2)) then
  if(irun.eq.1) then
    write(*,*) 'Enter kr (' ,kr,') : '
    read(5,'(a)') rec
    if(rec .ne. ' ') read(rec,*) kr
  end if
  write(*,*) 'Enter ks+ (' ,ksp,') : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) ksp
elseif(irun.eq.3) then
!
!   write(*,*) 'Enter a case ID number (used by the optimization routine) (' ,ID,') :
!
!   read(5,'(a)') rec
!   if(rec .ne. ' ') read(rec,*) ID
end if

if(irun.eq.1) then
  call run1()
elseif(irun.eq.2) then
  call run2()
elseif(irun.eq.3) then
  call run3(ID)

```

```

end if

call solver_deallocate()

end program main
!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!
subroutine run1()
  use solver
  implicit none
  integer :: ierror,i
  real :: local_velocity(m,1), local_viscosity(m,1)
  character*(50) :: filename1,filename2

  100 FORMAT (1X, 1000ES25.15)
  filename1 = 'viscosity.txt'
  filename2 = 'velocity.txt'

  open(unit = 10, File = filename1, status="replace", action = "write", iostat =
ierror)
  open(unit = 20, File = filename2, status="replace", action = "write", iostat =
ierror)
  write(10,*) '      y/R                                kr=',kr,' ks+=',ksp
  write(20,*) '      y/R                                kr=',kr,' ks+=',ksp

  call run_case(local_viscosity(:,1),local_velocity(:,1),1)

  do i=m,1,-1
    write(10,100) 1.0-r_hat(i),local_viscosity(i,:)
    write(20,100) 1.0-r_hat(i),local_velocity(i,:)
  end do
  close(10)
  close(20)

end subroutine run1
!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!
subroutine run2()
  use solver
  implicit none
  integer :: ierror,i
  real :: local_velocity(m,6), local_viscosity(m,6), krvals(6)
  character*(50) :: filename1,filename2
  krvals(1) = 0.034
  krvals(2) = 0.016
  krvals(3) = 0.0083
  krvals(4) = 0.0039
  krvals(5) = 0.0020
  krvals(6) = 0.00098

  100 FORMAT (1X, 1000ES25.15)
  filename1 = 'viscosity.txt'
  filename2 = 'velocity.txt'

  open(unit = 10, File = filename1, status="replace", action = "write", iostat =
ierror)
  open(unit = 20, File = filename2, status="replace", action = "write", iostat =
ierror)
  write(10,*) '      y/R                                kr=0.034                kr=0.016
kr=0.0083                &
kr=0.0039                kr=0.0020                kr=0.00098'
  write(20,*) '      y/R                                kr=0.034                kr=0.016
kr=0.0083                &
kr=0.0039                kr=0.0020                kr=0.00098'

  do i=1,6
    kr = krvals(i)
    call run_case(local_viscosity(:,i),local_velocity(:,i),1)
  end do

  do i=m,1,-1
    write(10,100) 1.0-r_hat(i),local_viscosity(i,:)
    write(20,100) 1.0-r_hat(i),local_velocity(i,:)
  end do
  close(10)
  close(20)

```

```

end subroutine run2
!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!
subroutine run3(ID)
  use solver
  implicit none
  integer :: ierror,i,ID,ngood,total
  real :: local_velocity(m,1), local_viscosity(m,1), kivals(8)
  real :: fitness,power,Ni_rough
  character*(50) :: filename
  kivals(1) = 0.034
  kivals(2) = 0.016
  kivals(3) = 0.0083
  kivals(4) = 0.0039
  kivals(5) = 0.0020
  kivals(6) = 0.00098
  kivals(7) = 0.00030
  kivals(8) = 0.000058

  100 FORMAT (1000ES25.15)
  write(filename,*) ID
  filename = 'case_'//trim(adjustl(filename))//'.txt'

  open(unit = 10, File = filename, status="replace", action = "write", iostat = ierror)
  write(10,*) ' kr Re_bulk 4CF
Nikuradse_FullyRough Colebrook'

  fitness = 0.0
  total = 0
  do i=1,8
    power = 3.0
    kr = kivals(i)
    Ni_rough = (2.0*log10(3.7/kr))**(-2.0)
    ngood = 0
    do while((Re_bulk<1.0e8).or.(ngood<3))
      ksp = 10**power
      call run_case(local_viscosity(:,1),local_velocity(:,1),0)
      if((Re_bulk<1.0e8).or.(ngood<3)) then

Ni_rough = func_Colebrook(Re_bulk)!This line should be commented out in the future, but
in the original work, the fitness
!was based on the deviation from the Colebrook equation, not the Nikuradse number. So
leaving that line uncommented
!yields results from the original work, where in the future, it should be compared to the
Nikuradse number

      fitness = fitness + (abs(Ni_rough-Cf4)/Ni_rough)**2
      ngood = ngood + 1
      power = power + 0.25
      write(10,100) kr,Re_bulk,Cf4,Ni_rough,func_Colebrook(Re_bulk)
    end if
  end do
  write(10,100)
  total = total + ngood
end do
close(10)

fitness = 100.0*sqrt(fitness/real(total))

write(filename,*) ID
filename = 'fitness_'//trim(adjustl(filename))//'.txt'
open(unit = 20, File = filename, status="replace", action = "write", iostat = ierror)
write(20,*) fitness,' = % RMS Error'
close(20)

write(*,*) '% RMS Error = ',fitness

end subroutine run3
!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!
subroutine run_case(local_viscosity,local_velocity,iprint)
  use solver
  implicit none
  integer :: niter,iprint
  real :: local_viscosity(m),local_velocity(m)
  if(model.eq.1) then
    call Phillips_k1(iprint,niter)

```


APPENDIX P

OPTIMIZATION CODE

The following Fortran code can be used in conjunction with an a.out executable to optimize input parameters.

```

module bfgs
  implicit none
  integer :: nvars
  integer :: diff_scheme !1 = central difference. 0 = forward difference
  integer :: iter
  integer :: nsearch

  real :: default_alpha
  real :: diff_delta
  real :: fitness_curr
  real :: stop_delta
  real :: alpha

  integer,allocatable :: opton(:)
  real,allocatable :: grad(:)
  real,allocatable :: vars(:)
  real,allocatable :: s(:)

contains

!-----

subroutine opt_allocate()
  allocate(opton(nvars))
  allocate(grad(nvars))
  allocate(vars(nvars))
  allocate(s(nvars))
  opton = 0; vars = 0.0; grad = 0.0; s = 0.0
end subroutine opt_allocate

!-----

subroutine opt_deallocate()
  deallocate(opton)
  deallocate(grad)
  deallocate(vars)
  deallocate(s)
end subroutine opt_deallocate

!-----

subroutine opt_run()
  integer :: i_iter,o_iter,i,ierror
  real :: vars_orig(nvars),vars_old(nvars),grad_old(nvars)
  real :: dx(nvars,1),NG(nvars,1),N(nvars,nvars),gamma(nvars,1)
  real :: mag_dx,denom
  character(LEN=50)::fn,command
  110 format (1X, I10, 100ES22.13)

  fn = 'optimization.txt'
  open(unit = 1001, File = fn, action = "write", iostat = ierror)
  write(1001,*) 'iter o_it i_it  sigma_k          Cr1          ar1
Cr2          &
ar4          &          &ar2          Cr3          ar3
mag(dx)'          &          &kr          Fitness          alpha
  close(1001)
  open(unit = 1001, File = 'gradient.txt', action = "write", iostat = ierror)
  write(1001,*) 'iter o_it i_it  sigma_k          Cr1          ar1
Cr2          &

```

```

ar4          &          &ar2          Cr3          ar3
              &          &kr          Fitness          alpha
mag(dx) '
close(1001)

command = 'rm input_* output_* case_* fitness_*'
call system(command)

write(*,*) 'Beginning Optimization Routine'
write(*,*) 'Optimization Variables: '
do i=1,nvars
  write(*,110) opton(i),vars(i)
end do
write(*,*) '          default alpha : ',default_alpha
write(*,*) 'differenceing delta : ',diff_delta
write(*,*) 'differencing scheme : ',diff_scheme
write(*,*) '          stopping delta : ',stop_delta
write(*,*) 'simultaneous search : ',nsearch

o_iter = 0
mag_dx = 1.0
do while(mag_dx > stop_delta)
  vars_orig = vars
  i_iter = 0

  do while(mag_dx > stop_delta)
    call gradient()
    call append_file(fn,o_iter,i_iter,mag_dx)

    if(i_iter .eq. 0) then !set N=identity
      N = 0.0
      do i=1,nvars
        N(i,i) = 1.0 !N = identity matrix
      end do
    else
      dx(:,1) = vars(:) - vars_old(:)
      gamma(:,1) = grad(:) - grad_old(:)
      NG(:,1) = matmul(N,gamma(:,1))
      denom = dot_product(dx(:,1),gamma(:,1))
      ! N = N + matmul(dx-NG,transpose(dx-NG))/dot_product(dx(:,1)-
NG(:,1),gamma(:,1)) !Rank One Hessian Inverse Update
      N = N +
(1.0+dot_product(gamma(:,1),NG(:,1)))/denom)*(matmul(dx,transpose(dx))/denom) & !BFGS
Update
      & - (matmul(dx,matmul(transpose(gamma),N)) +
matmul(NG,transpose(dx)))/denom
    end if
    s(:) = -matmul(N,grad)
    vars_old = vars
    grad_old = grad

    call line_search()

    dx(:,1) = vars(:) - vars_old(:)
    mag_dx = sqrt(dot_product(dx(:,1),dx(:,1)))
    i_iter = i_iter + 1
    iter = iter + 1
  end do

  call append_file(fn,o_iter,i_iter,mag_dx)

  dx(:,1) = vars(:) - vars_orig(:)
  mag_dx = sqrt(dot_product(dx(:,1),dx(:,1)))
  o_iter = o_iter + 1
end do

call sleep(1)
fitness_curr = case_fitness single(0)
call append_file(fn,o_iter,i_iter,mag_dx)
end subroutine opt_run

!-----
subroutine write_bfgs_file(fn)
character(50) :: fn
integer :: ierror
110 format (100ES22.13)
120 format (100I22)

```

```

fn = trim(adjustl(fn))
write(*,*) 'writing ',fn
open(unit = 200, File = fn, status = "replace", action = "write", iostat = ierror)
write(200,*) nvars,'          num vars'
write(200,*) 'Variable names can be inserted on this line'
write(200,110) vars(:)
write(200,120) opton(:)
write(200,*) default_alpha, '    default line search alpha'
write(200,*) diff_delta, '    delta step size used for gradient calculations'
write(200,*) diff_scheme, '    differencing scheme (1=central diff, 0=forward diff)'
write(200,*) stop_delta, '    stop delta'
write(200,*) nsearch, '    number of simultaneous cases in the line search'
close(200)
end subroutine write_bfgs_file

```

```

!-----
subroutine append_file(fn,o_iter,i_iter,mag_dx)
character(50)::fn
real :: mag_dx
integer :: o_iter,i_iter,ierror
110 format (3I5, 100ES22.13)
write(*,110) iter,o_iter,i_iter,vars(:),fitness_curr,alpha,mag_dx
open(unit = 1001, File = fn, status = "OLD", access = "append", iostat = ierror)
write(1001,110) iter,o_iter,i_iter,vars(:),fitness_curr,alpha,mag_dx
close(1001)
open(unit = 1001, File = 'gradient.txt', status = "OLD", access = "append", iostat =
ierror)
write(1001,110) iter,o_iter,i_iter,grad(:),fitness_curr,alpha,mag_dx
close(1001)
end subroutine append_file

```

```

!-----
real function case_fitness(case_num)
integer :: case_num,ierror
character(50)::filename

write(filename,*) case_num
filename = 'fitness_'//trim(adjustl(filename))//'.txt'
open(unit = 10, File = filename, action = "read", iostat = ierror)
read(10,*) case_fitness
close(10)
end function case_fitness

```

```

!-----
subroutine forward_diff()
integer :: i
character(50) :: command
real :: vars_orig(nvars)
vars_orig(:) = vars(:)

grad = 0.0
call start_case(0)
do i=1,nvars
  if(opton(i).eq.1) then
    vars(i) = vars(i) + diff_delta
    call start_case(i)
    vars(:) = vars_orig(:)
  end if
end do

do while(.not.all_done())
  call sleep(1)
end do
call sleep(1) !one more time to ensure all files are totally written

fitness_curr = case_fitness(0)
do i=1,nvars
  if(opton(i).eq.1) grad(i) = (case_fitness(i) - fitness_curr)/diff_delta
end do

call sleep(1)
command = 'mv case_0.txt curr_case.txt'
call system(command)
command = 'rm input_* output_* case_* fitness_*'
call system(command)

```

```

end subroutine forward_diff

!-----

subroutine gradient()
  real :: temp(nvars)
  call forward_diff()
  if(diff_scheme.eq.1) then
    temp(:) = grad(:)
    diff_delta = -diff_delta
    call forward_diff()
    grad(:) = 0.5*(grad(:) + temp(:))
    diff_delta = -diff_delta
  end if
end subroutine gradient

!-----

subroutine line_search()
  real :: local_fitness, f1, f2, f3, a1, a2, a3, da
  real :: xval(0:nsearch), yval(0:nsearch), vars_orig(nvars)
  integer :: i, j, mincoord
  write(*,*) 'line search -----'

  alpha = max(default_alpha, 1.1*stop_delta/sqrt(dot_product(s(:), s(:))))
  vars_orig(:) = vars(:)

  xval(0) = 0.0; yval(0) = fitness_curr
  do
    call run_mult_cases(nsearch, alpha, vars_orig, xval(1:nsearch), yval(1:nsearch))
    do j=0, nsearch
      write(*,*) j, xval(j), yval(j)
    end do
    if(yval(1)>yval(0)) then
      if(alpha*sqrt(dot_product(s(:), s(:))) < stop_delta) then
        write(*,*) 'Line search within stopping tolerance : alpha = ', alpha
        return
      end if
      write(*,*) 'Too big of a step. Reducing Alpha'
      alpha = 0.5*alpha
    else
      mincoord = minimum_coordinate(nsearch+1, yval)-1
      write(*,*) 'mincoord = ', mincoord
      if(mincoord.ne.nsearch) exit
      alpha = 2.0*alpha
    end if
  end do
  a1 = xval(mincoord-1)
  a2 = xval(mincoord)
  a3 = xval(mincoord+1)
  f1 = yval(mincoord-1)
  f2 = yval(mincoord)
  f3 = yval(mincoord+1)

  da = a2-a1
  alpha = a1+da*(4.0*f2-f3-3.0*f1)/(2.0*(2.0*f2-f3-f1))
  if((alpha > a3).or.(alpha < a1)) then !For parabolas whose min is not in bounds
    alpha = a2
    if(f2 > f1) alpha = a1
  end if
  vars(:) = vars_orig(:) + alpha*s(:)
  write(*,*) 'final alpha = ', alpha
end subroutine line_search

!-----

integer function minimum_coordinate(num, vals)
  integer :: num, i
  real :: vals(num), minval
  minval = vals(1)
  minimum_coordinate = 1
  do i=2, num
    if(vals(i)<minval) then
      minval = vals(i)
      minimum_coordinate = i
    else
      exit
    end if
  end do
end function

```

```

end function minimum_coordinate

!-----

subroutine run_mult_cases(ncases,start_alpha,vars_orig,x,y)
  integer :: ncases,i
  real :: start_alpha,vars_orig(nvars)
  real :: x(ncases),y(ncases)
  character(50) :: command

  do i=1,ncases
    x(i) = real(i)*start_alpha
    vars(:) = vars_orig(:) + x(i)*s(:)
    call start_case(i)
  end do
  vars(:) = vars_orig(:)
  do while(.not.mult_done(ncases))
    call sleep(1)
  end do
  call sleep(1) !one more time to ensure all files are totally written

  do i=1,ncases
    y(i) = case_fitness(i)
  end do
  call sleep(1)

  command = 'rm input_* output_* case_* fitness_*'
  call system(command)
end subroutine run_mult_cases

!-----

logical function mult_done(ncases)
  implicit none
  integer :: ncases,ios(ncases),i
  character(50)::filename

  do i=1,ncases
    write(filename,*) i
    filename = 'fitness_'//trim(adjustl(filename))//'.txt'
    open(i*100,file=filename,status='old',iostat=ios(i))
  end do
  if(count(ios==0)==size(ios)) then
    mult_done = .true.
  else
    mult_done = .false.
  end if

  do i=1,ncases
    if(ios(i)/=0) cycle
    close(i*100)
  end do
end function mult_done

!-----

real function case_fitness_single(case_num)
  integer :: case_num,ierror
  character(50)::filename,command

  call start_case(case_num)
  do while(.not.one_done(case_num))
    call sleep(1)
  end do
  call sleep(1) !one more time to ensure file is totally written

  case_fitness_single = case_fitness(case_num)
  command = 'rm input_* output_* case_* fitness_*'
  call system(command)
end function case_fitness_single

!-----

subroutine start_case(case_num)
  implicit none
  integer :: case_num,ierror,i
  character(50)::file_i,file_o,command

  write(file_o,*) case_num
  file_i = 'input_'//trim(adjustl(file_o))//'.txt'

```

```

file_o = 'output_'//trim(adjustl(file_o))//'.txt'
open(unit = 10, File = file_i, status="replace", action = "write", iostat = ierror)
do i=1,nvars
  write(10,*) vars(i)
end do
write(10,*) case_num
close(10)
command = './a.out < '//trim(adjustl(file_i))//' > '//trim(adjustl(file_o))//' &'
!   write(*,*) command
call system(command)
end subroutine start_case

```

```

!-----
logical function one_done(case_num)
  implicit none
  integer::case_num,ios
  character(50)::filename

  write(filename,*) case_num
  filename = 'fitness_'//trim(adjustl(filename))//'.txt'
  open(100,file=filename,status='old',iostat=ios)
  if(ios==0) then
    one_done = .true.
    close(100)
  else
    one_done = .false.
  end if
end function one_done

```

```

!-----
logical function all_done()
  implicit none
  integer ::ios(nvars+1),i
  character(50)::filename

  !Check for 0_file
  open(100,file="fitness_0.txt",status='old',iostat=ios(nvars+1))
  if(ios(nvars+1)==0) close(100)

  do i=1,nvars
    if(opton(i).eq.1) then
      write(filename,*) i
      filename = 'fitness_'//trim(adjustl(filename))//'.txt'
      open(i*100,file=filename,status='old',iostat=ios(i))
    else
      ios(i)=0
    end if
  end do
  if(count(ios==0)==size(ios)) then
    all_done = .true.
  else
    all_done = .false.
  end if

  do i=1,nvars
    if(ios(i)/=0) cycle
    close(i*100)
  end do
end function all_done

end module bfgs

```

```

!-----MAIN PROGRAM -----
program main
  use bfgs
  implicit none
  character*(50) :: rec,fn
  integer :: i,ierror

  !Defaults
  iter = 0
  default_alpha = 1.0e-8
  diff_delta = 1.0e-8
  diff_scheme = 1
  stop_delta = 1.0e-12
  nsearch = 8

```

```

fn = 'none'
write(*,*) 'Enter filename (none=use interactive console instead of file) (' ,fn, ' )
:'
read(5,'(a)') rec
if(rec .ne. ' ') read(rec,*) fn

if(fn.ne.'none') then
  open(unit = 10, File = fn, action = "read", iostat = ierror)
  read(10,*) nvars
  read(10,*) !names of variables

  call opt_allocate()

  read(10,*) vars(1:nvars)
  read(10,*) opton(1:nvars)
  read(10,*) default_alpha
  read(10,*) diff_delta
  read(10,*) diff_scheme
  read(10,*) stop_delta
  read(10,*) nsearch
  close(10)
else
  write(*,*) 'Enter number of variables : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) nvars

  call opt_allocate()

  do i=1,nvars
    write(*,*) 'Enter variable ',i, ' : '
    read(5,'(a)') rec
    if(rec .ne. ' ') read(rec,*) vars(i)
    write(*,*) 'Optimize this variable? (1=yes,0=no) (' ,opton(i), ' ) : '
    read(5,'(a)') rec
    if(rec .ne. ' ') read(rec,*) opton(i)
  end do

  write(*,*) 'Enter default line search alpha (' ,default_alpha, ' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) default_alpha

  write(*,*) 'Enter delta step size used for gradient calculations (' ,diff_delta, ' )
:'
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) diff_delta

  write(*,*) 'Enter differencing scheme (1=central diff, 0=forward diff)
(' ,diff_scheme, ' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) diff_scheme

  write(*,*) 'Enter stop delta (' ,stop_delta, ' ) : '
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) stop_delta

  write(*,*) 'Enter number of simultaneous cases in the line search (' ,nsearch, ' )
:'
  read(5,'(a)') rec
  if(rec .ne. ' ') read(rec,*) nsearch
end if

fn = 'bfgs_start.txt'; call write_bfgs_file(fn)
call opt_run()
fn = 'bfgs_end.txt'; call write_bfgs_file(fn)

call opt_deallocate()
end program main

```

APPENDIX Q

CLOSURE COEFFICIENT EVALUATION ON THE FLAT

The following code can be used to evaluate the closure coefficients on the flat given a value for σ_k .

```

subroutine Flat(sigma_k,C_r2,a_r2,C_r3,a_r3,a_r4)
implicit none
real sigma_k,C_r2,a_r2,C_r3,a_r3,a_r4
if(sigma_k.le.0.10)then
  a_r2=-0.048511567*sigma_k**2 + 0.001955827*sigma_k + 0.012102649
else if(sigma_k.lt.1.75)then
  a_r2=-0.000206593*sigma_k**4 + 0.004754408*sigma_k**3 - 0.018765359*sigma_k**2 &
    -0.001559875*sigma_k + 0.012141171
else if(sigma_k.le.2.00)then
  a_r2=-0.028076632*sigma_k + 0.024614722
else
  a_r2=-0.000061468*sigma_k**4 + 0.000684628*sigma_k**3 - 0.000750473*sigma_k**2 &
    -0.031056442*sigma_k + 0.029083802
end if
if(sigma_k.le.0.10)then
  a_r3=0.010523059*sigma_k**2 - 0.004315527*sigma_k + 0.005620078
else if(sigma_k.lt.1.75)then
  a_r3=-0.000357793*sigma_k**4 + 0.003598258*sigma_k**3 - 0.010812353*sigma_k**2 &
    -0.000967269*sigma_k + 0.005489307
else if(sigma_k.le.2.00)then
  a_r3=-0.013340859*sigma_k + 0.009956552
else
  a_r3=-0.000003988*sigma_k**4 - 0.000034537*sigma_k**3 + 0.001537985*sigma_k**2 &
    -0.018724532*sigma_k + 0.014912497
end if
if(sigma_k.le.0.10)then
  a_r4=-0.01893511*sigma_k**2 - 0.00127611*sigma_k + 1.13352452
else if(sigma_k.lt.1.75)then
  a_r4=-0.00053239*sigma_k**4 + 0.00534831*sigma_k**3 - 0.01699619*sigma_k**2 &
    -0.00109238*sigma_k + 1.13347404
else if(sigma_k.le.2.00)then
  a_r4=-0.02288151*sigma_k + 1.14322147
else
  a_r4=-0.00001166*sigma_k**4 + 0.0001342*sigma_k**3 + 0.00070418*sigma_k**2 &
    -0.02678935*sigma_k + 1.14733412
end if
if(sigma_k.lt.0.10)then
  C_r2=0.00173260*sigma_k**(-0.49813563)
else if(sigma_k.le.0.25)then
  C_r2=0.00179141*sigma_k**(-0.48401989)
else if(sigma_k.le.2.00)then
  C_r2=0.00179141*sigma_k**(-0.48401989) + 0.0000271629*sigma_k**4 -
    0.0001736396*sigma_k**3 &
    +0.0003974614*sigma_k**2 + 0.0000754681*sigma_k - 0.0000412428
else
  C_r2=0.00179141*sigma_k**(-0.48401989) + 0.0000002950*sigma_k**4 -
    0.0000011327*sigma_k**3 &
    -0.0000292549*sigma_k**2 + 0.0005645091*sigma_k - 0.0002627585
end if
if(sigma_k.lt.0.10)then
  C_r3=0.0558450*sigma_k**(-0.4986960)
else if(sigma_k.le.0.25)then
  C_r3=0.0571034*sigma_k**(-0.4892824)
else if(sigma_k.le.2.00)then
  C_r3=0.0571034*sigma_k**(-0.4892824) + 0.00073867*sigma_k**4 - 0.00434008*sigma_k**3 &
    +0.00809117*sigma_k**2 + 0.00185242*sigma_k - 0.00090757
else
  C_r3=0.0571034*sigma_k**(-0.4892824) - 0.00002129*sigma_k**4 + 0.00039801*sigma_k**3 &
    -0.00311039*sigma_k**2 + 0.01389982*sigma_k - 0.00594463
end if
return
end

```

CURRICULUM VITAE

Douglas F. Hunsaker
December 2011

EDUCATION

- | | | | |
|---|--------------------------|----------|-----------------|
| PhD Mechanical Engineering | Utah State University | 3.97 GPA | <i>Dec 2011</i> |
| Dissertation: Evaluation of an Alternate Incompressible Energy-Enstrophy Turbulence Model | | | |
| MS Mechanical Engineering | Brigham Young University | 3.63 GPA | <i>Apr 2007</i> |
| Thesis: A Numerical Vortex Approach to Aerodynamic Modeling of SUAV/VTOL Aircraft | | | |
| BS Mechanical Engineering | Brigham Young University | 3.58 GPA | <i>Dec 2005</i> |

EXPERIENCE

Aerodynamics / Numerical Fluids

- Developed compressible codes capable of predicting the lift-to-drag ratio of a half-angle cone hypersonic wave rider and a supersonic airfoil. Evaluated the effects of grid resolution on a supersonic NACA airfoil in Fluent.
- Taught a full semester course on aerodynamics for senior-level and graduate students in the Mechanical and Aerospace Engineering Department at Utah State University.
- Authored a numerical lifting-line code for modeling interactions of multiple lifting surfaces with deflected control surfaces. Added a blade-element propeller model to account for propeller effects on the wings. The code required only a fraction of the computational cost of conventional CFD methods and was used almost real-time in a flight simulator to predict the aerodynamics of a VTOL airframe as it transitioned from vertical take-off to horizontal flight.
- Developed a 3D object-oriented potential flow solver for Sandia National Laboratories in Fortran to interface with their in-house code. The code can be used to model the potential flow effects of bodies near turbine blades.
- Authored an interactive 2D finite-volume RANS-based turbulent flow code with pressure coupling in C++. The code included coordinate transformations to allow for second-order computations on a non-uniform grid.
- Developed a 1D finite-difference turbulent flow solver for channel and pipe flow in Fortran. Implemented four traditional RANS models including $k-\epsilon$ and $k-\omega$ models. Authored a higher-order finite differencing scheme allowing the models to be solved using 6th-order finite differencing.
- Decreased computation time to 1/5th of the original time by parallelizing time-consuming subroutines of a CFD code.

Flight Mechanics / Simulation

- Developed a 6-DOF flight simulator based on quaternion attitude estimation and fourth-order Runge-Kutta time integration. Added user-interface components including landing detection and an intuitive HUD using OpenGL. Used the code to demonstrate common dynamic modes of aircraft to university students.
- Taught a full course on the mechanics of flight for four semesters at Utah State University. Received exceptionally high marks as a teacher. Most recent teacher evaluation: 5.6/6.0.
- Authored a linearized, coupled longitudinal and lateral aircraft dynamic stability code in Fortran. The code was used to evaluate the handling qualities of a newly-designed airframe.
- Redesigned the empennage on the RQ-1 Predator UAV for a class project. The new design decreased empennage weight by 30% and met specific static stability requirements.

Aircraft Research and Development

- Conducted wind tunnel tests to evaluate the stall characteristics of a low aspect ratio wing. Supported the design of the test setup, instrument construction, data collection, and post data analysis.
- Designed, built, and flight tested a 4-foot UAV for a research lab at Brigham Young University (team size: 2). The final airframe increased flight endurance over previous designs by a factor of 4, solved a costly landing problem, doubled user viewing range, and matched design endurance to within 5%.
- Propulsion system design lead in the development of an 18-inch UAV airframe (team size: 6). Ran design optimization studies to select motor, speed controller, battery, and propeller combination. Piloted initial flight tests. Later, on a team of 2, increased the endurance of the aircraft by 50% through an aerodynamic optimization study.

Business Development

- Submitted a proposal to a government subcontractor as part of a small business venture. Received the contract to complete a cost-benefit analysis for an aircraft designed for low-altitude air sampling. This consulting work included evaluating advantages of several different airframe types, interfacing with potential suppliers, and estimating costs and risks involved with each configuration.
- Authored a proposal for the construction of three hand-launchable long-endurance airframes for Flying Sensors, a local aerial photography company.
- Established a business relationship and conducted the initial meeting between an international company and AeroVironment, Inc.

Teaching / Leadership

- Taught full semester graduate courses on aerodynamics and flight mechanics in the Mechanical Engineering and Aerospace Department at Utah State University.
- Instructed university students in preparation for the Fundamentals of Engineering Exam.
- Consulted university senior design teams during airframe design phases.
- President: community organization, 50 members – conducted weekly organizational activities.
- Trained non-profit organization associates in Frankfurt, Germany – Fluent in conversational German.

AWARDS / FELLOWSHIPS

- NASA Space Grant Consortium Fellowship, *annually renewed 2005-2010*.
- International Society of Transport Aircraft Trading (ISTAT) Scholarship, *2007-2008*.
- 1ST Place Western Regional AIAA Student Competition Masters Division, *2006*.
- Office of Research and Creative Activities Graduate Research Fellowship, *BYU, 2006-2007*.
- International Science and Engineering Fair Contestant, *1997, 1998, 1999*.

PUBLICATIONS

Journal Publications

Phillips, W. F., Hunsaker, D. F., and Niewoehner, R. J., "Estimating the Subsonic Aerodynamic Center and Moment Components for Swept Wings," *Journal of Aircraft*, Vol. 45, No. 3, pp. 1033-1043, June 2008.

Hunsaker, D. F., and Phillips, W. F., "Momentum Theory with Slipstream Rotation Applied to Wind Turbines," *Wind Energy*, submitted for review.

Phillips, W. F., Hunsaker, D. F., and Spall, R. E., "Smooth-Wall Boundary Conditions for Dissipation-Based Turbulence Models," will be submitted soon to a computational mathematics journal.

Meeting Papers

- Hunsaker, D. F., "A Numerical Lifting-Line Method Using Horseshoe Vortex Sheets," Rocky Mountain Space Grant Consortium Meeting Proceedings, Logan, Utah, May 2011.
- Hunsaker, D. F., "A One-Dimensional Finite-Difference Solver for Fully-Developed Pipe and Channel Flows," Rocky Mountain Space Grant Consortium Meeting Proceedings, Logan, Utah, May 2010.
- Phillips, W. F., Hunsaker, D. F., and Spall, R. E., "Smooth-Wall Boundary Conditions for Dissipation-Based Turbulence Models," 48th AIAA Aerospace Sciences Meeting, Orlando, Florida, Jan. 4-7, 2010.
- Hunsaker, D. F., "Application of a Coordinate Transformation and Discretization Method for Computational Fluid Dynamics," Rocky Mountain Space Grant Consortium Meeting Proceedings, Salt Lake City, Utah, May 2009.
- Phillips, W. F., Hunsaker, D. F., Alley, N. R., and Niewoehner, R. J., "Pitch Dynamics of Unmanned Aerial Vehicles," 47th AIAA Aerospace Sciences Meeting, Orlando, Florida, Jan. 5-8, 2009, AIAA-2009-62.
- Hunsaker, D. F., "Evaluation of an Alternate Incompressible Energy-Enstrophy Turbulence Model," Rocky Mountain Space Grant Consortium Meeting Proceedings, Salt Lake City, Utah, May 2008.
- Phillips, W. F., Hunsaker, D. F., and Niewoehner, R. J., "Estimating the Subsonic Aerodynamic Center and Moment Components for Swept Wings," 46th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, Jan. 7-10, 2008, AIAA-2008-192.
- Hunsaker, D. F., "Post Stall Behavior of a Lifting Line Algorithm," Rocky Mountain Space Grant Consortium Meeting Proceedings, Salt Lake City, Utah, May 2007.
- Hunsaker, D. F., Larson, G., Condie, S., "Optimization of 'Iris', a Small Autonomous Surveillance UAV," 45th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, Jan. 8-11, 2007, AIAA-2007-360.
- Hunsaker, D. F., "A Numerical Blade Element Approach to Estimating Propeller Flowfields," 45th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, Jan. 8-11, 2007, AIAA-2007-374.
- Hunsaker, D. F., Snyder, D. O., "A Lifting-Line Approach to Estimating Propeller / Wing Interactions," 24th Applied Aerodynamics Conference, San Francisco, California, June 5-8, 2006, AIAA-2006-3466.
- Hunsaker, D. F., "A Numerical Vortex Approach to Aerodynamic Modeling of SUAV/VTOL Aircraft," Rocky Mountain Space Grant Consortium Meeting Proceedings, Salt Lake City, Utah, May 2006.
- Boyce, J., Carr, R., Chipman, D., Larson, G., Hopkins, N., Hunsaker, D., and Bowman, W. J., "Design of 'Iris', a Small Autonomous Surveillance UAV," 44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, Jan. 9-12, 2006, AIAA-2006-824.