

TURBULENCE MODELS WITH ADAPTIVE MESHING FOR INDUSTRIAL CFD

A thesis presented by

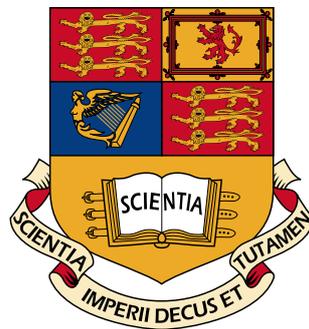
JONATHAN R. BULL

in partial fulfilment of the requirements for the degree of
Doctor of Engineering in Computational Fluid Dynamics
and the Diploma of Imperial College London

Supervisors:

Dr Matthew D. Piggott
Prof. Chris C. Pain
Ian Kiltie

Applied Modelling and Computational Physics Group
Department of Earth Science and Engineering
Imperial College London



February, 2013

Declaration

I herewith certify that all material in this dissertation which is not my own work has been properly acknowledged.

Jonathan R. Bull

Prologue

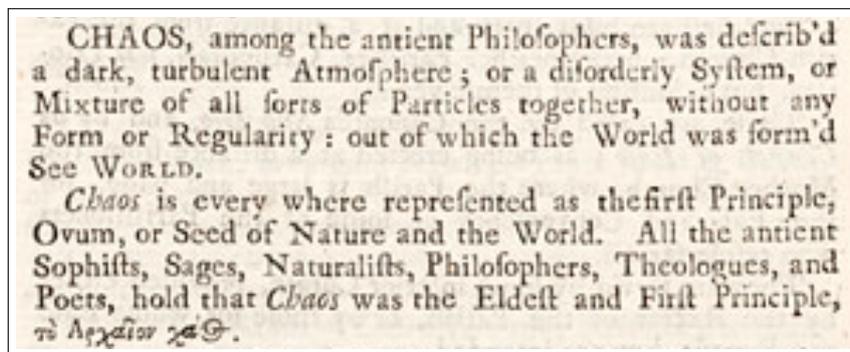


Figure 1: Excerpt from the entry on chaos in one of the first definitive compendia of Western scientific knowledge, Chambers' Cyclopaedia of 1728 (Chambers, 1728).

As this extract from Chambers (1728) illustrates, the chaotic turbulent dynamics of the world we inhabit have fascinated us throughout the ages, from the great thinkers of the classical era, to Leonardo da Vinci, into the Renaissance period and the dawning of the modern scientific age and right up to the present day. Even now we are far from a complete understanding of turbulence and many fundamental challenges remain for the future.

Abstract

Computational fluid dynamics (CFD) and affordable computing power have advanced considerably in recent years, bringing full 3D simulation of complex high Reynolds number flows within reach of industry. However, providing accurate and trustworthy results in diverse flows with constraints on computational resources is still a considerable challenge. Owing to the complexity of commonly-encountered turbulent flows, robust turbulence models are required which do not have to be manually tuned to specific flow conditions to ensure their accuracy.

In this regard, a highly effective approach is unstructured mesh adaptivity which automatically refines or coarsens the mesh locally in order to achieve a desired accuracy with minimum computational effort. However, the use of such adaptive meshes with turbulence models raises questions about the origins and interactions of various errors. This thesis describes the development, verification and validation of robust turbulence models suited to high Reynolds number single-phase turbulent flow using unstructured adaptive meshes.

The main focus of this thesis is a new tensorial dynamic large eddy simulation (LES) model. The novel combination of the dynamic LES method with a tensorial definition of filter width is ideal for capturing the anisotropy and inhomogeneity of turbulence. This model is designed for use with unstructured mesh adaptivity, which enables the simulation of turbulent flow with high efficiency in terms of mesh resolution. Furthermore, the model is robust since both the resolution and the sub-filter-scale (SFS) stresses adapt to local flow conditions so that little *a priori* knowledge of the flow is required. Verification tests of the filtering method and validation of the new LES model in the 3D backward-facing step are presented.

To provide context for the research, the contribution made by CFD simulations to the analysis of nuclear reactor safety and performance is discussed. The practicalities of performing simulations on high performance computing (HPC) facilities are also discussed. Background theory necessary to understand the research is presented, including a mathematical description of turbulent flow and the classes of CFD methods used to approximate it. A

review of turbulence models, discretisation methods, boundary conditions and adaptive meshing methods is included.

The construction and testing of a Reynolds-averaged Navier-Stokes (RANS) $k - \varepsilon$ turbulence model and a scale-adaptive very large eddy simulation (VLES) model in the open-source CFD code Fluidity are also described. The development of a law-of-the-wall boundary condition for turbulent flow in variational (weak) form is also presented. Verification tests are performed to establish that the $k - \varepsilon$ model has been coded correctly. Validation of the RANS model and the wall function using fixed and adaptive meshes is carried out in the 2D backward-facing step.

Finally, results of simulations of a vortex diode device using various turbulence models are presented and compared to results from the commercial CFD code CFX and experimental results. This study was carried out during the industrial component of the Engineering Doctorate, which was intended to further the development and understanding of CFD at Rolls-Royce Nuclear. The device presents a challenging test case for CFD but some useful conclusions are reached about how to model it. The thesis concludes with a summary of findings and proposals for further research.

Acknowledgements

I am deeply grateful to all those who helped me with the research presented in this thesis. When I came for interview at the Applied Modelling and Computation Group (AMCG) at Imperial College London, I got the impression of a well-run, closely-knit group in which I would receive plenty of help and support. That impression, which swung my decision to do the Engineering Doctorate, was repeatedly confirmed throughout the last four years. Dimitrios Pavlidis deserves a medal for his patience in assisting me with my first forays into computer programming. I thank David Ham for his brilliance as the architect of the code Fluidity; in particular the logical structure of the Femtools library has made my life so much easier. I hope there is a Tim Bond in every scientific computing research group: he works tirelessly to keep the good ship AMCG afloat. I also thank the numerous people have helped to debug my code and to set up automated tests of its functionality.

Professor Chris Pain, Matthew Piggott and Ian Kiltie have been superb supervisors. Chris introduced me to a lot of exciting ideas, some of which I have had the time and intelligence to investigate. Matt, always diligent, professional and thorough, has inspired me to be the same. Ian Kiltie at Rolls-Royce Submarines organised the industrial components of this project and his efforts are enormously appreciated. With team leader Simon de Haas he went to great lengths to set up my funding, security clearance and computer accounts, negotiating the labyrinthine pathways of company policies and procedures. His depth of engineering experience has been invaluable to me.

I am grateful to the University of Manchester for organising my project and EPSRC and Rolls-Royce Submarines for providing generous funding. Imperial College London is a wonderful place to do research and I would like to thank the Department of Earth Science and Engineering for looking after me so well. Simon Burbidge and his team at the Imperial high-performance computing facility have been instrumental in my research. To Simon I am also thankful for granting me an interview which provided much of the material at the end of chapter two.

My fellow PhD and EngD students have been friends and allies through

the ups and downs. The party pod reminded me that I'm not alone and that I do need two coffee and cake breaks every day. Sheldon Hall and I have developed a camaraderie from the many shared hotel rooms, beers and jaded views on EngD courses in Manchester. Patrick Farrell offered friendship and mathematical inspiration and all I had to do was cook him roast dinners. I thank him for showing me the beauty and elegance of CFD, which was somewhat obscured from my engineer's perspective of dirty details and coefficients.

My yoga teachers, Jon and Beth Monks, have helped me to find the space inside myself to take a breath and contemplate. Through them I have found union between science and spirituality: we are all seeking truth in one way or another. Without them I could not have attained the clarity of mind and detachment from ego which I believe are essential to pursue any endeavour to one's full potential.

This wonderful journey would have been so much harder and more fraught with self-doubt were it not for the love, support, humour, enthusiasm and intellectual curiosity of my girlfriend Magdalena. She has shown me afresh that what I am doing is worthwhile, that I am good enough, and that proper timetabling and planning are very important. My mother has endlessly encouraged me to pursue my interests. I have always respected my father for doing a PhD and I wanted to join him on that lofty plane. Their shared love for all things scientific set me on the path which led me here and I am eternally grateful for everything that they have done for me.

Finally, this thesis is dedicated to my little brother Nick, who died six months after I started this project. I made the choice that day to honour him by living my life with honesty, integrity and moral purpose. Out of tragedy was born my determination to make this a life worth living.

Contents

1	Introduction	6
1.1	Introduction	7
1.2	Current Challenges in CFD for Nuclear Reactor Safety Analysis	9
1.2.1	Computing Power Limitations	9
1.2.2	Range of Validity of Turbulence Models	9
1.2.3	Appropriate Resolution in CFD Solutions	10
1.3	Research Objectives	11
1.3.1	Application of Mesh Adaptivity to Industrial CFD Problems	11
1.3.2	Development of a Range of Turbulence Models	11
1.3.3	Model Validation in Industrial CFD Problems	12
1.4	Plan of Thesis	13
1.5	Statement of Original Contribution	14
2	CFD For Nuclear Safety and Design	16
2.1	Introduction	17
2.2	Pressurised Water Nuclear Reactors	17
2.2.1	Thermal Hydraulic Characteristics	20
2.3	CFD in Nuclear Engineering	22
2.3.1	Model Accuracy and Uncertainty	23
2.3.2	Towards Best-Estimate Modelling	24
2.3.3	A Brief History of Nuclear CFD	25
2.3.4	Current Challenges	27
2.3.5	Best Practice	28
2.4	High Performance Computing	37
2.4.1	Hardware	38
2.4.2	Software	40
2.4.3	Efficiency of Cluster Use	41
2.4.4	Costs	42

3	CFD Methods and Turbulence	43
3.1	Introduction	45
3.2	Turbulent Flow	46
3.2.1	Equations of Motion	46
3.2.2	Spatial and Temporal Scales	47
3.2.3	Departures from Isotropy	50
3.2.4	Turbulence in Engineering	51
3.3	Modelling The Fluid Equations	52
3.3.1	The Multiscale Problem	52
3.3.2	Reynolds Decomposition	52
3.3.3	Filtered Decomposition	54
3.3.4	Boussinesq Eddy-Viscosity Hypothesis	57
3.3.5	Levels of Approximation	58
3.4	Turbulent Flow Boundary Conditions	64
3.4.1	Near-Wall Turbulent Dynamics	64
3.4.2	Turbulent Inflow Conditions	67
3.5	Discretising The Fluid Equations	68
3.5.1	Consistency, Stability and Convergence	69
3.5.2	Finite Element Method	69
3.5.3	Time Discretisation	76
3.5.4	Finite Volume Method	78
3.5.5	Control-Volume Finite Element Method	79
3.5.6	Stabilised Finite Element Methods	80
3.6	Meshing and Optimisation	85
3.6.1	Unstructured Mesh Adaptivity	86
3.6.2	Goal-Based Adaptivity and Adjoint Methods	90
4	Weak Boundary Conditions for Turbulent Flow and Verification and Validation of the RANS $k - \varepsilon$ Model	92
4.1	Introduction	93
4.2	RANS Model Formulation	95
4.2.1	Mixing-Length Model	95
4.2.2	One-Equation Model	95
4.2.3	$k - \varepsilon$ Model	96
4.2.4	$k - \omega$ Models	98
4.2.5	Further RANS Models	99
4.3	Boundary Conditions for Turbulent Flow	99
4.3.1	Wall Shear Stress Boundary Conditions	100
4.3.2	Improved Wall Functions	101
4.3.3	Weakly Enforced Boundary Conditions	102
4.3.4	Wall Functions for the $k - \varepsilon$ Model	103

CONTENTS

4.4	Model Implementation	104
4.4.1	Discretisation	104
4.4.2	Solution Algorithm	106
4.4.3	$k - \varepsilon$ Model With Mesh Adaptivity	109
4.4.4	Scale-Adaptive $k - \varepsilon$ Model (VLES)	109
4.5	Verification	110
4.6	Validation	113
4.6.1	Simulation Setup	114
4.6.2	Results	115
4.7	Discussion	125
5	Development of An Adaptive Dynamic Tensorial Large Eddy Simulation Model	127
5.1	Introduction	129
5.1.1	Uses of LES	129
5.1.2	The Resolution Problem	130
5.1.3	Implicit and Explicit LES	131
5.1.4	Ideal LES Model Qualities	132
5.1.5	Smagorinsky LES Model	133
5.1.6	Avenues of Improvement	133
5.2	A New Tensorial Dynamic LES Model	142
5.2.1	Tensor Eddy Viscosity	142
5.2.2	Dynamic Method	143
5.2.3	Tensorial Dynamic Method	150
5.2.4	Improved Filter Width Definitions	151
5.3	A Tensorial Differential Filter For LES	152
5.3.1	Introduction to Filtering	152
5.3.2	The Inverse Helmholtz Filter	155
5.3.3	Filter Comparisons	156
5.3.4	Commutation Errors	158
5.3.5	Implementation of the Tensorial Inverse Helmholtz Filter	161
5.3.6	Discrete Filters	164
5.3.7	Verification of Filters	165
5.4	Adaptive LES Modelling	180
5.4.1	Errors in LES Solutions	180
5.4.2	Convergence of LES Solutions	181
5.4.3	Complete LES Modelling Strategies	183
5.4.4	Interpolation Error-Based Adaptivity with LES	184
5.4.5	Goal-Based Adaptivity with LES	186
5.4.6	Sub-Filter Kinetic Energy Estimation	188
5.4.7	Discussion	190

6	Validation of the Dynamic Tensorial LES Model in the 3D Backward-Facing Step	191
6.1	Introduction	193
6.1.1	Modelling Strategies	194
6.1.2	Benchmark Data	195
6.2	Simulation Setup	196
6.2.1	Geometry	196
6.2.2	Dynamic LES Parameters	196
6.2.3	Meshing and Discretisation	197
6.2.4	Boundary conditions	200
6.2.5	Diagnostics	201
6.2.6	Computing Resources	203
6.2.7	CFX Simulations	203
6.3	Results	204
6.3.1	Mesh Type and Resolution of Turbulent Structures	204
6.3.2	LES Model Diagnostics	208
6.3.3	Reattachment Length	212
6.3.4	Mean Velocity Profiles	214
6.3.5	Reynolds Stresses	221
6.3.6	Near-Wall Profiles	221
6.3.7	CFX Results	224
6.3.8	VLES Results	224
6.4	Discussion	226
6.4.1	LES and Adaptivity	229
6.4.2	Tensor Dynamic LES	229
6.4.3	VLES	230
6.4.4	Boundary Conditions	231
6.4.5	Further Work	231
7	Comparison of Fluidity and CFX in Simulations of a Vortex Diode	233
7.1	Introduction	234
7.2	Experimental Data	238
7.3	Simulation Setup	241
7.3.1	Geometry and Meshing	241
7.3.2	CFX Model Setup	242
7.3.3	Fluidity Model Setup	244
7.4	Results	246
7.4.1	Flow Dynamics: Forward	246
7.4.2	Flow Dynamics: Reverse	251
7.4.3	Pressure Drop vs. Mass Flow Rate: Forward	253

CONTENTS

7.4.4	Pressure Drop vs. Mass Flow Rate: Reverse	257
7.4.5	Diodicity	260
7.5	Discussion	262
7.5.1	Summary of Findings	262
7.5.2	Conclusions and Recommendations	263
8	Epilogue	265
8.1	Summary	266
8.2	Conclusions	268
8.2.1	Towards a Range of Robust Turbulence Models	268
8.2.2	Efficient Use of Computational Power	271
8.2.3	Estimating and Providing Appropriate Resolution in LES	272
8.2.4	Model Validation in the Vortex Diode	273
8.3	Future Work	273
8.3.1	Revised Filter Width Definitions	273
8.3.2	Towards Complete LES Modelling	274
	Bibliography	276

List of Tables

2.1	Operational states of a land-based civil nuclear reactor and their frequency of occurrence in reactor-years	21
3.1	Variation of velocity with wall distance in wall units including ‘law of the wall’	66
4.1	Order of convergence obtained in the $k - \varepsilon$ MMS verification test	112
4.2	Adaptivity parameters in 2D backward-facing step simulations	115
4.3	Reattachment length predictions in the 2D backward-facing step	116
5.1	Top-hat, Gaussian and inverse Helmholtz filter kernels, transfer functions and second moments	156
5.2	MMS test two mesh characteristics	169
6.1	Dynamic LES parameter space and experiments performed . .	197
6.2	Fixed mesh parameters in 3D backward-facing step simulations	198
6.3	Adaptivity parameters for meshes A1, A4 and A5 in 3D backward-facing step simulations	198
6.4	Time-averaged reattachment length $R_{L,av}/h$ in the 3D backward-facing step	213
7.1	CFX transient simulation solution parameters	242
7.2	Fluidity simulation solution parameters	244

List of Figures

1	Excerpt from the entry on chaos in one of the first definitive compendia of Western scientific knowledge, Chambers' Cyclopaedia of 1728	iii
2.1	PWR nuclear power plant schematic	18
2.2	Coolant flow path in a Korean PWR RPV	19
2.3	Stacking of safety margins in the determination of allowable plant conditions to prevent critical heat flux (CHF)	22
2.4	Schematic showing normal operating value of a parameter, simulated transient condition with uncertainties, acceptance criterion and actual failure point	24
2.5	Plexiglas model of ROCOM RPV	33
2.6	Exponential increase in supercomputing power in the Top500 list, 1993-2011 with projection to 2020	39
3.1	Velocity signal demonstrating superposition of scales	48
3.2	Kolmogorov's energy spectrum	49
3.3	von Kármán vortex 'street' in the wake of a cylinder	51
3.4	RANS velocity decomposition $u = \langle u \rangle + u'$	60
3.5	LES velocity decomposition $u = \bar{u} + u'$	61
3.6	U-RANS or hybrid RANS/LES velocity decomposition $u = \langle u \rangle + u'$	63
3.7	Linear shape functions in 1D	72
3.8	Piecewise constant shape function on a triangular mesh	78
3.9	2D finite volume mesh	79
3.10	2D finite element mesh and dual control volume mesh formed from the centroids and edge midpoints of the parent mesh	80
3.11	Piecewise linear and SU-modified weighting functions in 1D	83
3.12	Local element operations to improve element quality $Q_M(\Delta)$	87
3.13	Mapping of an ideal element from metric space to an anisotropic element in physical space	88

LIST OF FIGURES

3.14	Metric superposition technique for adapting to error in multiple fields, showing two metrics and the combined metric	89
4.1	Contours of the manufactured solutions in the $k - \varepsilon$ MMS test	111
4.2	2D backward-facing step geometry showing coarse mesh	113
4.3	Normalised velocity profiles at five positions downstream of the step showing coarse/med/fine/A5 mesh results, Ilinca numerical results and experimental data	118
4.4	Snapshot of velocity and eddy viscosity predicted by VLES model in 2D backward-facing step simulations	119
4.5	Snapshot of VLES model filter in 2D backward-facing step simulations	120
4.6	Portion of final adaptive meshes in 2D backward-facing step simulations showing the influence of the choice of target field .	121
4.7	Turbulent kinetic energy dissipation ε in step region of 2D backward-facing step	123
4.8	Contours of solution fields on A5 mesh comparing Fluidity to model of Ilinca and Pelletier (1997)	124
5.1	Top-hat, Gaussian and inverse Helmholtz filter kernels $G(r)$ in terms of radius r , for width $\overline{\Delta} = 0.5$	157
5.2	Plot of filter transfer functions with width $\overline{\Delta} = 0.1$	157
5.3	Effect of inverse Helmholtz filter on a signal in physical space .	166
5.4	Effect of inverse Helmholtz filter on a signal in spectral space .	167
5.5	Tracer error convergence test results comparing different filters	170
5.6	Difference between filtered and source tracer fields $(\varphi - \overline{\varphi})$ on structured and unstructured ‘B’ meshes comparing three filters	172
5.7	Tracer commutation error magnitude $ \overline{\nabla(\varphi)} - \nabla(\overline{\varphi}) $ on structured and unstructured ‘B’ meshes comparing three filters . .	173
5.8	Vector MMS test results comparing different filters	176
5.9	Difference between filtered and source vector fields $ \mathbf{u} - \overline{\mathbf{u}} $ on structured and unstructured ‘B’ meshes comparing three filters	177
5.10	Vector commutation error magnitude $ \overline{\nabla(\mathbf{u})} - \nabla(\overline{\mathbf{u}}) $ on structured and unstructured ‘B’ meshes comparing three filters . .	178
5.11	Contributions to modelling, discretisation and convergence errors in large eddy simulations	180
5.12	Speculative dependence of a functional $Q^m(\Delta)$ on mesh resolution with an asymptote Q_I^m in the inertial range $l_{DI} < \Delta < l_{EI}$	182
6.1	Sketch of flow over a backward-facing step	193

LIST OF FIGURES

6.2	Domain for the backward-facing step simulations showing cut plane A (red) and B (blue)	196
6.3	Mean velocity profile from DNS data used in 3D backward-facing step simulations	201
6.4	Mesh A1 adapting to velocity field, showing velocity magnitude	205
6.5	Mesh A4 adapting to combination of average and fluctuating velocity fields, coloured by field magnitude	205
6.6	Mesh A4 coloured by time-averaged Reynolds stress magnitude	206
6.7	First filter width $\overline{\Delta}^2$ and its gradient on plane A for dynamic LES on adaptive mesh A4	207
6.8	Velocity magnitude $ \overline{\mathbf{u}} $ on plane A for no-model and tensor Smagorinsky LES results on P3 and A1 meshes	209
6.9	$ \overline{\mathbf{u}} $ on plane A for tensor dynamic LES with isotropic filter on fixed (P3, F3) and adaptive (A1, A4, A5) meshes	210
6.10	Dynamic Smagorinsky coefficient c_S , eddy viscosity ν_T and resolved fine scales $ \overline{\mathbf{u}} - \tilde{\mathbf{u}} $ on plane A through mesh A4	211
6.11	Mean velocity profiles in 3D backward-facing step using isotropic filter on mesh A1	215
6.12	Mean velocity profiles in 3D backward-facing step using anisotropic filter on mesh A1	216
6.13	Mean velocity profiles in 3D backward-facing step using isotropic filter on adaptive meshes A1, A4 and A5 and periodic mesh P3	218
6.14	Mean velocity profiles in 3D backward-facing step using isotropic implicit/explicit first filter on mesh A1	219
6.15	Mean velocity profiles in 3D backward-facing step using anisotropic implicit/explicit first filter on mesh A1.	220
6.16	Reynolds stress profiles at $x = \frac{2}{3}R_{L,av}$ in 3D backward-facing step using isotropic and anisotropic filters on mesh A1	222
6.17	Reynolds stress profiles at $x/h = \frac{2}{3}R_{L,av}$ in 3D backward-facing step using isotropic filter on meshes P3, A1 and A4	223
6.18	Velocity profiles in wall units at $x/h = 19$ in 3D backward-facing step compared to log law, DNS and experimental data	223
6.19	Mean velocity profiles in CFX simulation of 3D backward-facing step vs. Fluidity dynamic LES with no SEM on P3 mesh, DNS and experimental data	225
6.20	Normalised Reynolds stress profiles in CFX simulation of 3D backward-facing step vs. Fluidity dynamic LES with no SEM on P3 mesh and DNS data	225

LIST OF FIGURES

6.21	VLES model velocity, eddy viscosity and damping (filter) function on plane B of 3D backward-facing step.	227
6.22	VLES model velocity magnitude $ \bar{\mathbf{u}} $ on plane A	227
6.23	Mean velocity profiles in VLES simulation of 3D backward-facing step with SEM on P3 mesh vs. DNS and experimental data	228
6.24	Normalised Reynolds stress profiles in VLES simulation of 3D backward-facing step with SEM on P3 mesh vs. DNS data	228
7.1	Vortex diode geometry	235
7.2	Streamlines in forward and reverse flow coloured by velocity in simulations using the VLES model in Fluidity	236
7.3	Side view of vortex diode experimental setup showing inlet and outlet plena and pressure tapping	239
7.4	Top view of vortex diode experimental setup showing inlet plenum, vortex chamber and pressure tapping	240
7.5	Flow dynamics in the forward direction, CFX, $k - \omega$ SST model, mass flow rate 0.43 kg s^{-1}	247
7.6	Flow dynamics in the forward direction, CFX, $k - \varepsilon$ model, mass flow rate 0.43 kg s^{-1}	247
7.7	Flow dynamics in the forward direction, Fluidity, $k - \varepsilon$ model, mass flow rate 0.29 kg s^{-1}	249
7.8	Flow dynamics in the forward direction, Fluidity, VLES and LES models, mass flow rate 0.29 kg s^{-1}	250
7.9	Flow dynamics in the reverse direction, CFX, $k - \omega$ SST model, mass flow rate 0.43 kg s^{-1}	252
7.10	Flow dynamics in the reverse direction, CFX, $k - \varepsilon$ model, mass flow rate 0.43 kg s^{-1}	253
7.11	Flow dynamics in the reverse direction, Fluidity, $k - \varepsilon$ and VLES models, mass flow rate 0.25 kg s^{-1}	254
7.12	Forward pressure drop vs. mass flow rate compared to experimental data	255
7.13	Effect of turbulence model on forward pressure drop, Fluidity, mass flow rate 0.29 kg s^{-1}	258
7.14	Reverse pressure drop vs. mass flow rate compared to experimental data	259
7.15	Diodicity compared to experimental data with $\pm 15\%$ error bars	261

Nomenclature

Latin Symbols

a_{ij}	advection matrix
B	log law additive constant
b_i	boundary vector
C	Kolmogorov constant
\mathcal{C}	commutation error
c_S, C_S	Smagorinsky coefficient
$C_{\varepsilon 1}, C_{\varepsilon 2}, C_\mu$	constants in $k - \varepsilon$ model
C_κ	von Kármán constant
c_{ij}	pressure gradient matrix
e	element
E	wall roughness
E_k	total turbulent kinetic energy
Eu	Euler number
F	VLES damping function or goal functional
f_i	force vector
g_N	Neumann boundary value
h	element size in 1D or step height
H	Hessian or filter transfer function
\mathcal{H}	Sobolev function space
i, j, k	indices in Einstein notation
\mathcal{J}	goal functional
k	turbulent kinetic energy
k_{ij}	diffusion matrix
l	characteristic lengthscale in RANS models
l_m	mixing lengthscale
l_0	integral lengthscale
\mathcal{L}^2	space of square-integrable functions
m	polynomial degree or index
\mathcal{M}	metric tensor

LIST OF FIGURES

M	mass matrix
M_{Helm}^e	discrete inverse Helmholtz operator
M_L	lumped mass matrix
M_n	n th moment
m_{ij}	elemental mass matrix
n	number of entities or VLES model parameter
\mathbf{n}	normal
\mathcal{O}	order
N_i	shape function
p	pressure
P	polynomial order of shape function
\tilde{p}	modified pressure
\mathcal{P}	function space
Pe	Peclet number
Pr	Prandtl number
Q	matrix of eigenvectors
Q_M	quality functional
R_L	reattachment length
Re	Reynolds number
S_{ij}	strain rate tensor
$ S $	magnitude of strain rate tensor
Str	Strouhal number
t	time
T	time period
\mathbf{T}	kinetic energy flux
t_0	characteristic timescale
\mathbf{u}	velocity vector
$\langle \mathbf{u} \rangle$	time-averaged velocity
$\bar{\mathbf{u}}$	filtered velocity
$\tilde{\mathbf{u}}$	test-filtered velocity
\mathbf{u}'	fluctuating velocity or Reynolds stresses
\mathbf{u}^{**}	most recent value of velocity
\mathcal{U}	function space
u_0	characteristic velocity
u_τ	friction velocity
u^*	nominal velocity
v	test function
V	volume
w	test function
\mathcal{W}	function space
\mathbf{x}	position
y	wall distance
y^+	non-dimensional wall distance

LIST OF FIGURES

Greek Symbols

α	filter width ratio
β	filter width ratio or VLES model parameter
γ	small number is convergence proofs
Γ	computational domain boundary
δ	infinitesimal increment
δ_{ij}	Kronecker delta
Δ	small increment
ε	dissipation rate of k or Helmholtz filter parameter
ε_u	interpolation error
ζ	local coordinate
η	Kolmogorov lengthscale or local coordinate
θ	timestepping implicitness parameter
θ_{nl}	nonlinear advective relaxation parameter
κ	wavenumber
λ_e	interpolation parameter
λ_i	eigenvalue
Λ	matrix of eigenvalues
ω	turbulence frequency (k/ε)
μ	dynamic molecular viscosity
μ_T	dynamic eddy viscosity
ν	kinematic molecular viscosity
ν_T	kinematic eddy viscosity
$\nu_{T,ij}$	kinematic tensorial eddy viscosity
$\bar{\nu}$	truncation error in CG discretisation
Π	production term
ξ	local coordinate
ρ	fluid density
σ_k	Prandtl number in k equation
σ_ε	Prandtl number in ε equation
τ	stress tensor
τ_{ij}	Reynolds or SFS stress tensor
τ_{ij}^R	deviatoric part of Reynolds or SFS stress tensor
τ_w	wall shear stress
φ	scalar
Ω	computational domain
Ω_{ij}	rotation rate tensor

LIST OF FIGURES

Acronyms

AMCG	Applied Modelling and Computation Group, Imperial College London
CFD	computational fluid dynamics
CFL	Courant-Friedrichs-Lax condition
CG	continuous Galerkin
CHF	critical heat flux
CPU	central processing unit
CV	control volume
CV-FE	control volume-based finite element method
DES	detached eddy simulation
DG	discontinuous Galerkin
DNB	departure from nucleate boiling
DNS	direct numerical simulation
DoE	design of experiments
FD	finite difference
FE	finite element
flops	floating point operations per second
FV	finite volume
GPU	graphical processing unit
HPC	high-performance computing
IAEA	International Atomic Energy Agency
ILES	implicit large eddy simulation
LBB	Ladyzhenskaya-Babuska-Brezzi stability condition
LES	large eddy simulation
LOCA	loss of coolant accident
MILES	monotonically integrated large eddy simulation
MMS	method of manufactured solutions
NEA	Nuclear Energy Agency
NRS	nuclear reactor safety
PDE	partial differential equation
PDF	probability density function
PWR	pressurised water reactor
RANS	Reynolds-averaged Navier Stokes
RPV	reactor pressure vessel
RR	Rolls-Royce Submarines
RSM	Reynolds stress model
SEM	synthetic eddy method
SFS	sub-filter scales
SGS	sub-grid scales
SST	shear stress transport model
SU	streamline upwind
SUPG	streamline upwind Petrov-Galerkin

LIST OF FIGURES

TKE	turbulent kinetic energy
U-RANS	unsteady Reynolds-averaged Navier Stokes
VLES	very large eddy simulation
VMS	variational multiscale
V & V	verification and validation

Mathematical Symbols

Δ	filter width
$\overline{\Delta}, \overline{\Delta}_{ij}$	tensor filter width
∇	grad
\cdot	dot product
\circ	Hadamard (pointwise) product
$\overline{(\dots)}$	filtered quantity
(\dots)	filtered quantity
$\langle(\dots)\rangle$	time-averaged quantity
$ (\dots) $	magnitude
$\ (\dots)\ _{\mathcal{L}^2}$	\mathcal{L}^2 norm
\in	in the set of...
\cup	union of...
\forall	for all...
\setminus	minus the set of...

1 Introduction

This brief introductory chapter explains the motivation for the research. Current challenges in computational fluid dynamics for nuclear reactor safety and design are reviewed. The aims and original contribution to science of this research are stated and the plan of the thesis is presented.

Contents

1.1	Introduction	7
1.2	Current Challenges in CFD for Nuclear Reactor Safety Analysis	9
1.2.1	Computing Power Limitations	9
1.2.2	Range of Validity of Turbulence Models	9
1.2.3	Appropriate Resolution in CFD Solutions	10
1.3	Research Objectives	11
1.3.1	Application of Mesh Adaptivity to Industrial CFD Problems	11
1.3.2	Development of a Range of Turbulence Models	11
1.3.3	Model Validation in Industrial CFD Problems	12
1.4	Plan of Thesis	13
1.5	Statement of Original Contribution	14

1.1 Introduction

Nuclear power has been providing clean, plentiful electricity for over 50 years. In almost 15,000 cumulative reactor-years of operation, only three major accidents have occurred: Three Mile Island, Chernobyl and Fukushima (WNA, 2012a). While these accidents have fuelled public doubts over the safety of nuclear energy, they have also fostered a worldwide culture of safety in the nuclear industry. Several countries have operated nuclear-powered navies for a similar period of time. In the United Kingdom, Rolls-Royce plc has been designing, building and servicing submarine nuclear propulsion systems for the Royal Navy since the 1950s with an impeccable safety record (Rolls-Royce, 2012). Stringent national and international safety regulations coupled with a comprehensive set of guiding principles governing nuclear reactor design and operation ensure that modern civil and naval nuclear power plants are safer than ever before (WNA, 2012a).

A central tenet of the engineering principles is the provision of *defence in depth*: the design of the reactor system ensures that reasonably foreseeable “abnormal operation and failures” have an extremely small probability of occurring, but if they do, additional safety features are in place to control, contain and mitigate the consequences in order to protect the public (HSE,

2006). The potential consequences of a fault must be analysed, from the initiating fault through the ensuing sequence of events to its safe (or otherwise) conclusion. Therefore, we require a physical or computational *model*.

Building life-sized realistic physical models for severe accident analyses in the laboratory, such as core meltdown in the Chernobyl disaster, is very difficult and expensive (Sehgal, 2006). Furthermore, scaled-down or simplified physical models may not recreate the exact conditions of the accident. Computational or numerical models have been used in a wide variety of nuclear accident analyses for over 30 years. Initially, they were coupled systems of one-dimensional (1D) approximations of individual components, incorporating complex parameterisations derived from a huge database of isolated-effect experiments (NEA, 2008).

However, many physical phenomena, in particular fluid flows, simply cannot be represented as 1D numerical models with an acceptable level of accuracy. Prime examples are turbulent coolant flow in the reactor pressure vessel (RPV) (Höhne et al., 2006), buoyancy-driven flow in the containment (Scheuerer et al., 2005b), boiling and multiphase flow (Yadigaroglu, 2005), core reflooding after dryout (Smith, 2010) and thermal fatigue in pipe junctions (NEA, 2008). In these situations and many more, computational fluid dynamics (CFD) methods based on 3D and transient numerical models is invaluable (NEA, 2008).

A limiting factor on the adoption of CFD for nuclear reactor safety (NRS) analysis is the relative scarcity of good quality 3D data from realistic experiments, particularly of transient multiphase flows, for code validation purposes. In comparison, the large amount of amassed knowledge from single-effect tests makes the 1D codes quite reliable in these cases, as long as they are only used within the range of validity of the correlations (Smith, 2010). Major coordinated international efforts are under way to remedy this situation, detailed in NEA (2008) and Smith (2010).

Whilst CFD may be in the early stages of use for such complex cases, it is quite well established for simulating single-phase flow in the nuclear industry (Bestion et al., 2004). Examples of accidents where single-phase CFD can be successfully applied include boron dilution transients, mixing of hot and cold coolant at a pipe junction and hydrogen combustion in the containment (Bestion et al., 2004). Nevertheless, there are some fundamental questions to be answered on the trustworthiness of CFD methods. These pertain to the range of applicability of numerical turbulence models, the requirement for massive computing power and the quantification of uncertainty in numerical models (NEA, 2008).

1.2 Current Challenges in CFD for Nuclear Reactor Safety Analysis

1.2.1 Computing Power Limitations

The computational demands for simulations of typical NRS cases can be daunting for two reasons. Firstly, quantifying uncertainty in the solution dictates that a large number of separate simulations are run (NEA, 2007a). Secondly, the computational demands of an individual simulation can be large owing to geometrical complexity and the length of time for accident scenarios to evolve (Smith, 2010). For example, LES of an 800-second transient in a whole RPV required 40 days on a 2.5 million node mesh split over 32 processors (Bieder et al., 2007). The cost of computational power is constantly coming down but as more computation resources become available, more demanding simulations are run to fill it: the demand for more power is never-ending (Smith, 2010). Therefore, it is vital to maximise the efficiency with which the simulation is run, e.g. only to use the resolution necessary to achieve sufficient accuracy in the solution.

1.2.2 Range of Validity of Turbulence Models

Because of the complexity of NRS cases, usually the only practical CFD approach is steady-state Reynolds-Averaged Navier-Stokes (RANS) modelling, but the simpler models are not applicable to a range of commonly occurring flows, being unable to account for phenomena such as buoyancy, swirling flows and transience. RANS models tend to be numerically stable on sub-optimal meshes, giving a rough estimate of the correct answer in most flow conditions. More complex RANS models such as Reynolds stress models may be able to account for the aforementioned phenomena, but are generally less numerically stable.

Models that account for limited large-scale transience such as unsteady RANS (U-RANS) or very large eddy simulation (VLES) may be more appropriate, but there is not yet a consensus on how to define the separation between resolved and modelled dynamics in a physically meaningful way (Travin et al., 2004). Similar problems face the many attempts to blend RANS and LES, such as detached eddy simulation (DES), because the two models are fundamentally different (Friess and Manceau, 2012).

In NRS cases where turbulence has to be well-resolved, such as mixing in a pipe junction, LES is the best option, e.g. Kuczaj et al. (2010). The commonly-used Smagorinsky model has several shortcomings, for example its inability to account for anisotropy of the sub-filter scales (SFS) (Pope,

2000). In this model, the assumption is made that the SFS are isotropic by citing the Kolmogorov local isotropy hypothesis, allowing for a very simple form based on a scalar eddy viscosity. The mesh resolution requirements to satisfy this assumption can be prohibitively fine (Pope, 2000). It is also not trivial to determine whether the solution has been sufficiently well-resolved (Meyers et al., 2003). There is evidence to suggest that the SFS in real flows are actually anisotropic (Fureby and Grinstein, 2002) and that an ‘anisotropy cascade’ from large to small scales exists (Sagaut, 2006). To improve model accuracy and practicality, we need a way to account for this in the LES model.

The range of different NRS cases means that there is not one ideal model for all of them:

*“Don’t use a pair of scissors to cut the grass on a football field.
On the other hand, don’t cut your hair with a lawn mower” (NEA,
2007a).*

Different levels of accuracy are required for different purposes: for design optimisation, a relatively simple, computationally undemanding model might be sufficient, while quantitative prediction of a safety-critical parameter demands the best affordable model (Rolls-Royce, 2010). A hierarchy of increasingly detailed but equally valid approximations to the true flow are needed.

1.2.3 Appropriate Resolution in CFD Solutions

Best-practice guidance on uncertainty quantification recommends the use of mesh-refinement studies to demonstrate mesh-independence of a CFD solution, i.e. that further refinement does not lead to significant accuracy improvements (NEA, 2007a). However, in typical NRS cases the mesh resolution is so demanding that mesh-independent solutions cannot be obtained or cannot be demonstrated (Menter and Hemstrom, 2002). Methods for demonstrating convergence of CFD solutions without resorting to mesh-refinement studies are required.

Furthermore, LES is sensitive to the mesh and does not converge to a mesh-independent instantaneous solution in the way that RANS does (Klein, 2005). The interaction of modelling and discretisation errors in LES results in unpredictable convergence behaviour (Meyers et al., 2003). Functions of the solution such as time averages or surface integrals may be better metrics of solution accuracy. Selection of appropriate functions of the solution by which to measure the accuracy of an LES is a subject of considerable importance for NRS. A related area is the determination of whether sufficient resolution has been provided to satisfy modelling assumptions, hence deem the LES well

resolved and provide confidence that the result is accurate. Both of these topics need to be investigated if LES is to become a trusted tool for NRS.

1.3 Research Objectives

In this thesis, steps toward the development of a hierarchy of robust turbulence models with mesh adaptivity suitable for nuclear reactor safety analysis are described. The objectives of the research are now presented.

1.3.1 Application of Mesh Adaptivity to Industrial CFD Problems

In order to resolve the important flow features without wasting resolution on regions of the flow where nothing is happening, the mesh can be refined or coarsened locally. The question is, where are the regions of interest? In general, we do not know in advance and the more complex the flow is, the harder it is to find out (Fröhlich et al., 1998). Indeed, it may vary in time. This is where unstructured mesh adaptivity is invaluable: the mesh is automatically refined or coarsened independently in all directions to meet a specified solution error criterion with a minimal number of mesh nodes (Pain et al., 2001). Use of adaptive meshing algorithms for NRS was suggested by Scheuerer et al. (2005a) in order to bring the most computationally demanding cases within reach. Unstructured mesh adaptivity has been implemented in Fluidity (Pain et al., 2001) and its potential for optimising mesh resolution and reducing computational work in turbulent flows is investigated in this project.

1.3.2 Development of a Range of Turbulence Models

A hierarchy of robust turbulence models at various levels of approximation to real turbulence is needed to suit different problem types. Near-wall modelling methods need to be developed for arbitrary surface geometry and mesh topology, which allow for coarse near-wall resolution in order to reduce computational effort. The open-source finite element code Fluidity, developed by the Applied Modelling and Computation Group (AMCG) at Imperial College London, is to be used as a test bed for the models developed.

Unstructured mesh adaptivity can be used to provide optimal resolution with any turbulence model. Strategies for combining mesh adaptivity with RANS, U-RANS/VLES and LES models need to be developed and tested. A RANS model is not mesh-sensitive, so the aim is simply to provide sufficient

resolution to control a discretisation error in the solution. However, the most suitable definition of that error may be problem-dependent and needs to be investigated. With mesh-sensitive models such as VLES and LES, the mesh should resolve the major flow structures while the turbulence model tailors the dissipation to an appropriate level for the local resolution.

LES on anisotropic inhomogeneous unstructured meshes poses particular challenges. Differing amounts of dissipation should be applied in different directions to take mesh anisotropy into account (Gallerano et al., 2005). One way to do this is with a tensorial eddy viscosity (Carati and Cabot, 1996). Filter width can also be defined as a tensor which is more appropriate than the usual scalar quantity on highly stretched elements; Bentham (2003) developed a tensorial definition of filter width for anisotropic tetrahedral meshes. However, inhomogeneous filter width causes commutation errors which may pollute the solution (Reynolds, 1990). These errors need to be quantified and, if possible, reduced. Inhomogeneity of the SFS on an adaptive mesh can be accounted for by applying the dynamic method of Germano et al. (1991) to a particular LES model. The resulting dynamic LES model is more robust as it can be used in a wide range of flows without tuning or modification. As yet, no attempts have been made to combine all of these aspects into a single LES model.

The last piece of this puzzle is to determine sufficient resolution for LES. Strategies and techniques for ensuring that the important flow dynamics are resolved are needed. Errors have to be quantified and controlled in order to provide confidence in the solution. Advanced techniques such as goal-based adaptivity will be considered.

1.3.3 Model Validation in Industrial CFD Problems

Validation of turbulence models in standard academic test cases is necessary to test their robustness and accuracy. It is important to show that equivalent results can be achieved at lower computational cost by using turbulence models in combination with mesh adaptivity. Models should also be validated against benchmark data in a flow of relevance to nuclear engineering. Additionally, code-to-code benchmarking is recommended to give extra confidence in the results and understand model behaviours. To this end, comparisons of Fluidity against the commercial code CFX (ANSYS, 2011) are performed in the vortex diode test case.

1.4 Plan of Thesis

Chapter 2 examines the role of CFD in nuclear engineering to provide context for the rest of the work. The thermal-hydraulic characteristics of flow in an RPV are described. A brief history of the uses of CFD in nuclear design and safety analysis is presented with industry guidelines on CFD best-practice. Current trends towards best-estimate modelling are explained, giving further motivation to this research into high-fidelity CFD methods. The chapter finishes with a breakdown of cost estimates and practicalities of running a high-performance computing facility for very large parallelised CFD studies.

In order to appreciate the development of the models in this thesis, some fundamental concepts require explanation. Chapter 3 presents an introduction to the physics of turbulence and the relevance of turbulent flow in engineering applications. The Reynolds and filtered decompositions of the Navier-Stokes equations and associated modelling strategies are explained. Boundary conditions for turbulent flow are presented next including the synthetic eddy method (SEM) of Jarrin *et al.* (2006) for inflow turbulence generation which proves crucial to correct simulation of the 3D backward-facing step. Next, the spatial/temporal discretisation methods used in this thesis are described, plus alternative numerical stabilisation methods which are compared to the physically-derived turbulence models developed. Finally, unstructured mesh adaptivity is described, both interpolation error-based and goal-based. The metric tensor used in the tensorial LES model is explained.

Chapter 4 moves on to develop adaptive RANS modelling methods for use in industrial CFD. The standard $k - \varepsilon$ model is discretised by the control volume-based finite-element (CV-FE) method. In addition, a log-law wall function in variational form is developed and a VLES model is implemented. Verification of the $k - \varepsilon$ model by the Method of Manufactured Solutions (MMS) is performed, confirming that the model has been correctly coded. Validation of the model against experimental data is performed using the 2D backward-facing step at $Re = 132,000$ with fixed and adaptive meshes.

In Chapter 5, the development of a robust new tensorial dynamic LES model, designed for use on adaptive meshes, is described. Deficiencies in the most popular LES model in industry, the Smagorinsky model, are laid out and improvements are discussed. These lead to the proposition of the new model. The implementation of this model follows, including a new tensorial inverse Helmholtz filter operator. Numerical properties of the filter are verified by MMS. Possible strategies for using the LES model with adaptivity are described, including a new combination of interpolation error-based and goal-based adaptivity designed to ensure that the LES is well resolved.

Future research directions in goal-based adaptivity for industrial CFD are signposted.

Chapter 6 presents results of the new tensorial dynamic LES model on fixed and adaptive meshes in the 3D backward-facing step at $Re = 5100$. The effect of several dynamic LES model settings and mesh adaptivity parameters on the accuracy of multiple dependent variables is investigated. Results are compared to the DNS data of [Le et al. \(1997\)](#) and a comparable dynamic LES simulation of [Panjwani et al. \(2009\)](#). VLES and CFX dynamic LES results are also presented.

In Chapter 7 a challenging industrial CFD problem at Rolls-Royce Nuclear, the vortex diode, is simulated using a variety of turbulence models in Fluidity and CFX. The device generates markedly different pressure drops depending on which way water flows through it, with a vortex being generated in one direction only. Pressure drops across the device are known from experimental data.

Finally, Chapter 8 sums up the findings of the research, draws conclusions from the findings and presents some promising future research directions in LES on adaptive meshes and goal-based adaptivity.

1.5 Statement of Original Contribution

The novel combination of the RANS $k-\varepsilon$ turbulence model discretised by the control volume-based finite element (CV-FE) method with law-of-the-wall boundary conditions in variational form and unstructured mesh adaptivity is developed. The model performs well in simulations of the 2D backward-facing step on fixed meshes. Using the model with mesh adaptivity reduces the number of nodes in the mesh by 50-80% compared to a fixed mesh with comparable reattachment length predictions. Mesh independence is demonstrated by reaching identical solutions using five different error measures to generate adaptive meshes. A modified scale-adaptive form of the $k-\varepsilon$ model, the VLES model of [Han and Krajnović \(2012\)](#) is implemented and initial results are promising.

The second significant contribution of this thesis is the novel combination of a the the dynamic method with the tensorial eddy-viscosity LES model based on the tensorial filter width of [Bentham \(2003\)](#) suitable for unstructured meshes. The method is ideally suited for use with unstructured mesh adaptivity, dynamically adjusting the amount of dissipation to suit the local resolution in each direction independently. This combination of the tensorial dynamic LES model with unstructured mesh adaptivity is proposed as a highly efficient and robust CFD model for diverse industrial flows. In

most flows encountered in industry, turbulence departs significantly from the isotropy and homogeneity assumptions which form the basis of commonly used LES models. A model which can capture both the anisotropy and inhomogeneity of real turbulence is expected to have a distinct advantage in these situations.

Test filtering in the dynamic method is achieved by an explicit tensorial inverse Helmholtz filter operator. Commutation error properties of the filter are verified and shown to be reduced by controlling the spatial variation of filter width via the mesh adaptivity settings. By using a tensorial eddy viscosity and filter width, the SFS are anisotropic by construction and the mesh requirements may be relaxed compared to isotropic models. Tensorial eddy viscosity also addresses a well-known problem with the eddy-viscosity hypothesis, namely that it incorrectly assumes that SFS stress and resolved strain are aligned (Meneveau and Katz, 2000). Stress-strain alignment in the new method is determined by anisotropy of the filter width and hence by the mesh anisotropy.

Validation in the 3D backward-facing step on coarse meshes by LES standards shows that it is indeed an efficient method for resolving the important scales of motion, equalling the predictive accuracy but using 60% fewer nodes than the same model on a fixed mesh. Reattachment length predictions are excellent, the best result falling within 4% of the DNS benchmark. Higher-order statistics are captured with satisfactory accuracy and show an improvement over the standard Smagorinsky model.

Finally, a mixed error-based/goal-based adaptivity method is proposed. Goal-based adaptivity adapts the mesh to minimise the error in a goal functional. Here the functional is an LES ‘resolution sensor’ defined as the proportion of resolved to total kinetic energy and the goal is to resolve 80% of the total energy. In this mixed method, the unstructured anisotropic mesh generated by conventional interpolation error-based adaptivity is rescaled to meet the resolution goal as well as satisfy the interpolation error bound. Provided that the measures of actual and desired resolution are appropriate, this method could be used to ensure that modelling assumptions have been satisfied without resorting to the conventional laborious trial-and-error approach to mesh design. Implementation and testing of the method, as well as investigation of alternative resolution sensors, are the subject of future research.

2 CFD For Nuclear Safety and Design

This chapter describes the thermal-hydraulic characteristics of a single-phase pressurised water nuclear reactor (PWR) and the safety principles governing its design. A brief history of computational fluid dynamics (CFD) in nuclear reactor safety and performance analysis is presented, along with current trends towards best-estimate modelling and best practice guidelines, pointing towards the methods described in later chapters. The chapter closes with a breakdown of the costs of high-performance computing (HPC) for running very large parallelised CFD models.

Contents

2.1	Introduction	17
2.2	Pressurised Water Nuclear Reactors	17
2.2.1	Thermal Hydraulic Characteristics	20
2.3	CFD in Nuclear Engineering	22
2.3.1	Model Accuracy and Uncertainty	23
2.3.2	Towards Best-Estimate Modelling	24
2.3.3	A Brief History of Nuclear CFD	25
2.3.4	Current Challenges	27
2.3.5	Best Practice	28
2.4	High Performance Computing	37
2.4.1	Hardware	38
2.4.2	Software	40
2.4.3	Efficiency of Cluster Use	41
2.4.4	Costs	42

2.1 Introduction

Computational fluid dynamics play an increasingly important role in the engineering design process, particularly in the early stages where basic design parameters are being established. Experimental testing is very expensive and engineers wish to avoid it during these early design iterations. Use of CFD on supercomputers is relatively commonplace now and it is typical to simulate flows with tens or hundreds of millions of degrees of freedom using tens of thousands of processors (Fischer et al., 2008; Chabard and Laurence, 2009; Sahni et al., 2009). CFD is now being applied to the very complex turbulent flows found in nuclear reactors for the purposes of safety analysis and design optimisation. Very high levels of confidence are required in the results if they are to be used to make safety-related decisions.

2.2 Pressurised Water Nuclear Reactors

The type of nuclear reactor to which this research is relevant is the single-phase pressurised water reactor (PWR). Currently, there are 325 civil PWR

2.2 Pressurised Water Nuclear Reactors

nuclear power stations worldwide which are operational or under construction (WNA, 2012b), and PWRs have been used for decades to provide propulsion for Royal Navy submarines (Rolls-Royce, 2012) as well as the surface and submarine fleets of several other countries.

Figure 2.1 shows a schematic of the main components of a land-based nuclear power plant and the primary and secondary coolant flows (indicated by arrows). Figure 2.2 shows representative core barrel and coolant flow path in more detail. Fissile material (uranium) is contained in arrays of fuel rods inside the core barrel, itself inside a thick steel reactor pressure vessel (RPV). Neutrons released by the decay of an atom of the fissile material knock neutrons out of other atoms, causing a chain reaction, or fission reaction. In the process the atoms split into pieces, releasing a large amount of energy as heat. Control rods of neutron-absorbing material are inserted into the core barrel to maintain the reaction at a constant energy output.

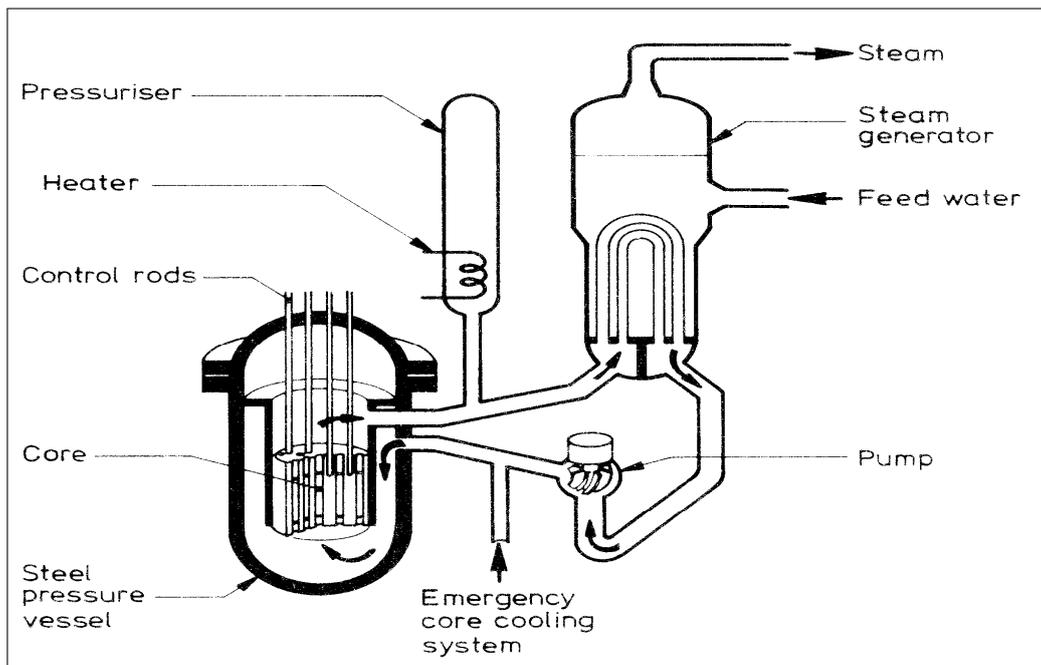


Figure 2.1: PWR nuclear power plant schematic showing flow path and primary components including pressure vessel, core barrel, hot and cold legs, steam generator, pump and pressuriser (Hewitt, 2010).

Liquid water at around 275°C and 150 bar pressure is pumped into the RPV via the ‘cold leg’, down the annular passage or *downcomer* between the RPV and core barrel and into the lower plenum. The lower plenum acts to homogenise the coolant flow into the core, reducing large-scale turbulent

2.2 Pressurised Water Nuclear Reactors

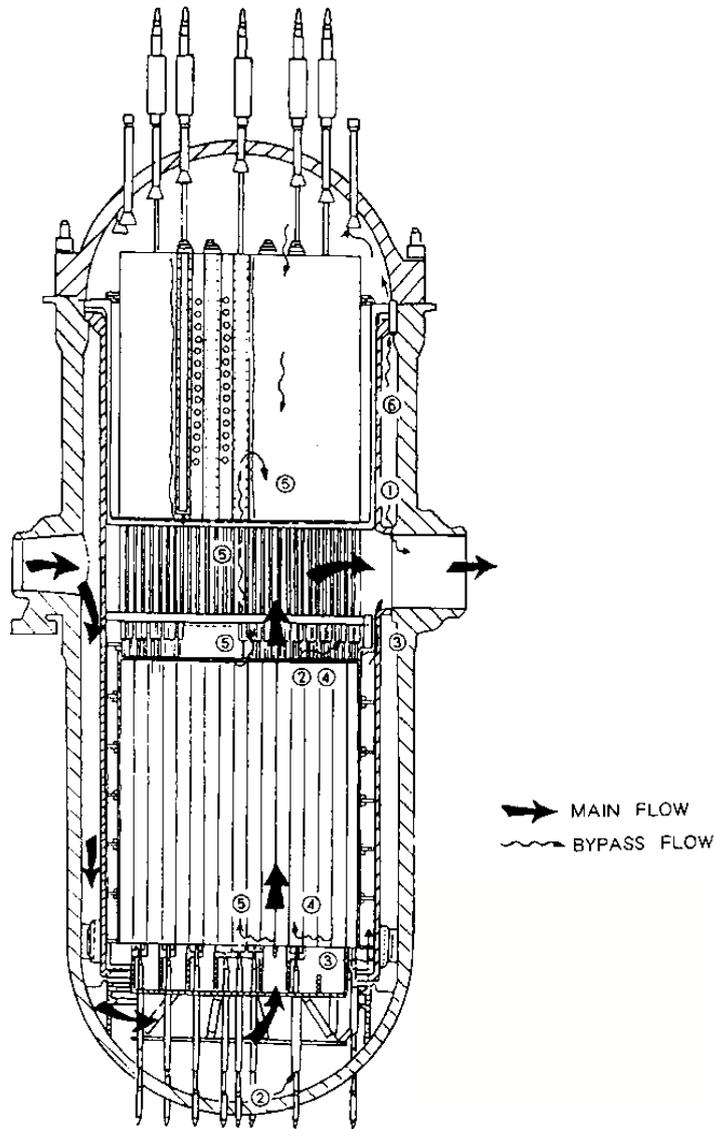


Figure 2.2: Detailed schematic of RPV internals featuring core barrel, annular downcomer, bottom plenum and cold/hot nozzles in a Korean PWR RPV (KAERI, 2000). Coolant flow path is also shown including a bypass flow which cools the RPV upper dome.

fluctuations. Obtaining approximately uniform flow across all channels is essential to ensure removal of heat and prevent local hot spots which could lead to an accident. The coolant then flows up through channels in the core barrel to cool the fuel assemblies by forced convective heat transfer as shown in Figure 2.2. A secondary ‘bypass flow’ is also shown. It does not cool the fuel but instead circulates around the upper parts of the RPV to provide cooling there.

Above the core the coolant, now close to saturation conditions at around 315°C, continues out of the RPV in the ‘hot leg’ and onwards to a heat exchanger where the energy of the fission reaction is transferred to a secondary loop. The secondary loop contains water at lower pressure which boils inside the steam generator, the steam being used to drive a turbine and generate electricity. The cycle is closed by condensing and cooling the primary and secondary loop coolants and pumping them back around the circuits.

Additional components shown in Figure 2.1 are the pressuriser, which maintains primary loop pressure, and the emergency core cooling system, which floods the core in the event of a loss-of-coolant accident (LOCA). ‘Large break’ and ‘small break’ LOCA events are caused by ruptures in the main coolant loop or subsidiary pipework respectively. The entire apparatus is housed within a large building-cum-pressure vessel called the containment. In the event of radioactive material escaping the core and RPV, the containment is designed as the next major barrier. These major accidents are prime subjects for CFD analysis.

2.2.1 Thermal Hydraulic Characteristics

Table 2.1 categorises the operational states of a PWR. Under normal operating conditions the reactor is designed to have a *negative feedback coefficient*: it automatically suppresses any intensification of the fission reaction by various feedback mechanisms, such as thermal expansion (Todreas and Kazimi, 1990). Other more exotic quantum effects are also involved in the coefficient, such as the reduction of a uranium atom’s absorption cross-section (the probability of absorbing a neutron in a collision) at higher temperatures (Hewitt, 2010). The overall effect is to maintain the fission reaction, and therefore power output, at a stable state.

An event which can disrupt this stability is boiling of the primary coolant inside the core¹. If the coolant begins to boil and surpasses a ‘point of no return’ called *departure from nucleate boiling* (DNB), its convective heat

¹Boiling Water Reactors (BWRs) allow the coolant to boil in the core, but this discussion pertains only to single-phase PWRs.

2.2 Pressurised Water Nuclear Reactors

Operational state	Examples	Frequency (per reactor-year)
Normal operation	operation at full power	continuous
Operational transients	startup and shutdown	1×10^1
Upset conditions	unexpected faults	1×10^0
Emergency conditions	break in small pipe	1×10^{-2}
Design basis accidents	loss of coolant accident (LOCA)	1×10^{-4}
Beyond design basis	earthquake/tsunami	1×10^{-6}

Table 2.1: Operational states of a land-based civil nuclear reactor and their frequency of occurrence in reactor-years (Todreas and Kazimi, 1990).

transfer coefficient drops, causing an abrupt rise in the fuel surface temperature. This is one way in which the *critical heat flux* (CHF) condition in the fuel channel can be reached. The engineer’s task is to prevent the system reaching CHF under all possible operating conditions, collectively called the *design basis conditions*, incorporating rows one to five in Table 2.1. The conditions in which CHF can occur are supposed to be ‘beyond design basis’: the likelihood of their occurrence is negligibly small.

In normal operation, the CHF limit is never reached because the steady-state coolant conditions are sufficiently far from the limit. However, under abnormal operating conditions the coolant may be closer to the limiting conditions. Safety margins are added to prevent the CHF limit from being reached under all design basis conditions.

Figure 2.3 gives an idea of the applied safety margins and their sources. It is a conservative picture, assuming that all the worst conditions happen at the same time and place. For example, the highest heat flux is assumed to occur in the channel with the smallest flow diameter that manufacturing tolerances allow. Under normal conditions, the average state of the coolant is called the nominal steady-state average condition (bottom line in Figure 2.3). Variations across the width and height of the core are added, giving the nominal steady-state hot-spot condition. Analysis of engineering uncertainties, for example coolant channel width variations due to manufacturing tolerances, are added on top of this to arrive at the maximum steady-state hot-spot condition. An ‘overpower’ factor, taking into account any allowable transient events, e.g. during reactor shutdown, is added on top to arrive at the limit for allowable transient conditions. Finally, any uncertainties in measurements of flow conditions are added on top to arrive at the CHF limit.

Turbulence is a prime example of a variation about the mean flow conditions. We need to understand turbulence in order to calculate the overpower

factor in Figure 2.3. Turbulent fluctuations arise from a variety of sources affecting the initial and boundary conditions of the coolant flow. For example, the flow from the main coolant pump is highly swirling due to the pump characteristics and twists in the pipework, generating turbulence which is transported into the RPV and must be quelled to acceptable levels before entering the core.

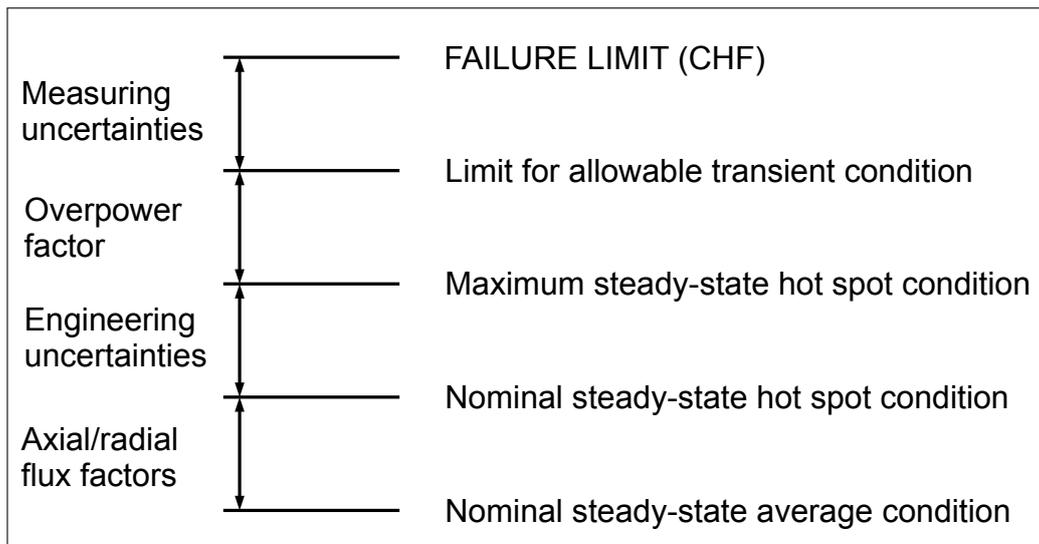


Figure 2.3: Stacking of safety margins in the determination of allowable plant conditions to prevent critical heat flux (CHF). The conservative assumption is made that worst-case values can occur simultaneously, hence the total safety margin is the sum of individual maximum uncertainty values.

2.3 CFD in Nuclear Engineering

There are many existing and potential areas of application of CFD in nuclear reactor safety and performance analysis. The types of physics present include single- and two-phase flows, solid-fluid interactions and fluid-neutron interactions. Problems range from the whole-system level, to individual components, right down to the micro-scale (for example, boiling at microscopic nucleation sites). Strict regulations and checks on the safety of nuclear plant demand close attention to the accuracy of computational predictions and uncertainties therein. CFD usage and best practice are comprehensively covered by technical reports produced by the Nuclear Energy Agency (NEA) (NEA, 2007a,b, 2008) and the International Atomic Energy Agency (IAEA) (IAEA,

2003, 2009). These and similar guidelines produced by Rolls-Royce Submarines (Pierce and Stevens, 1975; Rolls-Royce, 2010; Kiltie, 2011) have been frequently cited in this chapter. In the following discussion, current trends towards best-estimate computational modelling are outlined, before presenting a brief history of CFD modelling of various nuclear thermal-hydraulic test cases.

2.3.1 Model Accuracy and Uncertainty

The concepts of *conservative* and *best-estimate* methods are applied to both computational models and any assumptions made about input data to those models. A conservative computational model is one which simplifies the physics of the problem to such an extent that its predictions are unavoidably conservative. An example is the assumption of laminar flow in a coolant channel, which would lead to under-prediction of heat transfer by a significant margin, and therefore a conservative estimate of DNB. ‘Best-estimate’ computational models such as CFD codes provide a semi-realistic model of the true physics, for example by resolving 3D flows.

A best-estimate assumption about input data would be that there is a 95% confidence level in a parameter value being in a certain range. Confidence levels can also be derived for combinations of values by sophisticated methods of combining uncertainties. Uncertainties come from many sources, either due to inherent randomness such as variations in component dimensions due to manufacturing tolerances (*aleatory* uncertainty), or due to imperfect knowledge, estimation or modelling of the physics (*epistemic* uncertainty) (Todreas and Kazimi, 1990).

Conservative assumptions about model input data would be that the worst possible parameter values occur simultaneously. There is a negligibly small chance of a worse situation arising than this. Therefore, it is not necessary to apply confidence levels to the data. Uncertainties still have to be calculated in order to determine what the worst case is, but they are combined in a conservative way. An illustrative example of conservative ‘stacking’ of uncertainties and safety margins was given above.

These concepts are compared in Figure 2.4. It illustrates best-estimate and conservative calculations of a transient condition where the value of a parameter changes from its normal operating point. The conservative acceptance criterion is arrived at by stacking safety margins, while the conservative model incorporates worst-case input data. The best-estimate model aims for a realistic prediction, hence confidence bands have to be applied. Even the upper limit of the band may be lower than the conservative prediction, giving a bigger safety margin or allowing the normal operating point to be raised.

2.3 CFD in Nuclear Engineering

Use of a more accurate model may give us a better idea of what happens than a simplistic model.

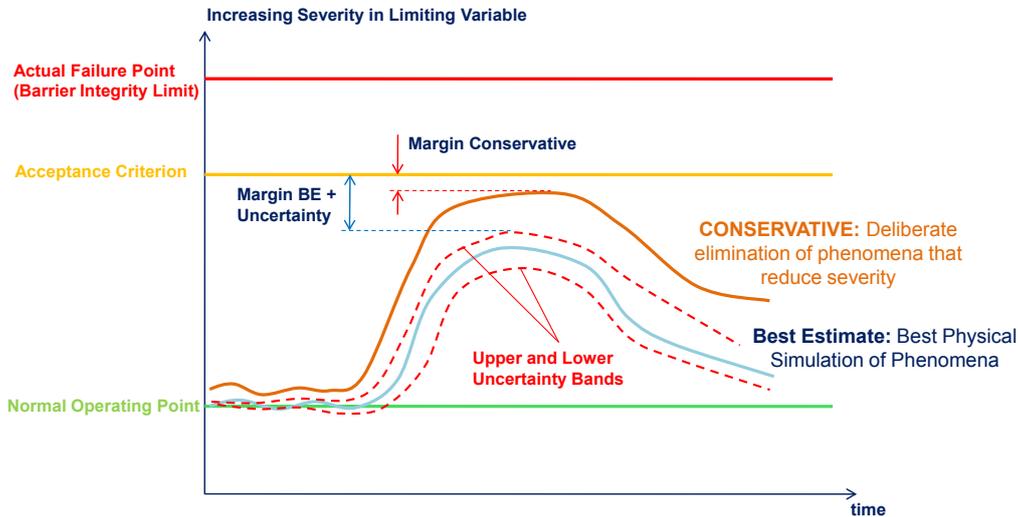


Figure 2.4: Schematic showing normal operating value of a parameter, simulated transient condition with uncertainty bands, acceptance criterion and actual failure point (Macian, 2010). With a best-estimate calculation the margin between the upper uncertainty band on the simulated value and the acceptance criterion is larger than with the conservative margin.

2.3.2 Towards Best-Estimate Modelling

There are three accepted computational methods for analysing design basis accidents which can be used to demonstrate safety for licensing purposes (IAEA, 2009):

1. Use of conservative computational models with conservative initial and boundary conditions (conservative analysis);
2. Use of best-estimate computational models combined with conservative initial and boundary conditions (combined analysis);
3. Use of best-estimate computational models with conservative and/or realistic input data but coupled with an evaluation of the uncertainties in the calculation results, with account taken of both the uncertainties in the input data and the uncertainties associated with the models in the best-estimate computer code (best-estimate analysis).

The first method is the traditional approach to nuclear reactor safety (NRS) analysis. Inherently conservative ‘systems codes’ are used: simplified 0D or 1D models of whole component interactions using lumped-parameter physical models such as RELAP-5 (Smith, 2010). Systems codes are based upon a very large database of correlations from many simple isolated-parameter experiments; the trustworthiness of their predictions rests on the large body of experimental data.

However, the use of results is necessarily conservative to compensate for the lack of physical realism in the models. For example, the power required to cause Departure from Nucleate Boiling (DNB) in reality could be 1.5 times the design basis power (Kiltie, 2011). Large safety margins have a deleterious effect on system performance and operational flexibility (IAEA, 2009). Furthermore, conservative assumptions may lead to incorrect prediction of accident progression or miss out relevant physical phenomena (IAEA, 2009). For example, it is not possible to represent complex 3D flows, including natural circulation and in the RPV, by 1D approximations (Smith, 2010).

The second method is an improvement on the first but does not take full advantage of the predictive power of best-estimate computational models. The third method is the most realistic approach, combining parameters and margins statistically and expressing the overall safety margin as a function of a chosen confidence level. While systems codes are conservative, CFD can be a best-estimate tool. Best-estimate 2D and 3D CFD methods have begun to be used for NRS analysis to address the inadequacy of system codes in many applications where the flow is strongly three-dimensional, providing physical insights and leading both to better designs at reduced cost and to more precise quantification of safety margins (IAEA, 2003).

Increasingly the best-estimate approach to safety is being adopted, combined with uncertainty evaluation. Many accident scenarios in civil reactors, including LOCAs, are now commonly investigated in this way (IAEA, 2009). Some examples of CFD studies of key nuclear engineering validation cases are given in the following section.

2.3.3 A Brief History of Nuclear CFD

Single-phase CFD has been used in nuclear engineering for at least 30 years (NEA, 2007a). Initially, the meshes employed were very coarse by modern CFD standards and codes relied on correlations rather than resolving boundary layers and underlying physics (NEA, 2007a). There are many examples of CFD use for safety cases, some of which are summarised here. Smith (2010) provides a comprehensive list of applications. For definitions of all the CFD terminology used here, see Chapter 3.

The development of a cold coolant plume in the inlet and downcomer of a VVER-440 RPV, replicating emergency core cooling in an accident, was simulated using the FLUTAN code (Macek et al., 1997). In the same RPV, the consequences of a large break LOCA were simulated using the commercial code CFX with a RANS model to provide a steady-state 3D flow field, which was supplied as initial conditions to the 1D systems code ATHLET (Kliem et al., 1999). While CFX results agreed well with experiments in normal operating conditions, the results in accident conditions (loss of one coolant pump) did not agree well with experiments because the simulations were steady-state and did not account for transient effects. Density-driven flow in the downcomer and lower plenum of an RPV in case of a large pipe break LOCA was simulated using a Reynolds stress model (RSM) in CFX and an LES model in the code TRIO_U (Höhne et al., 2006). Both calculations agreed qualitatively with experimental data. It was recommended to use a hybrid RANS-LES or detached eddy simulation (DES) approach to achieve greater accuracy with higher computational efficiency.

Most studies have used steady-state RANS turbulence models but there are exceptions, albeit predominantly as demonstrations. Flow across a tube bundle, representing an array of fuel rods submerged in coolant, was simulated using an RSM model and an LES model on coarse and fine unstructured grids by Benhamadouche and Laurence (2003). They found that LES compared well with experiments and DNS on the fine grid, and still reasonably well on the coarse grid. Rolfo et al. (2010) conducted similar tests of LES and hybrid RANS/LES in flow over tube bundles, finding that the hybrid method was inaccurate near the wall because of the formation of the blending function. Veber and Andersson (2004) applied DES to the flow and thermal mixing in a T-junction. It was found that DES generated more realistic fluctuations than U-RANS with the $k - \varepsilon$ model, but that the results were still quantitatively different to experimental data.

A lot of CFD for NRS studies have been conducted using commercial codes, the aforementioned CFX being one. FLUENT has been used by many research teams since the late 1990's, beginning with the Nuclear Energy Agency (NEA) International Standard Problem 43, "Rapid Boron Dilution Transient Tests For Code Verification" (Gavrilas and Kiger, 2001). Simulations of the Swiss PANDA containment test facility (Scheuerer et al., 2005b), and the RPV and primary circuit of a VVER-1000 reactor (comparing turbulence models and numerical methods) (Rohde et al., 2007) were also performed using FLUENT.

Simulations of tracer mixing in the VVER-1000 RPV were done with the $k - \omega$ SST turbulence model in CFX (Moretti et al., 2009). This work concentrated on the effect of improved estimates of boundary conditions. A

mesh of 6.5M elements was generated with hexahedral elements in the down-comer annulus and tetrahedral elements in the lower plenum because of its geometric complexity. Quantitative agreement of the distribution of a tracer with experiments was limited but quite satisfactory qualitative agreement was obtained.

The developers and vendors of commercial CFD codes have a major role to play in NRS analyses. The main industrial CFD vendors operate in a very competitive market and have traditionally been resistant to participating in validation exercises which could lead to embarrassing code-to-code comparisons. Validation has in some cases been performed at the behest of a customer but the results may have been restricted or not (yet) published. More recently, awareness among the users of the importance of quality and trust in CFD has increased and the vendors have become more active in open benchmarking activities (Smith, 2010).

2.3.4 Current Challenges

Comparatively little relevant and detailed 3D experimental data is currently available for validation of CFD (NEA, 2008). Typical CFD problems in nuclear engineering are also very demanding in terms of complexity, requiring massive computational resources. A lot of effort is needed both to provide benchmark data and to validate physical models before attempting practical applications in nuclear safety. This is particularly the case for two-phase flow which is at a rudimentary stage of development, although some aspects are fairly well predicted by advanced models. Simulation of more complex physics such as combustion is also in its infancy (IAEA, 2003).

Nor is incompressible single-phase CFD completely adequate. According to the International Atomic Energy Agency (IAEA, 2003),

“...single-phase [CFD] applications, in which the geometric complexity is the only challenging issue, are dealt with quite successfully these days using commercial CFD software, although further improvements need to be made in the turbulence modelling area.”

Extensive code-to-experiment and code-to-code benchmarking is recommended by the IAEA to address these shortfalls. Furthermore, since most CFD practitioners’ knowledge comes from personal experience, advice from co-workers and code manuals, development of best-practice guidelines is essential to the nuclear industry (NEA, 2007a).

The computational overhead of transient CFD simulations in comparison with system codes is a serious limitation on the use of CFD for safety demonstrations, even for single-phase flows. Part of the reason is that uncertainty

analysis must be done with a high level of rigour. The uncertainty quantification methodology for system codes requires a large number of computations (NEA, 2008). A Monte Carlo approach to uncertainty quantification is generally prescribed by regulatory authorities. This involves selecting a random set of parameter values from within the range of likely values as inputs to a particular simulation. A large number of simulations have to be run to establish uncertainties in the outcome. The advice given by the Nuclear Energy Agency (NEA) to those who wish to conduct CFD for NRS is therefore to buy a large high-performance computing (HPC) facility (NEA, 2007a).

From a regulatory point of view it is useful to have open access to the CFD source code; this is a point of contention with commercial code vendors such as ANSYS (Smith, 2010). Open-source software such as Fluidity offers a viable alternative. Turbulence modelling is at a more advanced, albeit experimental, stage in open-source CFD codes. Geometrical complexity is dealt with in a superior way in unstructured mesh codes with adaptive capability, e.g. Sampaio et al. (2004). Adaptive meshing techniques are promising because they tackle numerical errors which in any case must be dealt with before improving the accuracy of physical models (IAEA, 2003).

2.3.5 Best Practice

This section discusses the practical details of the best-practice use of CFD for NRS applications. Simply put, best-practice CFD usage follows these three steps:

1. Validate model vs. benchmark data in system state X ;
2. Quantify uncertainty in results and sensitivity of model to changes in input data around state X ;
3. Make prediction of dependent variable in new state Y .

The level of accuracy required, resources available and impact of the results are considered. Verification of code, validation of models, trustworthiness or usefulness of experimental data and interpretation of results are also discussed. To round off the section, methods for quantifying uncertainties and sensitivity to changes in input data are described. The discussion is illustrated with examples from Rolls-Royce Submarines and elsewhere in the nuclear industry.

For definitions of particular CFD methods, refer to the glossary and to Chapters 3, 4 and 5. For a full report on best practice for CFD of NRS, see NEA (2007a).

2.3.5.1 Scope of CFD Study

The complexity required of a CFD model (or level of approximation, to use the terminology from Chapter 3) depends on the details of the flow under consideration, the impact of the result and the accuracy required in the result (Rolls-Royce, 2010). However, a compromise is often made between the accuracy desired and what is computationally affordable. In general, RANS turbulence models have been used thus far for NRS cases, partly because they are simple and quick to solve compared to LES or DNS, but also because they exhibit convergence with mesh refinement (NEA, 2008). LES is much slower and more resource-intensive than RANS (a single LES run might take days) for a gain of accuracy that might not be needed.

Steady-state phenomena such as pressure drop across a component can be investigated with RANS simulations using a simple turbulence model which are often ‘good enough’ in terms of accuracy (NEA, 2008). The level of effort and expense required to make further improvements in accuracy by moving to a more detailed CFD model may be unjustified (NEA, 2007a). However, if the RANS model is unable to recreate a fundamental physical phenomenon, the extra effort may be worthwhile. The simulation of a large break LOCA with a RANS model by Kliem et al. (1999), detailed above, is a prime example.

In very large and complex simulations, such as an entire RPV, even moderately fine meshes contain many millions of elements. Demonstrating that a simulated result is mesh-independent by refining the mesh further is infeasible (Moretti et al., 2009). In that case it is necessary to demonstrate that best-practice has been followed in mesh design, for example in terms of near-wall resolution.

Industry timescales can be severely limited: a CFD study may be expected to be finished in a matter of a few weeks to a few months. A single RANS run may converge in an hour or less depending on the number of processors used. This means that a batch of runs, covering a space of several parameters in rudimentary detail, can be run overnight or over a weekend. With the resources typically available in a private company (small HPC cluster, limited commercial software licenses, several users; cf. §2.4) there is often no choice but to use RANS.

A slight gain in accuracy can be had with U-RANS (cf. §3.3.5.4), which resolves some large-scale transient dynamics. For example, U-RANS of turbulent mixing inside RPVs was performed by Rohde et al. (2007). In terms of specific turbulence models, they found that the $k-\varepsilon$ and $k-\omega$ SST models (cf. §4.2) were suitable for momentum-driven mixing, whereas more sophisticated models were needed for buoyancy-driven mixing. However, U-RANS

was found to be unsatisfactory because it was only capable of representing large slow-moving transient structures, while the turbulent eddies in the RPV downcomer (responsible for mixing) were smaller and more rapid. They therefore recommended using LES or another scale-resolving method such as DES for this problem (NEA, 2007a).

If rapidly evolving transient phenomena are present in the real flow, they will not be resolved by RANS/U-RANS study and LES is required. LES is then justified if RANS has been shown to be inadequate, for example to investigate thermal fatigue due to oscillating temperatures in a pipe (Westin et al., 2006). Nevertheless, RANS is not made obsolete by LES since it provides a good first approximation and the solution can be used as initial conditions for LES. RANS can also provide an initial sweep of parameter space, reducing the scope of the LES study.

Certain turbulent phenomena are present in the Rolls-Royce RPV, which could only be simulated using LES. Rolls-Royce Submarines could save large amounts of money and time spent on laboratory experiments if they were able to investigate these with CFD in a sensible amount of time. One reason why LES is seldom used in Rolls-Royce Submarines is the large amount of validation effort needed, even though similar efforts may be spent on calibrating a hydraulic test rig. Indeed, having built a rig, the justification for performing expensive CFD studies is somewhat lessened. The key question is which technology offers the best value for money (Kiltie, 2011).

Although potentially more accurate for transient flows and flows with 3D characteristics, LES does not converge to a solution in the same way that RANS does. Rather, the instantaneous solution depends on the mesh spacing, so any attempt to demonstrate convergence in an instantaneous variable will be confounded (Meyers et al., 2003). Convergence for LES can be demonstrated by selecting a dependent variable that represents averaged flow behaviour (e.g. mean flow velocity, turbulence spectrum) and that is still important to the goals of the analysis (NEA, 2007a). State of the art LES error estimation methods will be considered in §5.4 with a view to applying them to NRS.

Use of LES makes greater demands on mesh resolution than RANS, particularly at the wall (cf. Chapter 5). LES use is limited by affordable computation to moderate Reynolds number flows in complex geometries (see, e.g. Höhne et al. (2006), Bieder et al. (2007)) or high Reynolds number flows in simple geometry. Hybrid methods combining RANS or wall functions near the wall with LES elsewhere, such as DES (Spalart, 2009), are expanding the range of wall-bounded flows which can be simulated by relaxing the resolution requirements at the wall. However, best-practice recommendations for these models have not yet been agreed on (NEA, 2007a).

2.3.5.2 Impact and Confidence

The impact of a CFD study relates to the safety consequences of getting the answer wrong, and so to some extent it determines the analytical methods used. According to Rolls-Royce Submarines best-practice guidelines, the impact of a computational study is split into 3 categories: high, medium and low impact (Rolls-Royce, 2010). An example of a high-impact study is one which makes quantitative predictions in an accident scenario that is included in the design basis. A medium-impact study, on the other hand, would be a best-estimate prediction in an accident scenario that is outside the design basis. A low-impact study includes qualitative ‘scoping’ calculations, for example to understand model behaviour.

The required accuracy and confidence level in the analytical method increase with the impact level. Confidence should be demonstrated in seven separate aspects:

1. Geometrical model: adequate detail, manufacturing tolerances taken into account.
2. Satisfaction of mathematical assumptions, e.g. homogeneity, eddy-viscosity.
3. Equations: sound scientific theory, attention to novel methods.
4. Numerical solution procedure: accuracy, convergence, predictability, suitability for situation.
5. Empirical data: appropriateness, veracity of tables, curve fits and correlations.
6. Appropriateness of safety margins.
7. Previous related validation evidence.

The overall confidence in the method is only as good as the lowest confidence in any of the above. If the confidence level is not high enough for the level of impact, validation work has to be undertaken in order to increase confidence. This might be through comparison with experimental data (which brings its own set of problems regarding accuracy and confidence) or other simulated results, analytical results or a sensitivity study. If validating against another code, that code must be validated against experimental data in a similar situation (Rolls-Royce, 2010).

2.3.5.3 Experimental Data

Benchmark data for nuclear applications are scarce (NEA, 2008). Monitoring in-service plant is difficult, not to mention hazardous. Building identical life-sized ‘safe’ versions (without radioactive materials) purely for testing is very expensive, but simplified test rigs may not recreate all of the physical effects present in the real plant. For example, full-scale experiments to investigate fuel melt scenarios such as the Chernobyl disaster are very expensive and difficult to perform, but scale models may not faithfully recreate the true conditions of a melt (Sehgal, 2006). A scale model of a large complex component like the RPV may cost on the order of £100,000 – £1 million (Kiltie, 2011). Rolls-Royce Submarines understandably wish to reduce the cost of rig testing by using CFD to reduce the cost of design development.

Even if a realistic model is built, it may suffer the same limitations as the real plant: a scale-model RPV for high-pressure hydraulic testing has to be made from steel, so the flow patterns cannot be observed directly without using sophisticated imaging methods such as neutron radiography (Takenaka et al., 1990). In this case a good compromise is to build an ambient-temperature, low-pressure transparent Acrylic or Plexiglas RPV (Figure 2.5) which permits the use of high-fidelity imaging techniques such as laser-doppler anemometry (Höhne, 2007). With limited data, the CFD modeller has the challenging job of determining where the important physics occurs as well as how accurate the simulated result is. Evaluation of uncertainties and sensitivity becomes crucial in this situation.

If building a scale model, it is important to ensure that the flow, as well as the geometry, is scaled correctly. Using the dimensionless variables of time ($t^* = t/t_0$), space ($\mathbf{x}^* = \mathbf{x}/l_0$), velocity ($\mathbf{u}^* = \mathbf{u}/\mathbf{u}_0$) and pressure ($p^* = p/\rho u_0^2$), where the subscript 0 denotes a reference value and ρ is the density, the non-dimensionalised Navier-Stokes equations for incompressible flow in the absence of external forces (cf. §3.2.1) are written:

$$\text{Str} \frac{\partial u_i^*}{\partial t^*} + \frac{\partial u_i^* u_j^*}{\partial x_j^*} = -\frac{\partial p^*}{\partial x_j^*} + \text{Re}^{-1} \frac{\partial^2 u_i^*}{\partial x_i^* \partial x_j^*}, \quad (2.1)$$

where two dimensionless groups have appeared (Ferziger et al., 1999). These are the Reynolds number $\text{Re} = l_0 \mathbf{u}_0 / \nu$ (ratio of inertial to viscous forces) and Strouhal number $\text{Str} = l_0 f / \mathbf{u}$ (describing oscillating flow systems). ν is the molecular viscosity, l_0 is a characteristic size of the flow geometry, e.g. hydraulic diameter and f is a characteristic vortex frequency. If scale-model testing is to bear any resemblance to the real plant, the dimensionless numbers Re, Str and Eu should be as close as possible to real plant conditions (Pierce and Stevens, 1975).



Figure 2.5: Plexiglas model of ROCOM RPV used to obtain detailed flow data (Höhne et al., 2006).

For practical reasons it is difficult to achieve equality of Re : the velocity cannot be high enough. However, at very high Re there is usually very little variation in flow dynamics, so a lower Re can be used. The Strouhal number measures the relationship between the dominant turbulent motions and the geometry. Strouhal number equality is satisfied if the geometry is a true scale model and the velocity is correctly scaled (Pierce and Stevens, 1975).

2.3.5.4 Verification and Validation

Verification and Validation (V & V) is an essential part of using CFD for NRS. Verification is the process of checking that the encoded equations and physical models actually are implemented correctly. This is often achieved using the Method of Manufactured Solutions (MMS) (Roache, 2002). MMS is a method by which an analytical solution is constructed (possibly bearing little resemblance to a real physical solution) and a series of simple tests on a set of increasingly fine meshes is performed, the objective being to check that the rate of convergence is that which is expected of the discretisation method. If there is an error in the model, it will show up in a reduced convergence rate as long as it doesn't also contaminate the analytical solution.

Commercial codes are claimed to be verified by the vendors but it is hard

to find published sources giving details. If user-defined ‘plug-ins’ are added to the commercial software, these will need to be verified independently (Rolls-Royce, 2010). In-house software requires its own set of verification tests. These are ideally set up as part of a code version-control system so that all the tests are run every time the code is updated (Rolls-Royce, 2010). For example, the open-source code Fluidity uses a fully-automated V & V method for quality assurance and to aid collaborative code development (Farrell et al., 2010, 2011).

Validation is a test of whether the CFD model is a good approximation of the real phenomenon: that is, how realistic the numerical model is. It is achieved by comparison of simulated results to benchmark data. A hierarchical approach is recommended, starting by testing each element of the physics on its own, then comparing against data from benchmark cases (semi-realistic, highly instrumented experiments designed to showcase particular combinations of physical effects), and building up to simulating the complete system (e.g. a whole containment building) (Oberkampf and Trucano, 2002). There may only be limited validation data for the whole-system case, but because of the validation efforts that have built up to it, greater confidence can be placed in the whole-system simulation results (Oberkampf and Trucano, 2002).

Acceptance or rejection of the simulated result is not necessarily a clear-cut decision (NEA, 2007a). It is for a group of experienced engineers to decide, on balance, whether the result is acceptably accurate for the purposes it will be used for. The balance depends on a weighing-up of the uncertainties, the behaviour of the model, the interdependence of errors and the impact of the decision (NEA, 2007a; Rolls-Royce, 2010).

In coming to a conclusion about the validity of a result, engineers must be wary of ‘model builder’s risk’, i.e. rejection of a valid computational model because of modelling and experimental errors, and ‘model user’s risk’, i.e. acceptance of an invalid model due to cancellation of errors from different sources (NEA, 2007a). The latter has the more disastrous consequences and is common in practice (Oberkampf and Trucano, 2002). For example, modelling and discretisation errors in LES can and do cancel each other out in an unpredictable fashion (Klein, 2005). A CFD result might agree with flawed experimental data, giving false confidence (NEA, 2007a).

2.3.5.5 Interpretation of Results

Several levels of comparisons can be made between the computed solution and benchmark data which vary in their veracity and difficulty of calculation (Oberkampf and Trucano, 2002). The most basic is an ‘eyeball norm’: does

the flow field (velocity, temperature etc.) look like the data? This is purely qualitative but is a useful first check of whether the simulation has succeeded, for example by checking maxima and minima.

The next level is a quantitative comparison of the dependent variable, which may be an integral or a time average. Selection of the right metric is crucial; it is too easy to think that the solution is correct by measuring something which is insensitive to the input variables. This is discussed in the context of LES by Pope (2004) and in relation to adaptive modelling in Chapter 5 of this thesis. In making the comparison, uncertainties should be taken into account: do the error bars on the computed variable and comparison data overlap, and to what extent?

The toughest comparison is between an ensemble result of several simulations and the data. The dependent variable can be represented as a probability distribution in this case, which is more useful from an NRS point of view. A distribution is easily incorporated into the probabilistic method of combining uncertainties for calculating safety margins (Oberkampf and Trucano, 2002).

2.3.5.6 Uncertainty

According to the NEA (NEA, 2007a),

“The field of experimental uncertainty analysis has a very long history, although the history of use within nuclear reactor safety is not nearly as long as it should be.”

Uncertainties come from a variety of sources, both aleatory and epistemic, affecting both the simulated result and the benchmark data. Physical models, initial conditions, boundary conditions and material properties, as well as iteration, interpolation and discretisation schemes all contain uncertainties. The predicted total uncertainty in a simulated result should be calculated by a statistical combination of all contributory uncertainties. Various methods are described in the literature, e.g. the framework of Oberkampf and Roy (2010) outlined below. It is advised to run a large number of simulations exploring the parameter space so that the variation of the outcome can be measured directly and compared to the predicted total uncertainty. If the predicted and measured uncertainties agree, then the statistical combination method is reliable (NEA, 2007a).

Oberkampf and Roy (2010) described a comprehensive method of uncertainty quantification for scientific computing. In their method, aleatory and epistemic uncertainties are treated differently in terms of how they are

propagated through the model. Aleatory numerical errors (such as discretisation errors) are estimated by verification techniques such as MMS. Epistemic modelling errors (such as simplifying assumptions) are validated against experimental data. Uncertainties are then extrapolated to areas where experimental data is not available. Their method may be suitably rigorous for application to CFD in nuclear safety cases.

2.3.5.7 Robustness and Sensitivity

Robustness is not a rigorously defined concept but is nonetheless important when designing and using a CFD model. A system described by a differential equation with initial and boundary conditions and containing some defining parameters (such as material properties) is said to be robust if its response to a small change in initial or boundary conditions or material properties is proportionate, i.e. a small change does not cause a sudden jump in output (Zienkiewicz and Taylor, 1997).

A more general concept of robustness is that a CFD model can be applied in a wide range of different flows with comparable accuracy and without ‘fudging’ or tuning of input parameters to the flow. A non-robust model in this sense is the RANS $k - \epsilon$ model because it is not accurate in all the situations listed in §4.2.3.1. It contains five model constants which were tuned to fit certain simple turbulent flows. The dynamic LES model, containing only one independent parameter, is more robust because it self-tunes to the flow conditions and mesh resolution (cf. §5.2.2). The hypothetical ideal CFD model would be universally applicable with no tuneable parameters.

However, there is no such thing as a universally applicable CFD model because all models contain simplifying assumptions which reduce their range of applicability. Thus, a model validated for one situation cannot necessarily be reliably applied to another. The response of the model to changes in input data must be measured to check whether it is robust to variations in the data. A method that is validated at state X but exhibits sensitivity (non-robustness) to a small change δX cannot be trusted at state $Y = X + \delta X$ in the absence of benchmark data at Y to validate the results.

Using the optimal CFD model found from the validation exercise, the sensitivity of the output variable to various input parameters can be systematically explored using the Monte Carlo approach, or more efficiently with the Design of Experiments (DoE) methodology (Chabard and Laurence, 2009). DoE is a structured way of investigating the sensitivity of some output variable from a system to the space of input parameters by independently testing the input parameters; an example is Latin Hypercube testing (Park et al., 2006). The results of a sensitivity study can be mapped as a response surface,

showing where the output varies most rapidly in response to small changes in input factors. In areas where it changes slowly, the system is said to be robust in relation to those factors (Eriksson et al., 2008).

At the most basic level of DoE, each input parameter (factor) has two possible levels (two numbers if a continuous variable or two categories if categorical). A more sophisticated study might involve > 2 values of some factors. If all the n factors are interdependent, then every permutation of the factors should be tested and DoE calls for levels ^{n} individual experiments. For four factors and two levels, this is already $2^4 = 16$: the number grows rapidly out of practicality.

In reality, combinations of large numbers of factors are assumed to be insignificant compared to varying individual factors, so the number can be reduced. If groups of factors are independent from each other, the number can be reduced even more. For example, in a space of seven factors $\{A, B, C, D, E, F, G\}$, if $\{A, B, C\}$ are independent of $\{D, E, F, G\}$ then the number of experiments needed is ideally $2^3 + 2^4 = 24$, much less than $2^7 = 128$. The number can be further reduced by other statistical sampling tricks (Eriksson et al., 2008).

The number of experiments required to get a good spread of results may still be large, however. In a CFD study this means that each run has to be quick, restricting us to RANS as the only suitable model and therefore limiting the kinds of variables that can be investigated. Even RANS may be too computationally intensive, in which case a ‘surrogate model’ of the actual geometry/setup can be used. This is a simplified model used only to show those factors to which the model is sensitive and those to which it is not (Queipo et al., 2005). A more detailed study can then probe the reduced space of parameters.

2.4 High Performance Computing

The demands of CFD for NRS are severe, even for moderately complex problems. Larger and more complex studies, such as mixing in an RPV and multiphase flows, need levels of computing power that are currently inaccessible to commercial enterprises. CFD simulations of any appreciable size have to be run across very large numbers of parallel processors if they are to be completed within a practical length of time. The current state of high-performance computing (HPC) and the practicalities of running a HPC facility or cluster are outlined in this section.

2.4.1 Hardware

The current trends in computational hardware development are encapsulated by the following system parameters:

- *Transistor density*: Increasing density of transistors on a microprocessor is governed by Moore's Law, an observation that the density doubles approximately every two years. Moore's Law has held true since the 1970s and is expected to do so for at least the next decade (Dongarra and Beckman, 2011). The trend is evidenced by the gradient of Figure 2.6.
- *Reliability*: as transistors get smaller, the relative amplitudes of random noise and signals get closer together, so building reliable components becomes ever more challenging (Dongarra and Beckman, 2011).
- *Power consumption*: current state-of-the-art HPC systems consume around 10MW of power which brings with it logistical challenges: most sites are not set up to supply such large amounts of power (Dongarra and Beckman, 2011).
- *Communication*: the overhead of moving data between processors increases with system size.
- *I/O*: ever-increasing amounts of data are passed in and out of the cluster, both input conditions for simulations and results, which causes a bottleneck in data input/output (I/O) and memory (Dongarra and Beckman, 2011).
- *Concurrency*: due to fundamental hardware limits, computing power is growing by increasing the number of cores, rather than the processor speeds (Dongarra and Beckman, 2011).

These trends relate mainly to cutting-edge systems, which are mainly bought by government laboratories and are beyond the reach of industrial users. However, the benefits and technology filter down at a consistent rate to second-class systems and beyond. Figure 2.6 overleaf shows the rate of increase of power in the top 500 supercomputers and demonstrates that the 500th fastest system keeps pace with the fastest.

According to Figure 2.6, the fastest HPC system in the world in 2018 is predicted to pass the one exaflops (1×10^{18} floating-point operations per second) mark². In 2012 the fastest system has recorded a peak rate of 16.32

²A floating-point operation is the most basic operation in a computation, e.g. changing a binary digit.

2.4 High Performance Computing

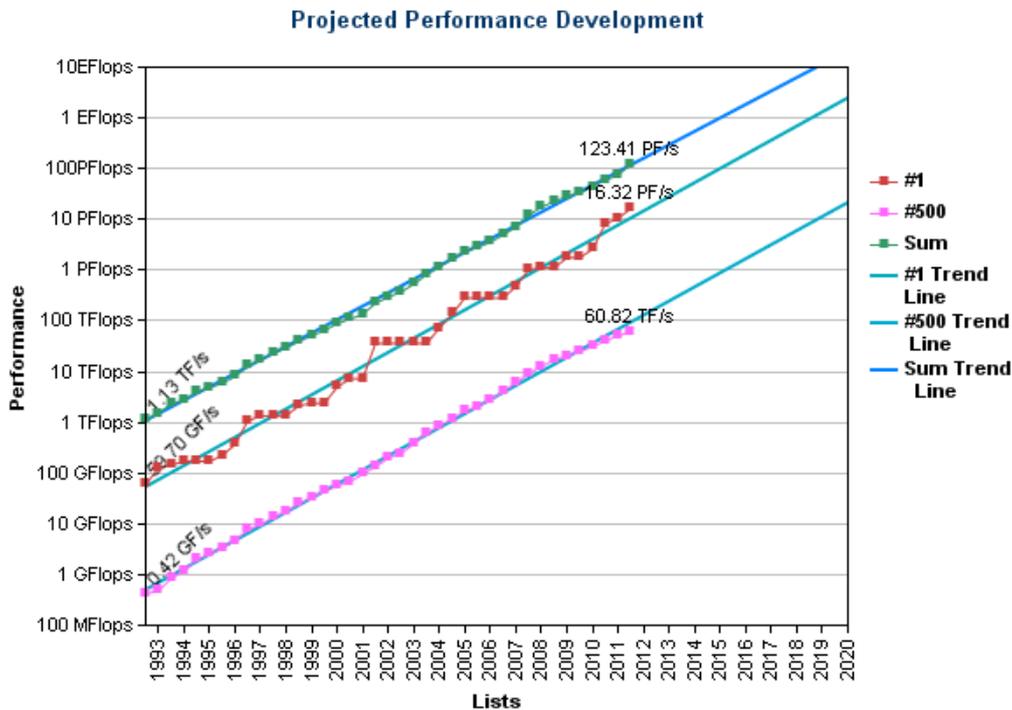


Figure 2.6: Exponential increase in supercomputing power in the Top500 list of the most powerful HPC systems in the world, measured by the Linpack benchmark test, 1993-2011 with projection to 2020 (Top500, 2012). Numbers one and 500 each year and the total power of all 500 are plotted. The trendlines show that the fastest HPC system in the world is predicted to pass one exaflops (1×10^{18} floating-point operations per second) by around 2018.

petaflops (1.632×10^{16}), while the 500th fastest system managed 60.82 teraflops (6.082×10^{13}), approximately 270 times slower (Top500, 2012).

Central processing units (CPUs) from ordinary PCs have traditionally been used in clusters. To get around hardware limits in CPUs, new types of HPC system are being developed. Graphical processing units (GPUs) developed for games consoles are gaining popularity because of their excellent performance in solving medium-sized matrix equations (WTEC, 2009). Some scientific codes, e.g. Fluidity are being translated for use on massively parallel arrays of GPUs and heterogeneous arrays of CPUs/GPUs (Markall et al., 2010).

2.4.2 Software

Due to the concurrency trend, simulations must be amenable to being spread across large numbers of concurrent processes if they are to be scaled up to large HPC systems successfully. The performance of parallelised software in relation to the number of processors is measured by its *scalability*. Two definitions of scalability are used: *weak* scalability, defined as the relationship of solution time to number of processors for a set problem size per processor, and *strong* scalability, defined as the relationship of solution time to number of processors for a set total problem size (Howes, 2010). The speed and efficiency of individual operations is measured by *latency*, defined as time required to send an 8-byte message from one node to another, and *bandwidth*, defined as message size divided by the time it takes to transmit a message (HPCAC, 2009).

Industrial CFD users demand quick turn-around times for results and consequently large numbers of processors for large computational meshes. As job size grows and domains are partitioned into ever larger numbers of subdomains, the overhead of passing data around during the solution procedure grows and can severely degrade the scalability of the problem (Gropp et al., 2001). Since CFD solvers use sparse data structures, they demand low latency and high memory bandwidth, so a large amount of communication between processors is inevitable. I/O needs to be handled in such a way that it does not add significant overhead (SGI, 2009). Examples of demanding jobs (in 2009) are a 100-million-cell steady state model on > 64 cores, or a moderate-sized, moderately-parallel transient model that requires many solution writes to collect time history data for post-processing (Posey et al., 2009).

Most companies running CFD simulations use commercial codes. Most commercial CFD codes rely on the distributed memory paradigm and use the message-passing interface (MPI) in order to make the code portable over many different computer architectures. MPI is commonly used in open-source CFD codes as well, e.g. Fluidity (Guo et al., 2010). Such portability is beneficial but it has a drawback: code scalability depends heavily on the topology and connectivity of the cluster (SGI, 2009). For example, strong scaling of an adaptive finite-element CFD code up to 32,768 cores was demonstrated by Sahni et al. (2009) on one architecture (IBM BlueGene), but it did not scale as effectively on other architectures.

Scaling tests of the commercial code FLUENT demonstrated weak scaling up to 256 processors for a model with heavy I/O data demands (Posey et al., 2009). The key to success was not the CFD solver, but the domain partitioning algorithm and a parallel file system which allowed parallelisation

of I/O tasks. The scalability was severely reduced on a conventional serial file system.

Increasingly, CFD software is being adapted to suit massively parallel architecture. [Göddecke et al. \(2009\)](#) modified a CFD solver to run on GPU clusters. [Fischer et al. \(2008\)](#) developed CFD solvers specifically for simulations at the petaflops (1×10^{15} flops) scale. They carried out strong scaling tests of the solver and presented results of a massively parallel LES of flow past wire-wrapped fuel pins.

2.4.3 Efficiency of Cluster Use

Any high-value industrial asset, such as an HPC cluster, should be carefully managed to maximise its efficiency or value for money. If demand for space is high, the system administrators should maintain high *availability*: in other words, minimise downtime for maintenance and maximise available time for computations. High *utilisation*, meaning maximal use of the available resources, requires that as many processors are used for as much of the available time as possible.

Simulations are scheduled by a queuing management system based on a queuing policy intended to maximise utilisation and fairness. The best policy in terms of utilisation and fairness depends on the spread of job sizes, the number and importance of jobs, the behaviour of users and the resources available ([Burbidge, 2011](#)). In industry, where time is strictly limited, users typically run large jobs overnight to have results on their desk the next morning. In academia, researchers are somewhat more flexible with their time so the workload may be more evenly spread.

The cluster is like a jigsaw into which jobs can be fitted: as the space fills up, smaller jobs fit into the remaining holes. As jobs finish, new jobs can replace them. A ‘one in, one out’ policy would replace a large job with a large job and small job with a small job. In a ‘fair share’ policy, jobs are given different priority and a user’s priority is reduced as they submit more jobs. This gives new users a boost: they get their jobs started faster ([Burbidge, 2011](#)).

Better use of resources requires that users are educated in how to choose the optimum number of processors for the job size. Each code and problem is associated with an optimal partitioning in which the sum of the communication overhead and the solution effort on each processor is minimised. Finding it may be a matter of trial-and-error and experience. HPC ‘tuning’ consultants may be a cost-effective solution to parallel optimisation of open-source codes ([Bischof et al., 2011](#)).

2.4.4 Costs

In tandem with the exponential increase in computing power, the cost of performance drops so that one can buy a system twice as powerful in two years' time for roughly the same money. With that in mind, any financial decision is out of date within two years (Burbidge, 2011). Computing is a large overhead for a modern engineering company so the size of the cluster must be considered carefully: do the benefits outweigh the costs? Future expansion of the facility can spread the cost, but a larger building to house the future facility is required. A common strategy taken in industry to reduce costs is to outsource running of the cluster. However, if the HPC function is business-critical, for example to run CFD studies for nuclear reactor safety, it might be advisable to keep it in-house to preserve control over its operation.

A sensible way to keep up with the pace of technological progress and spread costs is to replace a proportion of the processors every year or two. Older equipment can be kept for much longer than two years and used to run lower-priority jobs while the newest equipment is reserved for high-priority jobs. Added advantages of this strategy are that cluster downtime is reduced and backward compatibility with software is maintained on the older equipment (Burbidge, 2011).

Aside from the large initial capital cost and the cost of continually replacing worn-out equipment, clusters have high running costs. A cluster converts large amounts of electricity into heat, requiring a powerful cooling system. For example, the 2.5 petaflops Tianhe-1A facility uses 4MW of electrical power, costing about \$3.5 million per year (NVIDIA, 2012). 50% of the running costs of the CX1 cluster at Imperial College London are electricity and cooling (Burbidge, 2011). It is possible to recover some of the cost of cooling by using the heat for building space heating, e.g. NREL (2012), though this is much easier in a purpose-built HPC building. Other major running costs include commercial software licences, floor space, staffing, support and training, insurance and asset depreciation.

3 CFD Methods and Turbulence

This chapter introduces fundamental concepts necessary to provide background to the rest of the thesis. These include the Kolmogorov isotropy hypothesis and energy cascade, and develops the Reynolds-stress and sub-filter-stress turbulence modelling frameworks from basic principles. Other methods used in this research, including boundary conditions for turbulent flow, discretisation techniques and mesh adaptivity are also introduced.

Contents

3.1	Introduction	45
3.2	Turbulent Flow	46
3.2.1	Equations of Motion	46
3.2.2	Spatial and Temporal Scales	47
3.2.3	Departures from Isotropy	50
3.2.4	Turbulence in Engineering	51
3.3	Modelling The Fluid Equations	52
3.3.1	The Multiscale Problem	52
3.3.2	Reynolds Decomposition	52
3.3.3	Filtered Decomposition	54
3.3.4	Boussinesq Eddy-Viscosity Hypothesis	57
3.3.5	Levels of Approximation	58
3.4	Turbulent Flow Boundary Conditions	64
3.4.1	Near-Wall Turbulent Dynamics	64
3.4.2	Turbulent Inflow Conditions	67
3.5	Discretising The Fluid Equations	68
3.5.1	Consistency, Stability and Convergence	69
3.5.2	Finite Element Method	69
3.5.3	Time Discretisation	76
3.5.4	Finite Volume Method	78
3.5.5	Control-Volume Finite Element Method	79
3.5.6	Stabilised Finite Element Methods	80
3.6	Meshing and Optimisation	85
3.6.1	Unstructured Mesh Adaptivity	86
3.6.2	Goal-Based Adaptivity and Adjoint Methods	90

3.1 Introduction

This chapter introduces the fundamentals of turbulence and the basic concepts involved in simulating it. Turbulence is a characteristic of many fluid flows encountered in engineering systems. Moreover, it is not simply a confounding characteristic which causes variation of the mean flow and bulk system properties, but often the most important phenomenon. Without turbulence, applications which rely on mixing or heat transfer, for example, would be vastly less efficient. Therefore, the study of turbulence is vital to the effective design of a vast array of systems.

However, it is not always necessary to know the dynamics of turbulent flow down to the last detail. Although system properties depend on all scales of motion to some extent, they may only depend on the finest details indirectly or statistically. The range of scales in turbulent flow can be very large indeed and computation of the entire range demands vast resources. The field of turbulence simulation is concerned with judiciously selecting the details that matter whilst minimising the effort required and errors incurred: in other words, striking a balance between accuracy and cost.

In the design of a hydraulic system, the pertinent physical phenomena and range of conditions may be a small subset of all the causes and effects present. Simulating every detail of the flow would be a waste of effort. We may simply want to know the effect of initial and boundary conditions (operating conditions) on a quantity of interest. This quantity is often related to the performance, efficiency or safety of equipment. Mean value predictions are usually related to system performance in the normal operating regime, and extreme values to the system's safety in extreme operating conditions. Best-estimate simulations of nuclear applications may be concerned with both of these cases.

This chapter begins by describing the nature of turbulent flow. Kolmogorov's description of fully-developed homogeneous isotropic turbulence is outlined, followed by extensions to the more general case of evolving inhomogeneous anisotropic turbulence. Next we come to a conceptual description of the hierarchy of CFD modelling approaches which form successive levels of approximation to the exact equations of fluid motion. Models for near-wall effects are also introduced. Implementation of these CFD models follows, including discretisation and solution methods. Adaptive meshing techniques are discussed in the last section.

3.2 Turbulent Flow

3.2.1 Equations of Motion

A complete description of turbulence is contained in the incompressible Navier-Stokes equations for a Newtonian fluid:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} - \nabla p - \nabla \cdot \boldsymbol{\tau} = \mathbf{F} \quad \text{in } (0, T) \times \Omega, \quad (3.1a)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } (0, T) \times \Omega, \quad (3.1b)$$

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) \quad \text{in } \Omega, \quad (3.1c)$$

$$\mathbf{u} = \mathbf{u}_D \quad \text{on } (0, T) \times \Gamma_D, \quad (3.1d)$$

$$\mathbf{n} \cdot \nu \nabla \mathbf{u} = g_N \quad \text{on } (0, T) \times \Gamma_N, \quad (3.1e)$$

where \mathbf{x} is position, t is time, Ω is the computational domain, Γ is the domain boundary divided into Dirichlet (Γ_D) and Neumann (Γ_N) parts, g_N is a specified Neumann boundary condition and \mathbf{F} is the vector of external forces. $(0, T)$ is the extent of the problem in time, where the notation $()$ indicates an open interval (one that does not include its end points). The isotropic stress tensor ∇p is due to hydrostatic (pressure) forces and $\boldsymbol{\tau}$ is the deviatoric stress tensor due to viscous effects, defined by the constitutive equation for a Newtonian fluid:

$$\boldsymbol{\tau} = 2\nu \left(S - \frac{1}{3}(\nabla \cdot \mathbf{u})\mathbf{I} \right), \quad (3.2)$$

where ν is the kinematic molecular viscosity. The rate-of-strain tensor S is

$$S = \frac{1}{2} (\nabla \mathbf{u} + (\nabla \mathbf{u})^T). \quad (3.3)$$

The equation for conservation of a scalar quantity (such as temperature) is

$$\frac{\partial \varphi}{\partial t} + \mathbf{u} \cdot \nabla \varphi + \nabla \cdot \left(\frac{\nu}{\text{Pr}} \nabla \varphi \right) = q_\varphi \quad \text{in } (0, T) \times \Omega, \quad (3.4a)$$

$$\varphi = \varphi_0 \quad \text{in } \Omega, \quad (3.4b)$$

$$\varphi = \varphi_D \quad \text{on } (0, T) \times \Gamma_D, \quad (3.4c)$$

$$\mathbf{n} \cdot \frac{\nu}{\text{Pr}} \nabla \varphi = g_N \quad \text{on } (0, T) \times \Gamma_N, \quad (3.4d)$$

where q_φ is a source or sink term for scalar φ and Pr is the Prandtl number.

The equations (3.1a) and (3.1b) are highly nonlinear, meaning that a minuscule change in initial or boundary conditions can cause vast differences

in the solution. They have yet to be shown to have a unique general solution. Even if a unique solution was proved to exist, it does not mean that finding it would be easy. Simplifications are sought in order to construct models of the equations and hence make finding solutions considerably easier.

When constructing models of the Navier-Stokes equations and producing meaningful solution data for engineering purposes, it is helpful to operate in physical (Cartesian) space and consider velocity and pressure¹. However, the underlying physics of turbulent flow are best introduced in spectral space as a kinetic energy spectrum $E_k(\kappa)$ in terms of *wavenumber* κ . In the next section, hypotheses concerning the energy spectrum of turbulence are introduced and the spectrum is broken down into contiguous ranges in which different physical effects are dominant.

3.2.2 Spatial and Temporal Scales

What follows applies to fully-developed homogeneous isotropic turbulence at very high Reynolds number. Homogeneity of a velocity field $\mathbf{u}(\mathbf{x}, t)$ means that it is statistically invariant under translation in \mathbf{x} . Isotropy means that a homogeneous field is also invariant under reflections and rotations in \mathbf{x} . Fully-developed means that it is invariant under translation in t . What constitutes a very high Reynolds number is not exactly specified.

Consider a velocity-measuring probe placed at a fixed point in a high-Reynolds number turbulent flow. It would display a complex time series composed of a spectrum of many superimposed frequency modes as shown in Figure 3.1. In physical space, turbulence is composed of a range of scales of motion with a range of turbulence lengthscales l . The largest scale in this figure is visible as a wave with period ≈ 0.7 seconds and a peak at $t \approx 0.3$ seconds. The smallest is only just resolvable by your printer or computer screen.

The lowest frequencies of the spectrum comprise the largest coherent structures (*production range*). They contain most of the kinetic energy and are hence also known as the energy-containing scales. These arise due to external forces such as pressure heads in pipes or from geometrical effects on the mean flow such as separation over sharp edges. They are usually the subject of interest in engineering simulations as they tend to have a significant effect on system performance. External forcing is responsible for significant anisotropy in the large scales, which generates turbulence: non-equilibrium

¹Solution of the Navier-Stokes equations in spectral space is a popular strategy in academia but due to limitations on mesh design it is not considered as a useful method for industrial problems at the moment and is not discussed here.

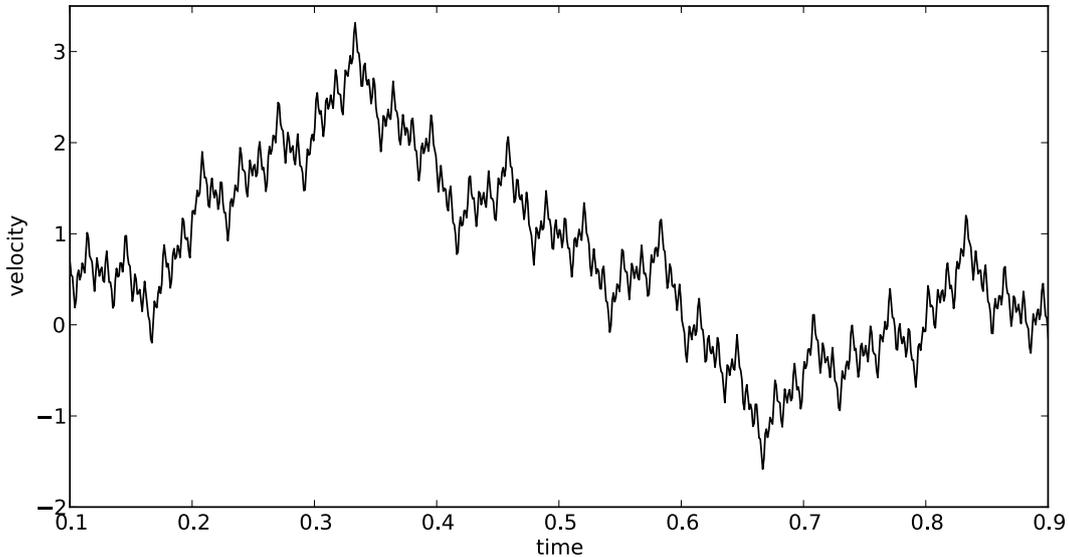


Figure 3.1: Velocity signal demonstrating superposition of scales.

of inertial forces leads to their dynamic break-up into smaller scales (Pope, 2000).

The largest scales are characterised by a lengthscale l_0 , the *integral length-scale*. It measures the ‘domain of influence’ of a coherent turbulent structure, or its statistical decorrelation length. There is an associated decorrelation time interval t_0 since a vortex passing the probe takes a certain time to do so (Mathieu and Scott, 2000). Molecular viscosity has a negligible effect on the largest scales in comparison to the inertial effect of their large characteristic velocity $u_0 = l_0/t_0$ so they are inherently unstable and break up.

As observed by Richardson (1922), the break-up of the largest coherent scales into smaller ones initiates a sequence of energy transfers, or *cascade*, to smaller and smaller scales in the so-called *inertial range*. Energy is mostly transferred by nonlinear interactions between scales. The rate of energy dissipation ε is related to the energy of the largest scales: $\varepsilon = \mathcal{O}(u_0^3/l_0)$. Kolmogorov (1941) argued that directional information in the largest scales is gradually lost as the chaotic motions of the cascade isotropise the scales. Hence, the local isotropy hypothesis is postulated:

“At sufficiently high Reynolds number, and sufficiently far from boundaries, the small scales of turbulence ($l \ll l_0$) are statistically isotropic” (Kolmogorov, 1941).

Dissipative effects become relatively more important and inertial effects less important to energy transfer as l decreases. At the *Kolmogorov scale*

3.2 Turbulent Flow

$\eta = (\nu^3/\varepsilon)^{1/4}$, viscous and inertial forces are of equal magnitude (i.e. the turbulent Reynolds number $\text{Re}(l) = \eta u_\eta/\nu = 1$). In the *dissipation range* ($l < \eta$) transfer of kinetic energy from the flow is mostly by molecular diffusion and is a random, isotropic process.

Derived from the local isotropy hypothesis, the cascade of energy in the inertial subrange is described by the power law:

$$E_k(\kappa) = C\varepsilon^{2/3}\kappa^{-5/3}, \quad \kappa \in [0, \infty], \quad (3.5)$$

where κ is the wavenumber, E_k is the total kinetic energy in the flow as a function of κ , $C \approx 1.5$ is the Kolmogorov constant (Pope, 2000) and ε is the kinetic energy dissipation rate. The three-way categorisation of the scales of motion (energy-containing, inertial and dissipative) and the inertial range power law are shown in Figure 3.2. The resolved (large) and modelled (sub-grid, or sub-filter) scales of large eddy simulation are also shown.

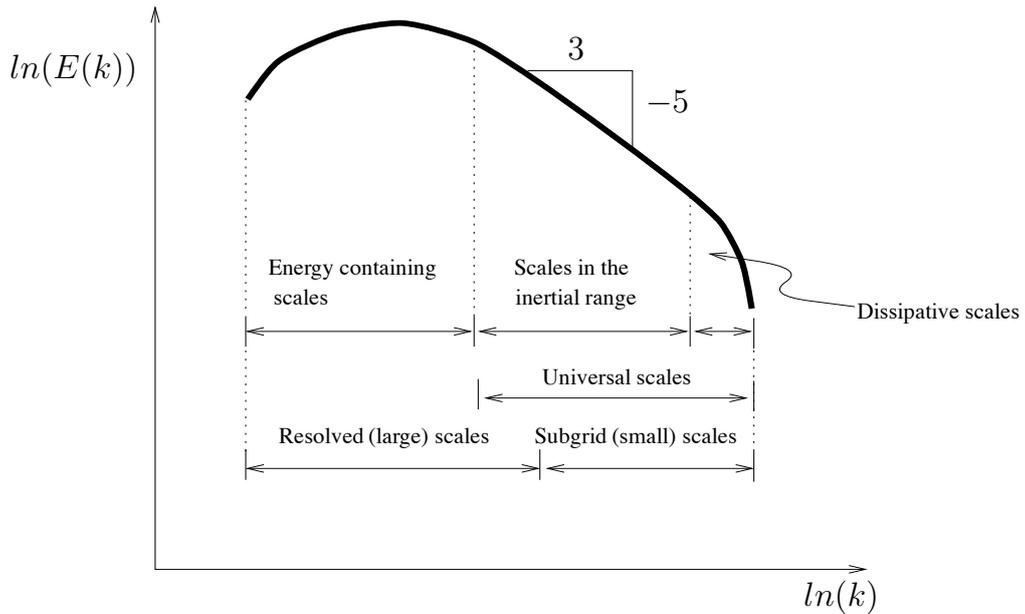


Figure 3.2: Kolmogorov's energy spectrum showing energy-containing, inertial and dissipative scales. Universal scales are independent of boundary conditions according to the first hypothesis. Also shown is the division between resolved (large) and sub-grid (small) scales in LES (cf. §3.3.3).

According to the local isotropy hypothesis, the small scales have no time memory, i.e. they are in energetic equilibrium with the large scales by immediate re-adjustment. Thus rendered independent of the large scales, universal

small-scale models can be developed. Theoretical analysis is only straightforward on statistically steady, homogeneous isotropic flows. Most turbulence models are based on theoretical arguments derived from fully-developed homogeneous isotropic turbulence. Real flows do not obey this pattern, being statistically unsteady, inhomogeneous or anisotropic (because of complex flow geometry and boundary conditions), or all three at once. Questions about the validity of these assumptions will be raised in the next section.

3.2.3 Departures from Isotropy

The Kolmogorov isotropy hypothesis has its shortcomings despite its undoubted value for understanding turbulence. For example, just what is a high enough Reynolds number for the local isotropy hypothesis to be true? At moderate Reynolds number ($Re \approx 10000$), anisotropy has been observed in the dissipative subrange (George and Hussein, 1991). Experiments in a shear layer (Mestayer, 1982) have shown that the isotropy assumption is invalid in the inertial subrange but holds close to η . Nevertheless, the $-5/3$ power law in the inertial subrange is approached in experiments as Re is increased (Mydlarski and Warhaft, 1998).

The assumption that energy is transferred solely in one direction and only between adjacent scales is a simplification. In reality, energy is transferred by triadic interactions between wavenumber triplets which may not be close together in spectral space (Pope, 2000). Furthermore, it has been shown that anisotropy in the large scales is transferred to the small scales, and that anisotropy transfer can occur without energy transfer (Sagaut and Méneveau, 2006).

Intermittency is another complicating factor. Direct numerical simulation (e.g. Méneveau and Katz (2000)) has shown that the equilibrium assumption may break down when the flow is subjected to unsteady forcing. Instantaneous values of dissipation ε were found by Sreenivasan (1991) to vary erratically.

In general, however, the discrepancies in the Kolmogorov isotropy hypothesis have an insignificant effect on the quantities of interest to the engineer. The effect of departures from the hypothesis on mean fields and Reynolds stresses is negligibly small. Large-scale flow evolution is well predicted by the theory (Pope, 2000). A few examples are now given to illustrate the importance of large-scale turbulent structures in engineering.

3.2.4 Turbulence in Engineering

The presence of turbulence in a flow can greatly influence the performance characteristics of hydraulic machinery; for example, heat transfer rates from heated walls are far greater if the flow is turbulent as opposed to laminar. Turbulence may increase wall shear stress in a plane channel or pipe but counterintuitively it reduces drag around a dimpled golf ball (Moin and Kim, 1997). Where fluids of different temperatures mix together near a surface, temperature fluctuations may enhance material fatigue (Kuczaj et al., 2010).

The geometry of a system influences the specific form and characteristic frequencies of the largest coherent structures in the flow. Large energy-containing structures affect pressure and temperature which are often the quantities an engineer is ultimately interested in. The von Kármán vortex street shed behind a cylinder in cross-flow (Figure 3.3) has a particular frequency determined by the cylinder radius, bulk velocity and fluid viscosity. In turn, large coherent structures in the flow affect flow conditions far downstream.



Figure 3.3: von Kármán vortex ‘street’ (flowing L to R) in the wake of a cylinder (Hoffman and Johnson, 2006).

A mixing layer with harmonic forcing demonstrates excitation of certain modes in response to the forcing which affect the growth of the layer (Oster and Wygnanski, 1982). Turbulence models based on the eddy-viscosity hypothesis (cf. §3.3.4) are notably deficient in this scenario, being unable to account for the excitation (Pope, 2000). Stability theory has some success in explaining the effect of geometry on the large coherent structures; see Oberleithner et al. (2011) for a full explanation.

3.3 Modelling The Fluid Equations

3.3.1 The Multiscale Problem

The smallest dynamically significant scale of motion in a turbulent flow is the Kolmogorov scale η , since scales smaller than η are dominated by viscous rather than inertial forces. The largest scale is the integral scale l_0 , measuring the size of the largest energy-containing scales (cf. §3.2.2). In a homogeneous isotropic turbulent flow, the smallest and largest scales are related to the Reynolds number by Wilcox (1998):

$$\frac{l_0}{\eta} = \mathcal{O}(\text{Re}^{3/4}). \quad (3.6)$$

Therefore, $\mathcal{O}(\text{Re}^{9/4})$ degrees of freedom are required to represent all the scales of motion in a cube of edge length l_0 . Furthermore, the temporal resolution required to resolve the most short-lived motions of timescale t is related to the Reynolds number by

$$t \propto \mathcal{O}(\text{Re}^3). \quad (3.7)$$

At typical Reynolds numbers in industrial flows ($\text{Re} > 10^6$), it is clear that it is impossible to simulate everything, even using the most powerful computers that exist today, and will be for the foreseeable future.

If we wish to reduce the number of degrees of freedom in space and time to practical levels, two questions arise. Firstly, how can the Navier-Stokes equations (3.1) be approximated or decomposed to reduce the number of degrees of freedom? Two different decompositions are introduced below, one statistical and the other deterministic. Secondly, how can we model the effect of the discarded information on the remaining (or resolved) scales? This is the subject of §3.3.5.

3.3.2 Reynolds Decomposition

Statistical tools can be used to reduce the complexity of the multiscale modelling task. According to Frisch (1995),

“For the moment the partial understanding of chaos in deterministic systems gives us the confidence that a probabilistic description of turbulence is justified.”

Since turbulence is a random phenomenon, a statistical description is natural. Note that ‘random’ does not mean that the velocity is *unpredictable*,

only that it has a different value every time the experiment is performed. Randomness in the solution to a deterministic set of equations (3.1) might seem counter-intuitive. Its cause is the acute sensitivity of the Navier-Stokes equations to initial and boundary conditions at very high Reynolds numbers.

Describing the solution variables in terms of statistical *moments* – the mean (first moment), variance (second), skewness (third) and kurtosis (fourth) – of a Probability Density Function (PDF) greatly reduces the volume of information that must be processed. By definition the PDF is an ensemble of many separate realisations of the flow. In practice, the *ergodicity principle* is invoked, which states that a system evolving for a long time ‘forgets’ its initial state, allowing the PDF is written as a time average in a single realisation (Germano, 2001).

A statistical decomposition of the Navier-Stokes equations is now given, following the presentation of Pope (2000). The velocity \mathbf{u} can be treated as a random variable if the flow is turbulent. Then its mean is given by a time average over a suitable interval:

$$\langle \mathbf{u}(\mathbf{x}, t) \rangle \approx \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T \mathbf{u}(\mathbf{x}, t) dt. \quad (3.8)$$

The time-averaging interval T should be in the range $t_1 \gg T \gg t_2$, where t_1 is the length of time for the evolution of the mean flow and t_2 is the characteristic time scale of the fluctuations \mathbf{u}' , rendering the solution $\langle \mathbf{u}(\mathbf{x}, t) \rangle$ statistically independent of time. If T is smaller, then some low-frequency modes are retained in the solution (cf. U-RANS, §3.3.5.4).

The *Reynolds decomposition* of \mathbf{u} is the sum of mean and fluctuating velocity components $\langle \mathbf{u} \rangle$ and \mathbf{u}' :

$$\mathbf{u}(\mathbf{x}, t) = \langle \mathbf{u}(\mathbf{x}, t) \rangle + \mathbf{u}'(\mathbf{x}, t), \quad (3.9)$$

and likewise for pressure, temperature and any transported scalar quantity. Now (3.1) can be written (in the absence of external forces, i.e. $\mathbf{F} = 0$) in terms of the mean variables $\langle \mathbf{u}(\mathbf{x}, t) \rangle$ and $\langle p(\mathbf{x}, t) \rangle$:

$$\frac{\partial \langle u_i \rangle}{\partial t} + \frac{\partial}{\partial x_j} \langle u_i u_j \rangle + \frac{1}{\rho} \frac{\partial \langle p \rangle}{\partial x_j} - \nu \frac{\partial^2 \langle u_i \rangle}{\partial x_i \partial x_j} = 0, \quad (3.10a)$$

$$\frac{\partial \langle u_i \rangle}{\partial x_i} = 0. \quad (3.10b)$$

This form is known as the Reynolds-Averaged Navier-Stokes (RANS) equations. The unknown nonlinear term $\langle u_i u_j \rangle$ has appeared and is written in terms of $\langle \mathbf{u} \rangle$ and \mathbf{u}' :

$$\langle u_i u_j \rangle = \langle u_i \rangle \langle u_j \rangle + \langle u'_i u'_j \rangle. \quad (3.11)$$

3.3 Modelling The Fluid Equations

The second term on the right-hand side representing the coupling between the mean and fluctuating velocity is called the *Reynolds stress tensor*:

$$\tau_{ij} = -\langle u'_i u'_j \rangle, \quad (3.12)$$

where the negative sign is a convention adopted to signify that the Reynolds stress tensor dissipates energy from the flow in the same manner as the viscosity term (fourth term on the left-hand side of (3.10a)).

The anisotropic or *deviatoric* part of the Reynolds stresses, which contributes to momentum transport, is

$$\tau_{ij}^R = -\langle u'_i u'_j \rangle + \frac{1}{3} \langle u_i u_i \rangle \delta_{ij}, \quad (3.13)$$

where δ_{ij} is the Kronecker delta with the property $\delta_{ij} = 0$ when $i \neq j$ and $\delta_{ij} = 1$ when $i = j$. The isotropic or *spherical* part does not contribute to momentum transport, behaving instead like an additional pressure term. Therefore, it is included in a modified pressure \tilde{p} :

$$\tilde{p} = p + \frac{1}{3} \delta_{ij} \tau_{ii} = p + \frac{2}{3} \delta_{ij} k, \quad (3.14)$$

where $k = \frac{1}{2} \langle u'_i u'_i \rangle$ is the turbulent kinetic energy. The significance of this is that we cannot calculate the true value of pressure in a simulation with an eddy-viscosity model, which may be a serious disadvantage in industrial CFD unless the true pressure can be reliably recovered from (3.14). A full discussion of the pressure modification is given in §5.1.6.5 in the context of LES models. An example of the problem of unknown pressure is found in Chapter 7.

With the spherical/deviatoric stress decomposition, the Reynolds-averaged momentum equation can now be written

$$\frac{\partial \langle u_i \rangle}{\partial t} + \langle u_j \rangle \frac{\partial \langle u_i \rangle}{\partial x_j} + \frac{1}{\rho} \frac{\partial \langle \tilde{p} \rangle}{\partial x_j} - \nu \frac{\partial^2 \langle u_i \rangle}{\partial x_j^2} - \frac{\partial \tau_{ij}^R}{\partial x_j} = 0. \quad (3.15)$$

In writing the equations for $\langle \mathbf{u} \rangle$ we require additional information in the form of a *closure* or *turbulence model* to account for the unknown deviatoric Reynolds stress τ_{ij}^R . Chapter 4 describes the class of RANS models for the Reynolds stress.

3.3.3 Filtered Decomposition

The statistical picture of turbulence is somewhat limited in its ability to describe small-scale dynamics. Most of the dynamics are consigned to the

Reynolds stresses, with the resolved fields representing only mean quantities. An alternative decomposition to (3.9) is to separate low frequencies and high frequencies, or equivalently large scales and small scales. The model equations thus derived are deterministic rather than statistical.

The *filtered decomposition* is obtained by applying a low-pass filter operator \mathcal{G} to the velocity:

$$\bar{\mathbf{u}}(\mathbf{x}, t) = \mathcal{G}(\mathbf{u}(\mathbf{x}, t)), \quad (3.16)$$

where the overbar $\overline{(\dots)}$ denotes a filtered variable. In effect, the filter performs the integration

$$\bar{\mathbf{u}}(\mathbf{x}, t) = \int_{\Omega} G(\mathbf{r}, \mathbf{x}) \mathbf{u}(\mathbf{x} - \mathbf{r}, t) d\mathbf{r}, \quad (3.17)$$

where G is the filter *kernel* (or shape) of \mathcal{G} and \mathbf{r} is a radial distance associated with the filter.

The fluctuating component is then defined in an analogous manner to the Reynolds decomposition:

$$\mathbf{u}'(\mathbf{x}, t) = \mathbf{u}(\mathbf{x}, t) - \bar{\mathbf{u}}(\mathbf{x}, t). \quad (3.18)$$

The salient difference to (3.8) is that $\bar{\mathbf{u}}(\mathbf{x}, t)$ is not a statistical mean like $\langle \mathbf{u} \rangle$, but contains a significant proportion of the frequency spectrum of \mathbf{u} up to some cutoff frequency κ_c (equivalently a lengthscale or *filter width* $\bar{\Delta}$). The filter has the effect of removing those scales of motion smaller than the filter width $\bar{\Delta}$, i.e. averaging the velocity locally in space. The two methods are similar, since by Taylor's frozen flow hypothesis it is reasonable to assume that the time and space dimensions are interchangeable, at least for small scales (Cenedese and Romano, 1991). Time averaging can then be thought of as analogous to a low-pass filter with $\bar{\Delta} \propto \langle \mathbf{u} \rangle / T$. Indeed, (3.8) can be rewritten as a filter operator (Pope, 2000):

$$\bar{f}(t) = \frac{1}{T} \int_{t-T}^t f(s) ds. \quad (3.19)$$

As a result, the overbar notation $\overline{(\dots)}$ henceforth denotes both mean and filtered variables to keep notation simple.

It is worth emphasising that the operations are not strictly equivalent and the separation of scales can be quite different. Time averaging operates on all wavenumbers in the spectrum at once (albeit to different degrees), i.e. it is non-local in spectral space. The effect of the low-pass filter can be local in spectral space and non-local in physical space, or vice-versa, or moderately non-local in both depending on its exact definition. Furthermore, the filter

3.3 Modelling The Fluid Equations

is usually an implicit part of the discretised equations of motion rather than an explicit operation, and its precise form is never known. These points are explained fully in §5.1.

Using the filtered decomposition, the filtered Navier-Stokes equations are written

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j} (\overline{u_i u_j}) + \frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_j} - \nu \frac{\partial^2 \bar{u}_i}{\partial x_j^2} = 0, \quad (3.20a)$$

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0. \quad (3.20b)$$

Unlike the Reynolds decomposition (3.9), here the nonlinear term $\overline{u_i u_j}$ does not have a unique decomposition. It can be written as a function of $\bar{\mathbf{u}}$ and \mathbf{u}' in the double decomposition:

$$\begin{aligned} \overline{u_i u_j} &= \overline{(\bar{u}_i + u'_i)(\bar{u}_j + u'_j)} \\ &= \overline{\bar{u}_i \bar{u}_j} + \overline{\bar{u}_i u'_j} + \overline{\bar{u}_j u'_i} + \overline{u'_i u'_j}. \end{aligned} \quad (3.21)$$

All the terms that are dependent on \mathbf{u}' are grouped together into a *sub-grid-scale* (SGS) or *sub-filter-scale* (SFS) stress tensor τ_{ij} :

$$\tau_{ij} = -C_{ij} - R_{ij} = \overline{\bar{u}_i \bar{u}_j} - \overline{u_i u_j} = -\overline{\bar{u}_i u'_j} - \overline{\bar{u}_j u'_i} - \overline{u'_i u'_j}, \quad (3.22)$$

where the cross tensor C_{ij} includes interactions between the filtered and fluctuating components and the Reynolds tensor (as in (3.12)) includes interactions amongst the fluctuating components:

$$C_{ij} = \overline{\bar{u}_i u'_j} + \overline{\bar{u}_j u'_i}, \quad (3.23)$$

$$R_{ij} = \overline{u'_i u'_j}. \quad (3.24)$$

Analogously with the Reynolds decomposition, the deviatoric part of the SFS tensor, τ_{ij}^R , is modelled while the spherical part, τ_{ii} , is implicitly added to the pressure, resulting in a modified pressure \tilde{p} given by (3.14). Now the filtered momentum equation is written

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j} (\overline{\bar{u}_i \bar{u}_j}) + \frac{1}{\rho} \frac{\partial \tilde{p}}{\partial x_j} - \nu \frac{\partial^2 \bar{u}_i}{\partial x_j^2} - \frac{\partial \tau_{ij}^R}{\partial x_j} = 0. \quad (3.25)$$

The term $\overline{\bar{u}_i \bar{u}_j}$ cannot be calculated directly because it requires a second filtering. As a remedy, Leonard (1974) proposed the triple decomposition:

$$\tau_{ij} = -L_{ij} - C_{ij} - R_{ij} = \overline{\bar{u}_i \bar{u}_j} - \overline{u_i u_j}, \quad (3.26)$$

where the *Leonard tensor* $L_{ij} = \overline{\overline{u_i u_j}} - \overline{u_i} \overline{u_j}$ includes interactions amongst the filtered scales². The triple decomposition is used in this thesis.

Now the filtered Navier-Stokes equations are written (with the filtered continuity equation (3.20b)):

$$\frac{\partial \overline{u}_i}{\partial t} + \overline{u}_j \frac{\partial \overline{u}_i}{\partial x_j} + \frac{1}{\rho} \frac{\partial \overline{p}}{\partial x_j} - \nu \frac{\partial^2 \overline{u}_i}{\partial x_j^2} - \frac{\partial \tau_{ij}^R}{\partial x_j} = 0, \quad (3.27)$$

which is identical in form to the Reynolds-averaged momentum equation (3.15), except that a filter has been used instead of a time average. Note that the triple decomposition allows the advection term (second term on the left-hand side of (3.27)) to be split, which allows this term to be specially treated in the time discretisation: \overline{u}_i and \overline{u}_j can be evaluated at different times (cf. §3.5.3).

The sub-grid tensor (3.26) has a different physical form to the Reynolds stress tensor (3.12) and a different class of model or closure is employed, namely the *sub-grid-scale* (SGS) or *subfilter-scale* (SFS) models (cf. §5.2) used in large eddy simulation (LES) (cf. §3.3.5.3). The acronyms SGS and SFS are used interchangeably, though SFS is preferred since it is the filter and not the grid which defines the separation of scales in (3.18).

3.3.4 Boussinesq Eddy-Viscosity Hypothesis

We have now derived model equations for the resolved scales $\overline{\mathbf{u}}$, \overline{p} containing fewer degrees of freedom than the original Navier-Stokes equations (3.1). The next problem is to define models for the missing information, either the Reynolds stresses or the SFS stresses. These two strands of modelling have much in common despite their separate origins.

The Boussinesq eddy-viscosity approximation is the basis for many commonly used turbulence models. It is extrapolated from the linear stress-strain relation for Newtonian fluids:

$$(\tau_{ij} + p\delta_{ij})/\rho = 2\nu S_{ij}, \quad (3.28)$$

$$S_{ij} = \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right), \quad (3.29)$$

where S_{ij} is the strain rate tensor and ν is the molecular kinematic viscosity. The assumption made is that the deviatoric part of the Reynolds stress (3.13)

²The form of the Leonard tensor here is slightly different than that used in the dynamic LES model cf. §5.2.2.

or SFS stress (3.26) is related to the mean or filtered velocity strain rate by an *eddy viscosity* ν_T :

$$\tau_{ij}^R = \nu_T \bar{S}_{ij}, \quad (3.30)$$

where \bar{S}_{ij} is the strain rate of filtered or mean velocity $\bar{\mathbf{u}}$. The analogous *gradient diffusion hypothesis* for the turbulent transport of a scalar φ states that the flux of φ is related to the mean scalar gradient by an eddy diffusivity ν_T/σ_T :

$$-\overline{u_i \varphi} = \frac{\nu_T}{\sigma_T} \frac{\partial \bar{\varphi}}{\partial x_i}, \quad (3.31)$$

where σ_T is the turbulent Prandtl number. (3.30) and (3.31) have the same form as the relationship between viscous stress and strain in a Newtonian fluid, (3.28). However, unlike molecular viscosity ν , the eddy viscosity (and diffusivity) is not a fluid property but a flow-dependent quantity. A model is required for ν_T which must involve some additional knowledge of the flow (Wilcox, 1998).

The Boussinesq hypothesis assumes that the strain rate tensor \bar{S}_{ij} and deviatoric stress tensor τ_{ij}^R are aligned, but as shown by DNS (cf. §3.3.5.1) studies (Clark et al., 1979; Liu et al., 1994), in many flows they are not aligned. Nor is the assumption of a scalar eddy viscosity borne out by experiments, for example in swirling flows (Weber et al., 1990) and strongly curved geometry (Bradshaw, 1973). However, in simple shear flows such as a mixing layer or channel flow it is approximately valid (Pope, 2000). The limitations of the eddy-viscosity hypothesis are examined in more detail in §5.1.6.

In the next section, a range of models for the Navier-Stokes equations is presented. They form a hierarchy of approximations to the full multiscale picture of turbulence.

3.3.5 Levels of Approximation

3.3.5.1 Direct numerical simulation (DNS)

Direct numerical simulation (DNS) is the solution of the discretised Navier-Stokes equations with all scales of motion between η and l_0 resolved and nothing modelled. The DNS solution is represented by the black line (u) in Figure 3.4. The number of degrees of freedom is given by the Reynolds number scaling relations above. The only approximations in DNS are those necessary for the discretisation of the Navier-Stokes equations on a mesh.

Each DNS solution is a single realisation of a turbulent flow but useful statistical information about all realisations can be obtained if the ergodicity

principle is invoked (cf. §3.3.2). DNS is used to examine fine structures of particular flows of academic interest and to provide reference data for models. It is currently too expensive to use in many problems of practical or industrial interest (both in terms of simulation time and computational hardware), such as those in complex geometry or at high Reynolds number, although these constraints are continuously becoming less restrictive. For example, Fischer et al. (2008) demonstrate that complex high-Re flows can be simulated by DNS using current leading-edge facilities.

3.3.5.2 Reynolds-Averaged Navier-Stokes (RANS)

The Reynolds-Averaged Navier-Stokes (RANS) method calculates the solution of (3.15). The fluctuating component is included by way of a turbulence model. RANS is computationally inexpensive since the degrees of freedom are vastly reduced compared to DNS. The solution is steady-state, i.e. it does not contain any information about transient flow effects which may be important in many engineering problems (U-RANS improves upon RANS by including some transience cf. §3.3.5.4). Nevertheless, RANS is the most popular modelling approach in industry and may be adequate for some purposes, for example in estimating normal plant operating conditions or running large numbers of ‘scoping’ calculations to assess the effect of changing variables (cf. §2.3.5). The separation of scales in RANS is shown schematically in Figure 3.4.

RANS models were the first turbulence models to gain widespread popularity and have been used in engineering for decades, thanks to their low computational requirements and predictable behaviour (Ferziger et al., 1999). Mesh resolution requirements are relatively low because fine turbulent scales do not have to be resolved: meshes are constrained purely by geometric accuracy requirements and the requirement to reduce numerical discretisation error below a given threshold (Ferziger et al., 1999).

Using the Boussinesq eddy-viscosity hypothesis the Reynolds-averaged momentum equation (3.15) is written

$$\bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial}{\partial x_j} \left[(\nu + \nu_T) \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right] + \frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} = 0, \quad (3.32)$$

where the time derivative has been dropped, resulting in a steady-state model. \bar{p} is a modified pressure term including the normal stresses (cf. §3.3.2). A model is required for the eddy viscosity in order to close (3.32). Chapter 4 discusses some basic RANS models in more detail.

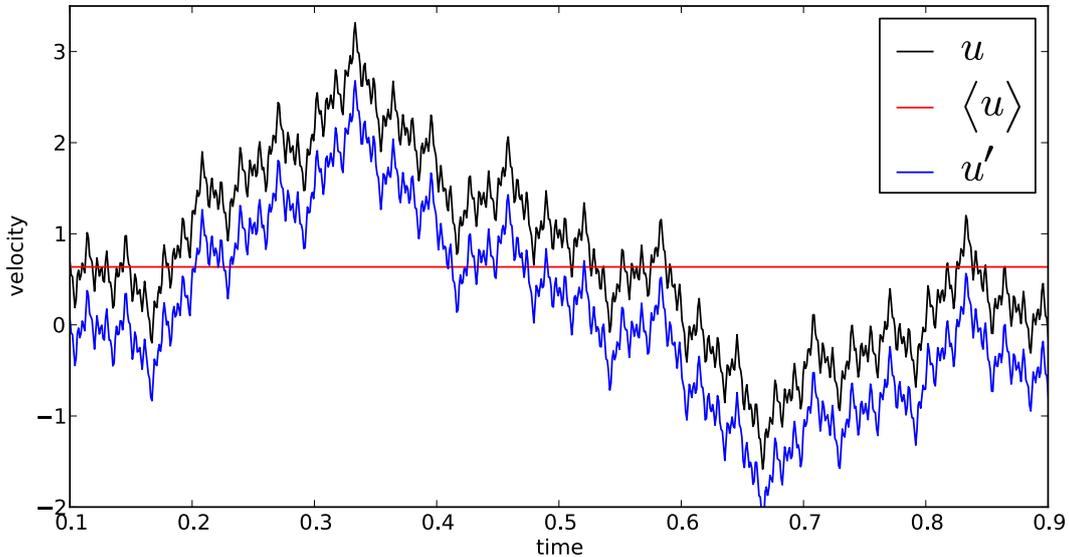


Figure 3.4: RANS velocity decomposition $u = \langle u \rangle + u'$. The resolved velocity $\langle u \rangle$ is constant, while the fluctuating velocity u' contains most of the scales of motion.

3.3.5.3 Large eddy Simulation (LES)

Large eddy simulation (LES) is based on the filtered momentum equation (3.27). LES is intermediate between RANS and DNS in terms of resolved detail: scales are resolved as long as they are above a certain size, determined by the filter width $\bar{\Delta}$. In the limit $\bar{\Delta} \rightarrow \eta$, LES becomes DNS. Scales that fall below the filter size are parameterised by an LES model using information from the resolved scales and theoretical or empirical arguments. An LES model for the deviatoric SFS stress tensor τ_{ij}^R in terms of the resolved scales is needed. Usually, LES models are based on the eddy-viscosity hypothesis (3.30). As for RANS models, only the deviatoric stresses τ_{ij}^R are modelled because they are the only components which contribute to momentum transport. §5.2 goes into detail on LES models. Figure 3.2 shows the theoretical split of LES into resolved and SFS in the energy spectrum, and the separation of scales in physical space is shown in Figure 3.5.

Many LES models are based upon the assumption that the SFS obey the local isotropy hypothesis. In practice this means that the filter width should be in the inertial subrange. A practical rule of thumb is that if 80% of the kinetic energy is in the resolved scales, the filter width is small enough and the LES is termed ‘well resolved’ (Pope, 2000). Models that are not based on the assumption of isotropic SFS may be able to break this rule, but they would have to be more sophisticated to account for anisotropic

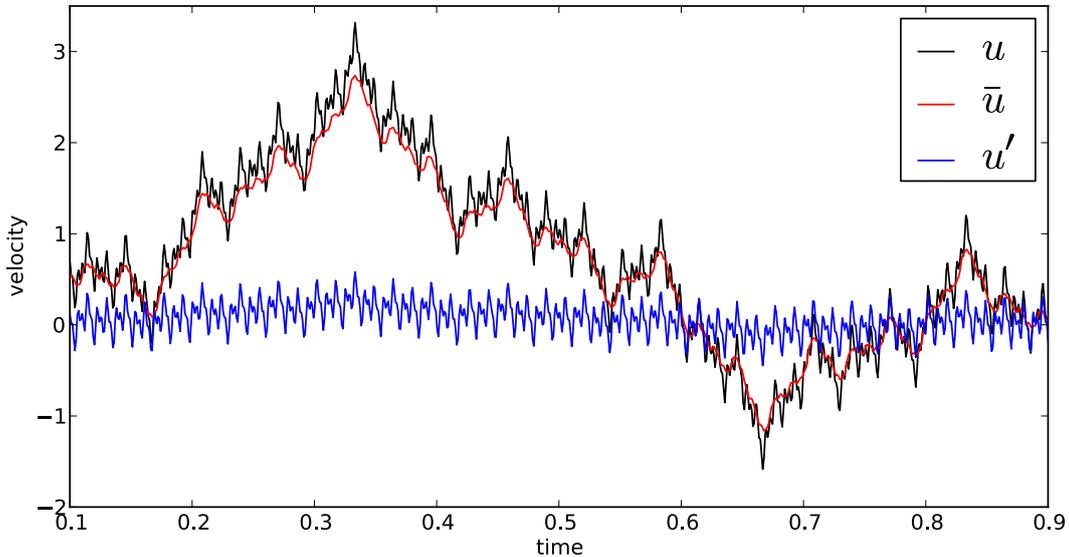


Figure 3.5: LES velocity decomposition $u = \bar{u} + u'$. The mean velocity \bar{u} is a reasonable representation of u and the SFS u' are relatively uniform and high-frequency.

or flow-dependent SFS in some way. In practice it is extremely difficult to determine whether a particular LES simulation is sufficiently resolved to meet the isotropy condition (cf. §5.2.2.5).

The filter size must satisfy the condition $\bar{\Delta} \geq \Delta$, where Δ is some metric of the local grid size (which requires careful definition on unstructured meshes; cf §5.1.6.3) (Berselli et al., 2006). It is common practice to link $\bar{\Delta}$ directly to Δ , which is why the SFS are often called the sub-grid (SGS) scales. If $\bar{\Delta}$ is spatially varying (inhomogeneous) then the filtering operation gives rise to *commutation errors* (Sagaut and Méneveau, 2006). If the mesh is anisotropic, then the SFS are anisotropic by construction and the local isotropy hypothesis is void. §5.1 contains a full discussion of filtering on anisotropic inhomogeneous meshes.

On a practical note, LES has an advantage over RANS in that it provides transient flow information. However, it can be prohibitively slow to solve because of the requirement to resolve all the scales of turbulence down to a point in the inertial range. As the financial cost of high performance computing decreases (cf. §2.4), LES meets these criteria at ever higher Reynolds numbers in more complex geometries. Nevertheless, the development of robust, accurate and fast LES models is crucial to meet the needs of industry. An interesting possibility is raised by performing LES on unstructured anisotropic inhomogeneous meshes: the SFS are anisotropic by construction, which *may*

make LES valid at larger grid spacing, provided that the LES model is suitably designed. Chapter 5 develops this idea further and Chapter 6 presents results of coarsely resolved LES of a 3D backward-facing step.

It has long been predicted that LES would replace RANS as the cost of computing power fell but RANS still holds considerable appeal to industrial users (cf. §2.3.5). Although advantages can be gained from simulating a problem in greater detail, there is also a need to simulate ever larger components, or even entire systems, and include extra physical effects such as fluid-structure interaction. The resulting increased complexity means that RANS is the only practical method for these problems (Menter, 2011; Pope, 1999).

3.3.5.4 Hybrid RANS/LES

A general class of methods have been developed in the last ten years which aim to smoothly bridge the gap between RANS and LES (Friess and Manceau, 2012). RANS and LES models are fundamentally different in their construction, being based on the statistical and filtered decompositions respectively, so there is a danger of inconsistency in attempts to smoothly transition from one to the other. For a complete review of these methods, see Fröhlich and von Terzi (2008) or Friess and Manceau (2012). Here attention is restricted to the generic scale-resolving characteristics of methods, which resolve some low-frequency modes or ‘coherent dynamics’ but are far from being true LES models. The separation of scales is shown schematically in Figure 3.6.

If the averaging operator is a phase average (3.19) with period T equal to the period of the largest scales, then the approximation is called unsteady Reynolds-Averaged Navier-Stokes (U-RANS). This can be achieved by including a time derivative in the Reynolds-averaged momentum equation (3.32). A long timestep Δt is used (cf. §3.5.3) so that high-frequency modes are not resolved, and the dissipation contributed by the RANS model might be reduced to allow transient dynamics to arise. When the flow exhibits periodicity of period $1/T$ due to deterministic forcing (e.g. a von Kármán instability, Figure 3.3) then this is a useful approach (Sagaut and Méneveau, 2006).

When the solution does not contain regular, predictable large scales, there is little consensus on a definition of the scale separation (Sagaut and Méneveau, 2006). In this case, a very large eddy simulation (VLES) approach may be appropriate. VLES adapts to the mesh resolution, becoming a DNS in the limit of $\Delta = \eta$, and RANS in the opposite limit of $\Delta \approx l_0$, while in between the model is intended to behave like an LES owing to the inclusion

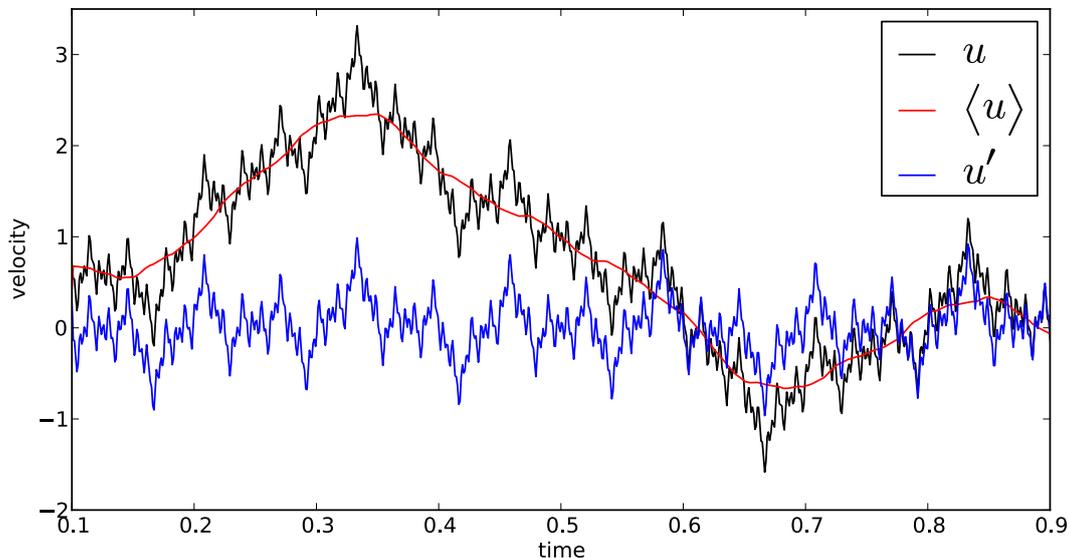


Figure 3.6: U-RANS or hybrid RANS/LES velocity decomposition $u = \langle u \rangle + u'$. The resolved velocity $\langle u \rangle$ encompasses the large coherent dynamics and the fluctuating velocity u' contains medium and small scales.

of the mesh size in the turbulence model (Han and Krajnović, 2012). This method was called the Flow Simulation Methodology (FSM) by Fröhlich and von Terzi (2008) in their comprehensive review of hybrid techniques, and was found to be effective in a range of flows.

In one of the earliest VLES models this is achieved by scaling the Reynolds stress tensor by a ‘filter’ or damping function based on the ratio of mesh size to Kolmogorov scale Δ/η (Speziale, 1997). Subsequent attempts have modified the damping function to make it more physically appropriate, e.g. Han and Krajnović (2012). The result is that the mean flow is damped to a greater or lesser degree by the modelled scales, depending on the ability of the mesh to represent a lesser or greater range of scales. It is claimed by Han and Krajnović (2012) that this method is analogous to controlling the proportion of resolved to total kinetic energy.

A very similar, but more established, strategy is detached eddy simulation (DES) (Spalart, 2009). DES uses a mesh refinement sensor to determine whether the mesh is fine enough to use LES. It is intended to reduce computational time by treating some parts of the domain with RANS, in particular the viscous sublayer (cf. §3.4.1.1), while LES is applied to the remainder to capture the significant transient flow structures (Spalart, 2009).

3.4 Turbulent Flow Boundary Conditions

The preceding discussion has considered only turbulent flow away from boundaries, where the Kolmogorov hypothesis is valid. Boundary conditions require special treatment and a separate class of models has been developed to do so. Broadly speaking, boundary conditions in simulations of turbulence are used to achieve two aims:

- To provide physically realistic bounding information for the problem,
- To reduce the computational effort of solving the problem.

The physics of turbulence at solid boundaries are complex in principle but modelling them statistically can be quite straightforward. Statistical modelling reduces the mesh resolution requirement near walls. An overview of near-wall physics and modelling is given in §3.4.1. Reduction of effort is achieved by specifying quasi-turbulent inflow conditions and recycling flow through outflow boundaries to speed up convergence to fully-developed turbulence, as discussed in §3.4.2.

3.4.1 Near-Wall Turbulent Dynamics

The preceding discussion of turbulence has been limited to high Reynolds number homogeneous isotropic turbulence. When a solid boundary is present, the flow is no longer homogeneous or isotropic and the picture of a one-way energy cascade from large to small structures is incomplete. Simply put, walls induce stresses in the fluid because fluid molecules stick to them, hence turbulence is generated. Instabilities are generated near the wall, growing into larger turbulent motions further from the wall and entering the free-stream flow by various complex anisotropic mechanisms (Pope, 2000). As a result, laminar flow over a plate at high Reynolds number will become turbulent as boundary layer instabilities grow downstream.

A vast amount of research has gone into near-wall turbulence. Even so, there is no equivalent of Kolmogorov's hypotheses here³. The interactions between the various dynamics are highly complex and a quantitative theory may be unattainable (Pope, 2000). However, some useful and important empirical results have emerged which are exploited as boundary conditions in simulations. Those that apply to fully-developed turbulent flow over a flat plate (*channel flow*), or flows reasonably close to that state, are described here.

³Proper Orthogonal Decomposition (POD) has had some success in elucidating the important processes in near-wall turbulence: see Holmes et al. (1993).

3.4.1.1 Turbulent Channel Flow

Consider a straight 3D channel of height 2δ , length $L \gg \delta$ and span $b \gg \delta$ containing fully-developed turbulent flow, i.e. in which the mean variables do not vary in the streamwise direction ($\frac{\partial \bar{u}}{\partial x} = 0$). The flow is assumed to be statistically independent of the spanwise direction ($\frac{\partial \bar{u}}{\partial z} = 0$), so the only nonzero mean gradient is normal to the wall ($\frac{\partial \bar{u}}{\partial y} \neq 0$). The total fluid stress is equal to the sum of the viscous and Reynolds or SFS stresses in one of the Navier-Stokes models previously described. The no-slip condition at the wall ($\mathbf{u} = 0$ at $y = 0$) dictates that $u'_i u'_j = 0$, therefore the Reynolds stresses at the wall are zero and the total stress at the wall, or *wall shear stress*, is only due to viscous stress:

$$\tau_w = \rho \nu \left(\frac{d\bar{u}}{dy} \right)_{y=0}. \quad (3.33)$$

From this we define characteristic near-wall quantities, the friction velocity u_τ and viscous lengthscale δ_ν :

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}}, \quad (3.34)$$

$$\delta_\nu = \nu \sqrt{\frac{\rho}{\tau_w}} = \frac{\nu}{u_\tau}. \quad (3.35)$$

The distance from the wall y and mean streamwise velocity \bar{u} are normalised by the friction velocity, defining the distance and velocity in *wall units*:

$$y^+ = \frac{u_\tau y}{\nu}, \quad (3.36)$$

$$u^+ = \frac{\bar{u}}{u_\tau}. \quad (3.37)$$

For sufficiently large Re (i.e. far from the range of laminar-turbulent transition), the near-wall velocity profile $u^+ = f(y^+)$ is independent of Re and has a universal linear-logarithmic form given in Table 3.1. The viscous sublayer, closest to the wall, is characterised by a linear relationship between u^+ and y^+ . The *log law* region has a logarithmic relationship with empirically determined coefficients C_κ (the von Kármán constant) and B (smallest value of u^+) commonly taken as 0.41 and 5.2 respectively. In the buffer layer the relationship is not well defined, graduating from linear to logarithmic. The log law (or law of the wall) holds in various flows with moderate pressure (hence shear stress) gradients (Pope, 2000).

3.4 Turbulent Flow Boundary Conditions

Region	Extent	u^+
Viscous sublayer	$y^+ < 5$	y^+
Buffer layer	$5 < y^+ < 30$	$y^+ \rightarrow (1/C_\kappa) \ln y^+ + B$
Log law region	$30 < y^+ < 0.3\delta$	$(1/C_\kappa) \ln y^+ + B$

Table 3.1: Variation of velocity with wall distance in wall units including ‘law of the wall’.

3.4.1.2 Modelling Implications

Since the turbulence lengthscale l varies linearly with y , it is practically impossible to resolve all near-wall dynamics as the mesh would have to be extremely fine (Pope, 2000). The problem is exacerbated in high-Reynolds-number flows because the boundary layer thickness ratio δ_ν/δ varies inversely with Re. The relations in Table 3.1 can be used as near-wall sub-grid models or *wall functions* to specify the near-wall variation of mean velocity, allowing mesh size to be large in terms of y^+ . To complicate matters, the wall shear stress is unknown *a priori*, so y^+ must be estimated or found iteratively during a simulation. Various methods exist for this, see for example Bazilevs and Hughes (2007), Bredburg (2000).

Whilst it is sufficient to model only the mean near-wall velocity in RANS simulations, in LES the wall function must supply the instantaneous value of velocity (Pope, 2000). The assumption may be made that the instantaneous velocity satisfies the log law (Grötzbach, 1981) or a power law (Werner and Wengle, 1989). However, for flows featuring separation or reattachment, these assumptions have been found lacking (Pope, 2000).

Instead of calculating the near-wall variation of velocity, the Reynolds or SFS stress can be specified as a near-wall turbulence model. A simple near-wall turbulence model was proposed by Prandtl (1925) which expresses a turbulent viscosity in terms of a *mixing length* l_m :

$$\nu_T = l_m^2 \left| \frac{\partial \bar{u}}{\partial y} \right|. \quad (3.38)$$

The mixing length in wall units $l_m^+ = l_m/\delta_\nu$ is specified by the van Driest damping function (van Driest, 1956):

$$l_m^+ = \kappa y^+ [1 - \exp(-y^+/A^+)], \quad (3.39)$$

where $A^+ = 26$ is based on the value $B = 5.2$ in the log law (Pope, 2000). This is an alternative expression of the log law in Table 3.1 since at large y^+

the damping function (the term in square brackets) tends to one. Wall functions for RANS and LES, parameterising the law of the wall, are developed in §4.3.

3.4.2 Turbulent Inflow Conditions

Many industrial simulations are of a component into which flow enters from another part of the system. In the real plant this flow will have certain qualities imparted by the geometry and conditions upstream. In a simulation it may be crucial to recreate these qualities in order to accurately predict the flow in the component under scrutiny. The ‘brute force’ solution is to simulate the entire upstream system. For example, in a nuclear reactor primary loop (cf. §2.2) the coolant is driven through a pump which imparts the flow with characteristic velocity and vorticity profiles and dominant turbulent frequency modes. Pipework from the pump to the reactor has bends and restrictions which further modify the flow, increasing swirl.

To simulate all of this as well as the downstream geometry is unfeasible so we need an approximate model for the incoming flow. Several methods exist for providing realistic inlet conditions for turbulent simulations. In increasing order of sophistication (and computational effort), some of these are:

1. *Laminar (constant in time) mean profile*

In order to obtain fully-developed turbulent flow in a pipe from a laminar inflow, a length of approximately 110 diameters is needed. This method is unsuitable for high-Reynolds number flows because it still requires a large amount of upstream geometry to be included (Jarrin et al., 2006).

2. *Mean profile with superimposed random fluctuations*

Adding random fluctuations spreads energy uniformly across all wavenumbers with no spatial or temporal correlations. The turbulence created is short-lived, decaying quickly downstream (Aider and Danet, 2006; Klein et al., 2003). An improved version of this method was developed by Lee et al. (1992) and applied to DNS of laminar-to-turbulent transition over a flat plate.

3. *Synthetic turbulence*

The principle of this method is to input large coherent structures in order to seed turbulence downstream. The synthetic eddy method (SEM) (Jarrin et al., 2006) combines mean and Reynolds-stress profiles from

DNS data with a specified turbulence lengthscale profile $l(y)$ to recreate a statistical sample of fluctuations. Time-varying coherent structures are created. SEM can significantly reduce the length downstream of the inflow required for fully-developed turbulence compared to laminar inflow conditions (Jarrin et al., 2006). Moulinec et al. (2005) found that the mean and Reynolds stress profiles attained equilibrium in only two diameters (compared to 110 for laminar inflow) at $Re = 57400$, making SEM very useful for industrial problems.

4. *Precursor simulation*

The most accurate but computationally expensive method (short of a full upstream simulation) is to run a moderate-resolution simulation (sufficient to resolve the large scales) beforehand and interpolate the results onto the current mesh at every timestep (Lund et al., 1998).

3.4.2.1 Periodicity

Another way to reduce the problem size is possible if the domain has some translational symmetry, such as a pipe or repeated structure. The flow may exhibit periodicity induced by the periodic geometry, although care must be taken to ensure that this is true (Ameri and Bunker, 2000). The number of degrees of freedom can then be reduced by imposing periodic boundary conditions (Klein et al., 2003). In the pipe this means recycling the outflow back to the inflow. Fully-developed turbulence is generated in a relatively short section of pipe and a few advective cycles (one advective cycle is the time taken for a particle suspended in the mean flow to traverse the entire domain from inlet to outlet) (Lund et al., 1998). Rotational symmetry, such as in an array of turbine blades, can be exploited by simulating one blade and recycling the flow from one side to the other (Ameri and Bunker, 2000).

3.5 Discretising The Fluid Equations

Having selected a decomposition of the fluid equations and a level of approximation, the model equations are discretised. Questions of accuracy, stability, convergence and robustness of the discretised equations are considered. The continuous equations are approximated at discrete locations in space-time. However, as a result of the propagation of information differing in space and time⁴, they are generally treated separately; methods for both

⁴Information can propagate in only one direction in time, whereas in space it can propagate in any direction.

are described. In hyperbolic systems, such as advection-dominated transient flow, space-time coupling is necessary to stabilise some discretisations (Donea and Huerta, 2003). Stabilisation techniques, described in §3.5.6 have been developed to address this problem. For simplicity the advection-diffusion equations (3.4) and the Poisson equation are used to illustrate the concepts. The Navier-Stokes equations will be returned to once the fundamental concepts have been introduced.

3.5.1 Consistency, Stability and Convergence

A *consistent* discretisation is one for which the discrete equations become the continuous differential equations in the limit of zero mesh size Δ and timestep Δt . Whilst consistency is satisfied for most discretisations, *stability* is often conditional on some parameter (Donea and Huerta, 2003). *Convergence* of the discretised equations in this context means that the solution of the discrete equations converges asymptotically to some limiting solution as the mesh spacing and timestep are reduced. Alternatively, in the context of the RANS models which do not evolve in time (cf. §3.3.5.2), convergence applies to an iterative solution method, where the solution is considered converged when some error measure (the difference between successive approximations) falls below a specified tolerance.

The Lax Equivalence Theorem states that given a well-posed⁵ linear initial-value problem, and a consistent discrete approximation to it, stability is a necessary and sufficient condition for convergence of the discrete solution (Donea and Huerta, 2003). The theorem is very powerful, since it is much easier to demonstrate consistency and stability than convergence directly, though convergence is the ultimate measure of quality and trust in the solution. The theorem can only be rigorously applied to linear initial-value problems, whereas in fluid dynamics the governing equations are nonlinear and of the boundary-value or mixed initial/boundary-value type. Thus, stability by the theorem is not sufficient to show that a CFD model is convergent.

3.5.2 Finite Element Method

3.5.2.1 Preliminaries

The finite element method restricts the solution of the governing partial differential equations (PDEs) to certain function spaces which satisfy the

⁵Well-posed means that the solution develops in a continuous manner from the initial conditions.

3.5 Discretising The Fluid Equations

criterion:

$$f : \bar{\Omega} \rightarrow \mathbb{R}, \quad (3.40)$$

where $\bar{\Omega} = \Omega \cup \Gamma$ denotes the union of the domain with its boundary Γ . If the derivatives of f up to order m exist and are continuous in Ω then the functions are of class $C^m(\Omega)$. Sobolev function spaces $\mathcal{H}^m(\Omega)$ include those in which the square of every derivative up to and including the m -th derivative is also continuous in Ω . Finite element functions belong to a particular class of Sobolev space, the space of square-integrable functions $\mathcal{L}^2(\Omega) = \mathcal{H}^0(\Omega)$ because second-order PDEs contain second derivatives of the prognostic variable which must be representable in discrete form⁶. This space has the standard definitions of inner product $(.,.)$ and norm $\|.\|$ of two scalar functions u, v :

$$(u, v) = \int_{\Omega} uv d\Omega, \quad (3.41)$$

$$\|v\|_0 = (v, v)^{1/2}. \quad (3.42)$$

These definitions are extended to vector-valued functions as follows:

$$(\mathbf{u}, \mathbf{v}) = \int_{\Omega} \mathbf{u} \cdot \mathbf{v} d\Omega, \quad (3.43)$$

$$\|\mathbf{v}\|_0 = \left(\sum_{i=1}^d \|v_i\|_0^2 \right)^{1/2}, \quad (3.44)$$

where d is the spatial dimension. The Sobolev space $\mathcal{H}^1(\Omega)$ is defined by

$$\mathcal{H}^1(\Omega) = \left\{ v \in \mathcal{L}^2(\Omega) \mid \frac{\partial v}{\partial x_i} \in \mathcal{L}^2(\Omega), i = 1, \dots, d \right\}. \quad (3.45)$$

The subspace $\mathcal{H}_0^1(\Omega)$ in which the function and its first derivatives disappear on Γ is essential to the finite element method:

$$\mathcal{H}_0^1(\Omega) = \left\{ v \in \mathcal{H}^1(\Omega) \mid v = 0, \frac{\partial v}{\partial x_i} = 0, i = 1, \dots, d \text{ on } \Gamma \right\}. \quad (3.46)$$

We now restrict attention to finite-dimensional function spaces. First a discretised domain composed of n_{ele} non-overlapping subdomains or *elements* $\bar{\Omega}^e$ is defined:

$$\bar{\Omega} = \bigcup_{e=1}^{n_{ele}} \bar{\Omega}^e. \quad (3.47)$$

⁶Whilst it is possible to represent discontinuous functions (such as first derivatives of piecewise linear functions) by discrete approximations, the same cannot be said for undefined functions, such as the second derivative of a piecewise linear function.

Each element has a boundary Γ^e and a characteristic size h (equivalent to the Δ notation used throughout this thesis), where $\text{diam}(\Omega^e) \leq h$ for each element.

In the finite element method two sets of finite-dimensional function spaces are required: the *test* (or weighting) functions \mathcal{W}^h , where the superscript h denotes a discrete quantity, and the *trial* functions (or admissible solutions) $\mathcal{U}^h \equiv \mathcal{W}^h + \bar{u}_D$ (where $\bar{u}_D = u_D$ on Γ_D). The former set of functions vanish on the Dirichlet boundary Γ_D and the latter satisfy the Dirichlet boundary conditions:

$$\mathcal{W}^h = \{w \in \mathcal{H}^1(\Omega) \mid w|_{\Omega^e} \in \mathcal{P}_m(\Omega^e) \forall e \text{ and } w = 0 \text{ on } \Gamma_D\} \quad (3.48a)$$

$$\mathcal{U}^h = \{u \in \mathcal{H}^1(\Omega) \mid u|_{\Omega^e} \in \mathcal{P}_m(\Omega^e) \forall e \text{ and } u = u_D \text{ on } \Gamma_N\}, \quad (3.48b)$$

where \mathcal{P}_m is the finite element interpolation space which determines the convergence properties of the finite element method. Homogeneous Dirichlet boundary conditions are $u_D = 0$. A popular choice in two and three dimensions is P_1 , the set of polynomials of degree $m \leq 1$ (Donea and Huerta, 2003).

3.5.2.2 Weak or Variational Form

Discretisation of the PDE by the finite element method begins by writing the weak integral or variational form of the PDE. Taking just the diffusion and source terms of the conservation equation (3.4a) and setting the diffusivity constant to one to simplify analysis, we have the Poisson equation (keeping the same Dirichlet and Neumann boundary conditions):

$$-\nabla^2 \varphi = q_\varphi. \quad (3.49)$$

Multiplying by a test function w , integrating and applying Green's theorem to the left-hand side:

$$\begin{aligned} - \int_{\Omega} w \nabla^2 \varphi d\Omega &= \int_{\Omega} \nabla w \cdot \nabla \varphi d\Omega - \int_{\Gamma} w (\mathbf{n} \cdot \nabla \varphi) d\Gamma \\ &= \int_{\Omega} \nabla w \cdot \nabla \varphi d\Omega - \int_{\Gamma} w g_N d\Gamma. \end{aligned} \quad (3.50)$$

The advantage of this procedure is that the second derivative has been replaced by first derivatives, reducing the regularity requirements on our test and trial function spaces: now $\varphi, w \in \mathcal{H}^1(\Omega)$. Furthermore, the Neumann boundary condition (3.4d) has been naturally introduced. The Dirichlet boundary conditions (3.4c) are satisfied by an appropriate choice of \mathcal{U} . Now the Galerkin method restricts the problem to the finite-dimensional spaces previously defined and we arrive at the problem statement:

Find $\varphi^h \in \mathcal{U}^h$ such that

$$\int_{\Omega} \nabla w^h \cdot \nabla \varphi^h d\Omega = \int_{\Omega} w^h q_{\varphi^h} d\Omega + \int_{\Gamma_N} w^h g_N d\Gamma_N, \quad \forall w^h \in \mathcal{W}^h. \quad (3.51)$$

3.5.2.3 Shape Functions

Suppose that $\eta = \{1, 2, \dots, n_{nodes}\}$ is the set of global node numbers where n_{nodes} is the number of nodes in the mesh. $\eta_D \subset \eta$ is the set of nodes on the Dirichlet portion of the boundary. The number of equations in the system n_{eq} is therefore the size of $\eta \setminus \eta_D$ (total number of nodes minus the number of nodes on the Dirichlet boundary). u^h and w^h are sums over these two sets:

$$u^h = \sum_{j \in \eta \setminus \eta_D} N_j u_j + \sum_{j \in \eta_D} N_j u_D, \quad (3.52a)$$

$$w^h = \{N_j\}, \quad j \in \eta \setminus \eta_D, \quad (3.52b)$$

where $N_j(\mathbf{x})$ is a *shape function* associated with node j defining the variation of values within the element Ω^e . The shape functions are defined on a reference element in terms of the three local coordinate axes $[\xi, \zeta, \eta]$, for example a line of unit length in 1D (see Figure 3.7), or an equilateral triangle of unit side length in a 2D triangular mesh. They make up the finite element interpolation space \mathcal{P}_m .

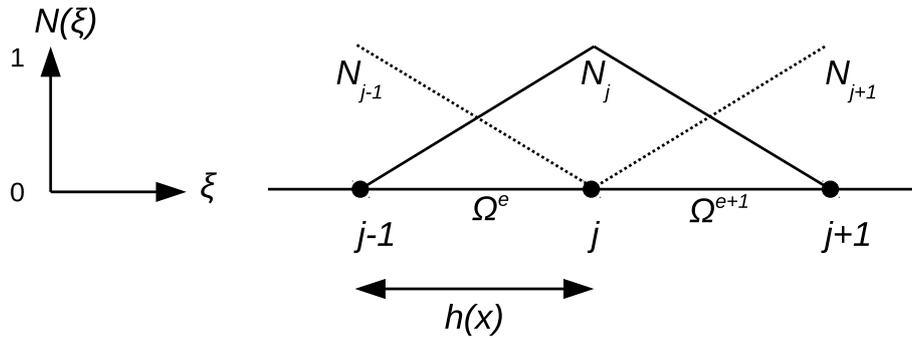


Figure 3.7: Linear shape functions in 1D. j is the node number, N_j is the shape function associated with node j and Ω^{e+1} is the line element spanning nodes j and $j+1$ with length h .

Linear shape functions in 1D are defined:

$$N_j(\xi) = \left\{ \begin{array}{ll} \frac{1}{2}(1 + \xi) & \text{in } \Omega^e, \\ \frac{1}{2}(1 - \xi) & \text{in } \Omega^{e+1} \end{array} \right\}, \quad \sum_{j=1}^{n_e} N_j = 1, \quad \text{for } \xi \in [-1, 1], \quad (3.53)$$

where n_e is the number of nodes in an element. Only functions which are continuous across element boundaries are considered for the moment; the resulting method is called the *continuous Galerkin* (CG) method. Use of linear continuous shape functions results in the P_1 CG discretisation.

The Poisson equation discretised by the Galerkin method (3.51) necessitated the choice of a function space for one variable, φ . The Navier-Stokes equations require the specification of separate function spaces for velocity and pressure. Many combinations are possible, for example piecewise quadratic continuous for velocity and piecewise linear continuous for pressure (P_2 CG – P_1 CG), piecewise linear discontinuous (cf. discontinuous Galerkin method, §3.5.6.7) for velocity and piecewise quadratic continuous for pressure (P_1 DG – P_2 CG), and so on. These have varying properties which render them useful for different kinds of flow problem, as explained in detail in (Pain et al., 2005).

3.5.2.4 Local Assembly

Using the definitions of test and trial functions (3.52), the terms of the weak finite-dimensional Poisson problem (3.51) can be written as the sum of integrals over an element Ω^e . The Neumann boundary condition term (3.4d) is written as an integral over a facet Γ_N^e of the element Ω^e if it is a portion of the domain boundary over which the Neumann condition applies. These integrals are the elemental matrices and vectors constituting the elemental matrix equation:

$$\mathbf{k}_{ij}^e \varphi_j^e = f_i^e + b_i^e, \quad (3.54)$$

where j denotes the node number in an element and

$$\mathbf{k}_{ij}^e = \int_{\Omega^e} \nabla N_i \cdot \nabla N_j \, d\Omega^e \quad (\text{diffusion matrix}), \quad (3.55a)$$

$$f_i^e = \int_{\Omega^e} N_i \, q_\varphi \, d\Omega^e \quad (\text{source vector}), \quad (3.55b)$$

$$b_i^e = \int_{\Gamma_N^e} N_i \, g_N \, d\Gamma_N^e \quad (\text{boundary vector}). \quad (3.55c)$$

Similarly, the finite element advection-diffusion problem based on (3.4) is

$$(\mathbf{a}_{ij}^e + \mathbf{k}_{ij}^e) \varphi_j^e = f_i^e + b_i^e, \quad (3.56)$$

where

$$\mathbf{a}_{ij}^e = \int_{\Omega^e} N_i \mathbf{u} \cdot \nabla N_j \, d\Omega^e \quad (\text{advection matrix}), \quad (3.57a)$$

$$\mathbf{k}_{ij}^e = - \int_{\Omega^e} \nabla N_i \cdot (\nu \nabla N_j) \, d\Omega^e \quad (\text{diffusion matrix}), \quad (3.57b)$$

$$\mathbf{f}_i^e = \int_{\Omega^e} N_i q_\varphi \, d\Omega^e \quad (\text{source vector}), \quad (3.57c)$$

$$\mathbf{b}_i^e = \int_{\Gamma_N^e} N_i g_N \, d\Gamma_N^e - \sum_{j=1}^{n_e} [\mathbf{a}_{ij}^e + \mathbf{k}_{ij}^e] \varphi_j^e \quad (\text{boundary vector}). \quad (3.57d)$$

3.5.2.5 Strong and Weak Dirichlet Boundary Conditions

The Dirichlet condition (3.4c) can be enforced *strongly* by specifying that $\varphi_j^e = \varphi_D(\mathbf{x}_j^e)$ if at node number j on the Dirichlet portion of the boundary and $\varphi_j^e = 0$ if not (Donea and Huerta, 2003). Thus, on the Dirichlet portion of the boundary, the boundary vector (3.57d) has a nonzero second term and vice versa.

Alternatively, the Dirichlet condition can be enforced *weakly* or in variational form by integrating the advection term by parts:

$$\mathbf{a}_{ij}^e = \int_{\Omega^e} \nabla N_i \cdot \mathbf{u} \varphi N_j \, d\Omega^e - \int_{\Gamma_D^e} N_i \mathbf{n} \cdot \mathbf{u} \varphi_D N_j \, d\Gamma_D^e. \quad (3.58)$$

When enforcing boundary conditions weakly, the discrete solution will not exactly satisfy the condition; rather, the solution converges to both the interior solution and the condition. The reason for this is that we consider the full set of test functions N_i , even though some of that set do not satisfy the boundary condition. Although strong imposition guarantees that the Dirichlet boundary condition is satisfied exactly, it does not guarantee faster convergence to the exact continuous solution than with weakly imposed boundary conditions. However, strongly imposed boundary conditions are sometimes required if the boundary condition needs to be satisfied exactly for physical reasons such as ensuring realistic values of temperature (AMCG, 2012).

3.5.2.6 Matrix Form of Navier-Stokes Equations

We are now ready to return to the arch-problem, the Navier-Stokes equations (3.1). The elemental matrix form of the momentum equation for a velocity component u is:

$$\mathbf{m}_{ij}^e \frac{du_j^e}{dt} + (\mathbf{a}_{ij}^e + \mathbf{k}_{ij}^e) u_j^e + \mathbf{c}_{ij}^e p_j^e = \mathbf{f}_i^e + \mathbf{b}_i^e, \quad (3.59)$$

where

$$m_{ij}^e = \int_{\Omega^e} N_i \cdot N_j \, d\Omega^e \quad (\text{mass matrix}), \quad (3.60a)$$

$$a_{ij}^e = \int_{\Omega^e} N_i \mathbf{u} \cdot \nabla N_j \, d\Omega^e \quad (\text{advection matrix}), \quad (3.60b)$$

$$k_{ij}^e = - \int_{\Omega^e} \nabla N_i \cdot (\nu \nabla N_j) \, d\Omega^e \quad (\text{diffusion matrix}), \quad (3.60c)$$

$$c_{ij}^e = \int_{\Omega^e} N_i \cdot \nabla p_j \, d\Omega^e \quad (\text{pressure gradient matrix}), \quad (3.60d)$$

$$f_i^e = \int_{\Omega^e} N_i f \, d\Omega^e \quad (\text{source vector}), \quad (3.60e)$$

$$b_i^e = \int_{\Gamma_N^e} N_i g_N \, d\Gamma_N^e - \sum_{j=1}^{n_e} [a_{ij}^e + k_{ij}^e] u_j^e \quad (\text{boundary vector}). \quad (3.60f)$$

Weakly applied boundary conditions on the velocity can be applied to the Navier-Stokes equations. In addition to simple Dirichlet and Neumann conditions, a whole class of wall functions are available for under-resolved turbulent flows which parameterise the near-wall variation of velocity in a turbulent boundary layer based on the relations in §3.4.1.1. Also known as ‘slip-with-friction’ boundary conditions, these can be applied weakly by integrating the viscous term (3.60c) by parts and specifying the viscous stress tensor on the boundary using empirical knowledge of universal properties of turbulent flows near solid walls.

In wall-bounded high-Re flows a large amount of vorticity (all the vorticity in a 2D flow driven purely by body force) is generated in the narrow region close to the wall where free-stream flow sharply transitions to the no-slip limit (Layton, 1999). This vorticity is transported into the free stream. If the near-wall region is under-resolved and the no-slip condition is strongly imposed, non-physical vortex structures may be generated (Layton, 1999). Weakly enforcing the no-slip boundary condition $\mathbf{u} = 0$ is equivalent to imposing a slip-with-friction condition and improves the physical behaviour at the wall in an under-resolved simulation (Bazilevs et al., 2007; Bazilevs and Hughes, 2007). In §4.3.3 a wall function in variational form is developed which improves accuracy in coarsely resolved near-wall turbulent flow.

3.5.2.7 Global Assembly

Assembly of the n_{eq} -dimensional global system of equations remains. The global system is constructed as the summation of the elemental matrices over the set of elements in Ω . The structure of the global matrices and vectors

depends on the topology of the mesh, i.e. the particular connections of each node to neighbouring nodes. The finite element method is naturally suited to unstructured meshes since it accommodates arbitrary topology. However, construction of the resultant system of equations is complicated by the fact that the matrices are not diagonal in general (Ferziger et al., 1999). By comparison, on structured meshes a node is always connected to the same number of nodes (for example Figure 3.9, §3.5.4), resulting in a diagonal matrix structure. The global matrix form of the steady advection-diffusion equation is

$$(A + K)\boldsymbol{\varphi} = \mathbf{F} + \mathbf{B}, \quad (3.61)$$

and the momentum equation is

$$M \frac{d\mathbf{u}}{dt} + (A + K)\mathbf{u} + C\mathbf{p} = \mathbf{F} + \mathbf{B}. \quad (3.62)$$

These equations are semi-discrete since we have not yet considered time.

3.5.3 Time Discretisation

Time is also approximated discretely since the solution cannot be found as a continuous function of time. The discretisation method for time is independent of the spatial discretisation method so this description applies to finite difference, finite volume and finite element methods. The time axis is split into a finite-dimensional series $t = t_0, t_1, \dots, t_n, \dots, T$, where the gap between time ‘nodes’ is the timestep $\Delta t(n) = t_{n+1} - t_n$. A scalar solution, denoted $\varphi^n = \varphi(t_n)$, is obtained by marching through time from some initial condition $\varphi(t = 0) = \varphi^0$ (and equivalently for velocity and pressure).

The solution method depends on the type of problem and level of approximation. Steady-state problems, such as steady laminar flow or fully-developed turbulent flow approximated by RANS, are often solved using a pseudo-time-marching method with a large timestep, using an implicit iterative scheme in which the equations are linearised (Zienkiewicz and Taylor, 1997). Unsteady problems, including turbulent flow approximated by DNS or LES, are solved using a time-marching approach with a smaller timestep determined by numerical stability constraints (Zienkiewicz and Taylor, 1997). For example, Fluidity employs a Picard iteration scheme which solves each unknown in a segregated manner using the best available solution for each variable (AMCG, 2012).

The size of Δt affects the stability of the solution. It is easy to see why this is so by considering a particle suspended in the flow. It will be advected a distance of $\mathbf{u}\Delta t$ in time Δt . If $\mathbf{u}\Delta t$ is bigger than the mesh size

Δ then instabilities can arise in the solution, depending on the particular discretisations of time and space and any stabilisation methods employed. Turbulent flow problems approximated by DNS or LES are *stiff*, meaning that there are a wide range of characteristic timescales associated with the dynamics. Thus both the timestep and mesh size are restricted if we wish to maintain stability (and resolve all resolvable dynamics).

A general method for discretising the time derivative is a weighted average of the values at the current and next timesteps called the *theta scheme*:

$$\frac{\varphi^{n+1} - \varphi^n}{\Delta t} = \theta\varphi^{n+1} + (1 - \theta)\varphi^n, \quad 0 < \theta < 1. \quad (3.63)$$

If $\theta = 0$ the method is called implicit; $\theta = 1$ gives the explicit method and $\theta = 0.5$ gives the midpoint or Crank-Nicolson method (Ferziger et al., 1999). The explicit and implicit methods are first-order accurate in time while the Crank-Nicolson method is second-order accurate (Donea and Huerta, 2003). Using the theta scheme, the semi-discrete form of the Navier-Stokes equations (3.62) are written:

$$\mathbf{M} \frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \theta(\mathbf{A} + \mathbf{K})\mathbf{u}^{n+1} = -(1 - \theta)(\mathbf{A} + \mathbf{K})\mathbf{u}^n - \mathbf{C}\mathbf{p} + \mathbf{F} + \mathbf{B}. \quad (3.64)$$

The advection matrix \mathbf{A} complicates time discretisation since it is nonlinear, i.e. $\mathbf{A} = \mathbf{A}(\mathbf{u})$. Treatment of this nonlinearity is particularly important when performing LES, since it is the primary mechanism of energy transfer from the resolved scales to the SFS and is the driver of the Kolmogorov energy cascade (Rollet-Miet et al., 1999). It cannot be treated fully implicitly and the velocity inside the matrix is often found using a Picard iteration:

$$\mathbf{u}^{n+1} \approx \theta_{nl}\mathbf{u}^{**} + (1 - \theta_{nl})\mathbf{u}^n, \quad (3.65)$$

where θ_{nl} is the nonlinear relaxation parameter and \mathbf{u}^{**} is the best available approximation to the current value, either the final value from the previous timestep if we are at the first iteration of a new timestep, or the previous nonlinear iteration at the current timestep.

Fluidity solves the Navier-Stokes equations in a segregated manner: in each nonlinear iteration, first the momentum equation is solved for an initial velocity guess \mathbf{u}^{**} with a pressure guess, followed by solving for a pressure correction δp from \mathbf{u}^{**} , and finally updating the velocity from \mathbf{u}^{**} to \mathbf{u}^{n+1} using the pressure correction. The process is repeated until the specified number of nonlinear iterations has been performed (generally two for LES). The reason that a pressure correction is needed is that the guess for velocity \mathbf{u}^{**} generally does not satisfy the continuity equation. It is achieved by

projecting the velocity to a space of divergence-free functions; however, then the velocity does not necessarily satisfy the momentum equation and the process must be repeated (Gresho and Chan, 1988).

3.5.4 Finite Volume Method

The finite volume method is equivalent to the discontinuous Galerkin method (cf. §3.5.6.7) with piecewise constant basis functions ($P_0 - DG$) (Wilson, 2009). A piecewise constant shape function has the values $N_A = 1$ at node A and $N_A = 0$ at the element boundaries, shown in Figure 3.8. Therefore, the shape functions are discontinuous from one element to another and shape function gradients cannot be evaluated across element boundaries; instead the gradients are written as fluxes across the element boundaries. This couples the elements together, since the fluxes depend on the values of the solution variables in adjacent elements.

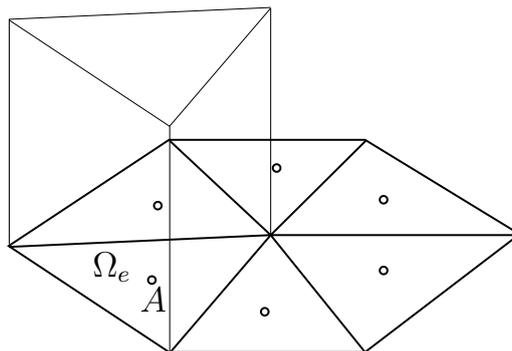


Figure 3.8: Piecewise constant shape function on a triangular mesh (Wilson, 2009).

The finite volume method has traditionally been implemented in Cartesian (structured) meshes and is easier to understand using the notation of that framework. Figure 3.9 shows a structured grid arrangement with the faces of the grey element Ω^P labelled n, s, e and w. In this context the elements are called *control volumes* (CV).

Using the advection-diffusion equation example, the variable φ_P is stored at the cell centre (black node) rather than the vertices, and surface fluxes are calculated at the hollow nodes. It is common to use a forward- or backward-difference approximation for the gradient of φ in the surface fluxes, leading to the following expression for φ at a surface node (for example φ_e on the

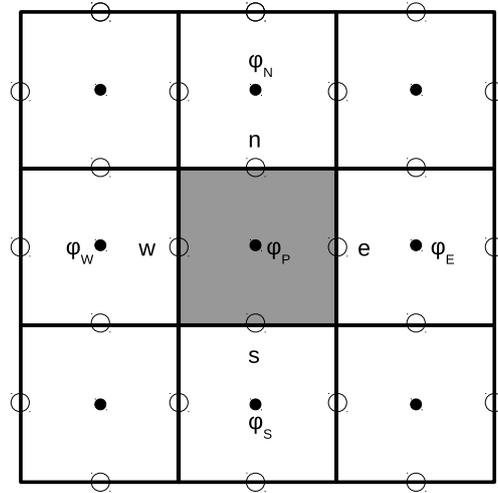


Figure 3.9: 2D structured quadrilateral finite volume mesh showing values of φ stored at cell centres and locations of fluxes on cell faces.

east face):

$$\varphi_e = \begin{cases} \varphi_P & \text{if } (\mathbf{u} \cdot \mathbf{n}) > 0, \\ \varphi_E & \text{if } (\mathbf{u} \cdot \mathbf{n}) < 0. \end{cases} \quad (3.66)$$

This is known as upwinding. It is non-oscillatory but also numerically diffusive owing to a truncation error of first order (Ferziger et al., 1999). Better approximations are given by linear interpolation:

$$\varphi_e = \varphi_E \lambda_e + \varphi_P (1 - \lambda_e), \quad (3.67)$$

where the interpolation factor $\lambda_e = (x_e - x_P)/(x_E - x_P)$. When $\lambda_e = 0.5$ and cell size is uniform, the scheme is a central-difference or trapezoidal approximation with a second-order truncation error, but it is not unconditionally stable (Ferziger et al., 1999).

3.5.5 Control-Volume Finite Element Method

The control-volume finite element method (CV-FE) is a hybrid of the finite volume and finite element methods which combines their advantages. It is applicable to unstructured meshes like the finite element method⁷ and retains the robustness of the control-volume finite difference method (Lin and Ebadian, 1999). Variation of the variables is described by linear shape functions in the elements of an unstructured ‘parent’ mesh, but their values

⁷The finite volume method can also be generalised to non-Cartesian meshes cf. Edwards (2002).

are stored at the element vertices. Control volumes are formed by joining the element centroids and face midpoints of the parent mesh with an element vertex at the centre as shown in Figure 3.10. Conservation equations are applied on these CVs with the surface fluxes and volume integrals calculated element-wise over the sub-volumes and sub-faces of each element portion in the CV.

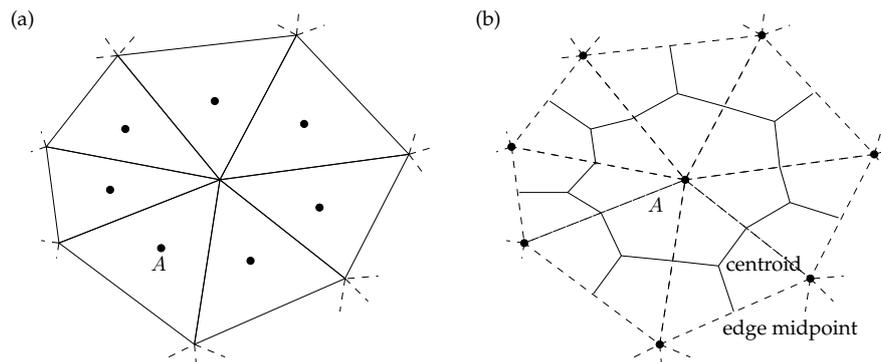


Figure 3.10: (a) 2D finite element mesh with centroids marked; (b) Dual control volume mesh formed from the centroids and edge midpoints of the parent mesh (AMCG, 2012).

As with the finite volume method there is a choice of advective flux schemes. In this case, as well as upwinding or the trapezoidal scheme, a finite element interpolation can be used in which the value of φ at a point on the CV surface is found by interpolating the shape functions of the parent mesh. Like the trapezoidal scheme it is not unconditionally stable and may require limiting. So-called *face-value limiting* can be achieved by estimating the upwind flux and limiting it; amongst many possible schemes the Sweby and Ultimate schemes are implemented in Fluidity (AMCG, 2012).

The diffusivity matrix is problematic when using a piecewise constant basis because the derivatives are discontinuous across element boundaries. An estimating scheme called the element gradient method, based on the gradients at the intersection of the CV boundary and parent mesh, can be used. Details of the CV-FE time discretisation and the implementation of the $k - \varepsilon$ turbulence model with CV-FE are given in §4.4.

3.5.6 Stabilised Finite Element Methods

Discretisation of the Navier-Stokes equations by the finite element method can encounter stability problems. Firstly, the choice of element pair for

velocity and pressure (cf. §3.5.2.3) affects stability according to the LBB stability condition. Secondly, the continuous Galerkin discretisation is unstable in advection-dominant flows. Methods for counteracting both of these problems are described. Advective stabilisations bear particular attention as they are viable alternatives to the LES models described in Chapter 5.

3.5.6.1 LBB Stability

According to the Ladyzhenskaya-Babuska-Brezzi (LBB) stability criterion, the pressure solution in the Navier-Stokes equations may contain spurious modes if equal-order basis functions are used for velocity and pressure, such as P_1 CG – P_1 CG. Stable discretisations according the LBB criterion are only those with a higher-order function space for velocity such as P_2 CG – P_1 CG (Donea and Huerta, 2003). However, P_1 CG – P_1 CG is a simple and convenient function space to use from an implementation point of view. It can be stabilised by adding a fourth-order term to the continuity equation with the dimensions of Δ^2 (Pain et al., 2005). Unfortunately, by doing so the continuity equation is no longer exactly satisfied, i.e. the velocity is not divergence-free. Pressure stabilisation is employed in Fluidity for the P_1 CG – P_1 CG element pair (AMCG, 2012).

In this thesis the equal-order interpolation, P_1 CG – P_1 CG, is adopted for simplicity of implementation as well as numerical considerations. This discretisation is often criticised in finite element research because it does not satisfy the LBB stability criterion and thus introduces spurious pressure modes, while the P_2 CG – P_1 CG discretisation (amongst others) does (Donea and Huerta, 2003). On the other hand, in a study by Rollet-Miet et al. (1999) it was found that P_1 CG – P_1 CG was the better choice for LES (compared to P_2 CG – P_1 CG) owing to its superior treatment of the velocity-pressure gradient correlation, which drives energy exchanges between velocity components. These exchanges are important in LES because they affect the small-scale flow dynamics, while spurious pressure modes are negligible on unstructured meshes Rollet-Miet et al. (1999).

Furthermore, in LES the pressure spectrum contains higher wavenumbers than the kinetic energy spectrum, and thus demands a higher-order basis to represent them. However, the LBB criterion demands a higher-order basis for velocity. Since neither can be satisfied, but representation of high frequency modes is essential, equal-order discretisation is the only sensible choice for LES with finite elements (Laurence, 2013).

3.5.6.2 Advective Stability

The second stability problem is that in advection-dominated problems the Galerkin method is only conditionally stable. A class of advective stabilisation techniques, similar in their effect to LES models, has been developed for this latter problem. Stabilised finite element methods and LES models can be thought of as examples of convergent evolution, having similar characteristics despite originating from different needs.

Discretising the advection-diffusion equation or momentum equation with the continuous Galerkin method leads to a truncation error equivalent to diffusion of magnitude

$$\bar{\nu} = \xi\nu\text{Pe}, \quad (3.68)$$

where

$$\xi = \frac{1}{\tanh(\text{Pe})} - \frac{1}{\text{Pe}}, \quad (3.69)$$

and the grid Péclet number Pe , characterising the relative strengths of advection and diffusion, is

$$\text{Pe} = \frac{\mathbf{u}\Delta}{2\nu}, \quad (3.70)$$

where Δ is the mesh size. Note that the overline notation does not refer to a filtered quantity as it did in §3.3.5.3. The truncation error $\bar{\nu}$ is an implicit negative diffusion term which outweighs the molecular viscosity when $\text{Pe} \geq 1$. Then the solution can become unstable (Donea and Huerta, 2003).

Besides refining the mesh (reducing Δ), two different but equivalent ways to counteract the instability are to balance the negative diffusion by adding dissipation of magnitude $\bar{\nu}$, or to modify the advection term to shift its weight towards the upwind direction (Donea and Huerta, 2003). A convenient way to apply upwinding is to use modified weighting (test) functions \bar{w} from a different class to the trial functions u . Such methods are collectively called Petrov-Galerkin methods. If the modified functions are applied to all terms in the discretised equation, the method is termed a consistent Petrov-Galerkin method because the solution of the original differential equation is also a solution of the discretised equation.

3.5.6.3 Streamline Upwind Method

The streamline upwind (SU) scheme adds numerical diffusion of magnitude (3.68) to balance the negative diffusivity of the Galerkin method. The method is equivalent to using modified test functions \bar{w} in the advection term only and as such it is not a consistent Petrov-Galerkin method. Figure 3.11 shows the modified advective test functions which shift the weight

towards the upstream direction. Note that they are discontinuous at the element boundaries, having support only on the element interiors.

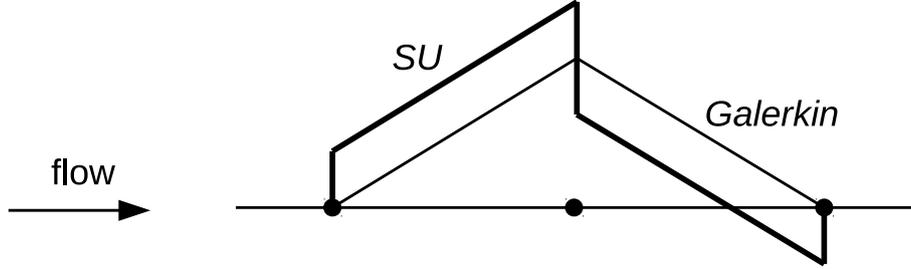


Figure 3.11: Piecewise linear and SU-modified weighting functions in 1D.

In multiple dimensions the weighting must be applied in the streamwise direction. This is achieved by replacing the scalar diffusivity $\bar{\nu}$ by a tensor (Hughes et al., 1979):

$$\bar{\nu}_{ij} = \bar{\nu} u_i u_j \|\mathbf{u}\|^2, \quad (3.71)$$

which is aligned with the flow direction and therefore does not add unwarranted transverse diffusion. The equivalent modified test function is then

$$\bar{w} = w + \frac{\bar{\nu}}{\|\mathbf{u}\|^2} (\mathbf{u} \cdot \nabla w). \quad (3.72)$$

The resulting finite element problem is like the Galerkin method with the following extra advection term calculated only on the element interiors:

$$\int_{\Omega} \frac{\bar{\nu}}{\|\mathbf{u}\|^2} (\mathbf{u} \cdot \nabla w) (\mathbf{u} \cdot \nabla \varphi) d\Omega. \quad (3.73)$$

This additional integral is like an LES model such as those based on the filtered decomposition (cf. §3.3.3). Whilst the SU method gives smooth solutions in simple cases, it can suffer from instability in more complex cases such as spatially varying source terms (Donea and Huerta, 2003).

3.5.6.4 Consistent Methods

Hughes and Brooks (1982) proposed using the modified test functions in all terms to obtain the consistent streamline upwind Petrov-Galerkin (SUPG) method. An asymmetric stabilisation term is derived from the residual of the equation multiplied by an SU-like diffusive stabilisation parameter. The SUPG method performs better than SU but determination of the optimal value of the stabilisation parameter is complicated by its asymmetry. The Galerkin least-squares (GLS) method remedies this with a symmetric stabilisation term. In both these methods the stabilisation vanishes with mesh size Δ , leading to a consistent discretisation.

3.5.6.5 Bubble Functions

Bubble functions are an enrichment of the space of basis functions which are defined on the element interiors but vanish on the element boundaries. The concept is equivalent to LES: the scales of motion are split into coarse-scale (resolvable on the computational mesh) and fine-scale (too small to resolve on the mesh) components: $\varphi = \bar{\varphi} + \varphi'$. The coarse-scale solution $\bar{\varphi}$ is discretised in a P_1 space and the fine scale solution φ' in a space of higher-order bubble functions defined only on the element interiors.

Initially, only quadratic enrichment of linear functions was considered and found to be excessively dissipative. They were also unable to resolve steep gradients inside elements (Candy, 2008). More general methods for enriching the space \mathcal{P}_m have been developed, for example the residual-free bubble method (Brezzi et al., 1998) or using Green's functions (Variational Multiscale Simulation, below) (Hughes, 1995) which can be applied to general multiscale problems.

3.5.6.6 Variational Multiscale Simulation (VMS)

In the variational multiscale simulation (VMS) the coarse scales are calculated numerically and the fine scales are analytically determined as in LES. Unlike LES, however, the two components may overlap or be disjoint and the fine scales can be defined locally or globally. Furthermore, instead of spatial filtering, the separation of scales is achieved by splitting the test function space: $\mathcal{W} = \bar{\mathcal{W}} \oplus \mathcal{W}'$. This provides a solid mathematical foundation for modelling turbulence and renders the approach ideal for complex geometry since there are no problems of filter commutativity on inhomogeneous meshes (Ramakrishnan and Collis, 2005).

VMS has been extended to non-local fine scales which are not confined to the element interiors by Codina and Blasco (2002). The model for the fine scales, which was originally conceived by Hughes (1995) as a simple algebraic model, has also been extended to turbulence modelling by Hughes et al. (2000) and Hughes et al. (2001). The local VMS framework of Ramakrishnan and Collis (2005) merges the Discontinuous Galerkin method (see below) with VMS.

3.5.6.7 Discontinuous Galerkin (DG)

The Discontinuous Galerkin (DG) method employs discontinuous shape functions so that the computational nodes are not shared between adjacent elements. Therefore, each element is an isolated problem and adjacent elements

are coupled via inter-element surface fluxes. Special treatment of shape function gradients on surfaces is necessary since the gradients are not defined there (Wilson, 2009). Stabilisation in the form of flux limiting is usually applied. DG is useful for its high-order accuracy on arbitrary grids, local hp -refinement ability and flexibility in using different models in adjoining elements (Arnold et al., 2002).

3.5.6.8 Implicit LES

Implicit LES (ILES) discretises the unmodified conservation laws and then uses the truncation error as an equivalent SFS dissipation term (Hickel et al., 2007). The problem is then to derive a physically realistic dissipation term. Smagorinsky-like models (cf. §5.2) were derived from numerical discretisations by Garnier et al. (1999), who also showed that low-order dissipative schemes have an equivalent Smagorinsky coefficient of $C_S = 0.2$, outweighing the standard physical form of the LES model where generally $C_S \leq 0.17$. The Monotonically Integrated LES (MILES) method, a type of ILES, introduces numerical dissipation via a combination of low- and high-order advective flux limiters (Boris et al., 1992).

3.6 Meshing and Optimisation

Two types of mesh topology have been mentioned in the preceding discussion. Structured meshes are defined as having constant connectivity, i.e. the number of edges meeting at a vertex is fixed; Cartesian structured meshes additionally have edges orthogonal to each other. Unstructured meshes are defined as having arbitrary connectivity and arbitrary edge lengths and directions. Traditionally structured meshes have been used in CFD, but when meshing curved surfaces or arbitrary geometry – which are the norm in industrial fluid problems – it can be exceedingly difficult to build a structured mesh. Even when the mesh is allowed to have non-orthogonal or curved edges or high aspect ratios, there remains the problem of variable refinement.

In order to resolve important flow details without wasting effort, it is sensible to refine the mesh locally. However, there is no simple way to progressively refine a structured mesh from one place to another because of the prescribed relationship of a CV to its neighbours (see Figure 3.9). A patch-refinement method is sometimes used in which the cells in a region of the mesh are subdivided. This leads to sudden jumps in edge length which can cause numerical errors such as shockwave reflection or commutation errors in LES models (Sagaut, 2006).

Unstructured meshing is a very practical alternative as it is easy to generate a mesh to fit an arbitrarily complex domain. Furthermore, sophisticated refinement methods are available. In the open-source CFD code Fluidity it is possible to efficiently optimise the mesh in response to the computed solution, resulting in a potentially minimal number of nodes representing the important dynamics. With this method of *unstructured mesh adaptivity* one does not need detailed knowledge of the flow and its evolution when designing the initial mesh. A description of unstructured mesh adaptivity follows.

Various mesh generation programs and adaptivity libraries have been used in this research: `GiD` (GiD, 2012) and `gmsh` (Geuzaine and Remacle, 2009) for mesh generation, the 3D mesh adaptivity library `libadaptivity` (Pain et al., 2001) for tetrahedra and the 2D mesh adaptivity library `libmba` (Vasilevskii and Lipnikov, 1999) for triangles.

3.6.1 Unstructured Mesh Adaptivity

Adaptivity is the generic term to describe adjusting the number and distribution of the degrees of freedom of a numerical method in order to reduce some solution error. Optimisation-based mesh adaptivity, implemented in Fluidity, seeks to optimise a mesh with respect to some measure of mesh quality or equivalently a measure of the error in the solution.

Mesh quality can be defined in different ways; here it is related to the size and shape (anisotropy) of each element. Quality of an element Δ is assessed by a functional $Q_M(\Delta)$ which has a maximum value when the element is a regular unit tetrahedron in 3D (or equilateral triangle in 2D) when measured with respect to a non-Euclidean metric \mathcal{M} . The regular unit tetrahedron in metric space is a reference or ‘ideal’ element in terms of the minimisation of solution error. Use of a non-Euclidean metric means that an ideal element in metric space is actually anisotropic in ‘real’ or physical space⁸.

Any deviation of element size or edge length aspect ratio from the ideal element is penalised. The quality functional of the entire mesh is the minimum value over all the elements (Piggott et al., 2009). Improvements in element quality are achieved by various geometric operations illustrated in Figure 3.12, namely node insertion/edge splitting, node deletion/edge collapse, edge swap and node movement.

⁸If instead a Euclidean metric were used then the quality optimisation procedure would return uniform elements in physical space.

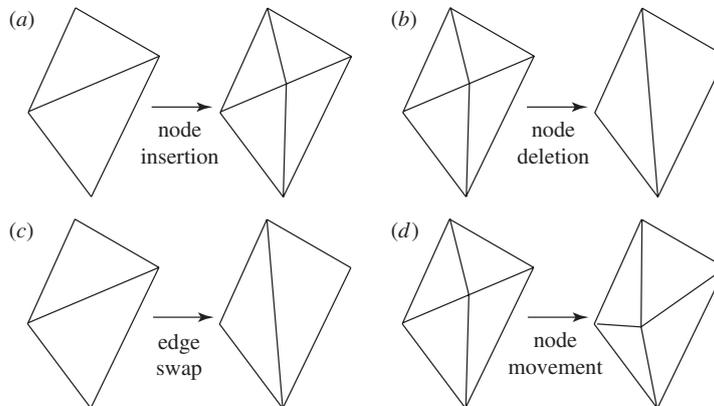


Figure 3.12: Local element operations to improve element quality $Q_M(\Delta)$. (a) node insertion or edge split, (b) node deletion or edge collapse, (c) edge swap, (d) node movement. (Courtesy of Piggott et al. (2009).)

3.6.1.1 Metric Formation

The metric \mathcal{M} is a measure of the solution error. The interpolation error of the solution is an ideal choice for defining \mathcal{M} as it bounds the error of finite element solutions to elliptic problems (Brenner and Scott, 2002). Even in non-elliptic problems it is an appropriate bound on the discretisation error (Alauzet et al., 2006). The exact interpolation error is not known unless *a priori* information is available. However, it is bounded by the product of the second derivatives (Hessian, or curvature) of the interpolant (solution) and the size of the region considered (the element in this case) (Piggott et al., 2009). An approximation to the interpolation error is thus given by the product of the Hessian H with the vector of element edge lengths, rendering the error estimate *a posteriori*. By this method, the mesh is refined in regions of high curvature of the interpolant (Farrell, 2009).

For a given interpolation error ε_u , we define the metric \mathcal{M} such that the element edges are of unit length with respect to \mathcal{M} if the element has the desired interpolation error ε_u , i.e.

$$\mathcal{M} = \frac{1}{\varepsilon_u} |H^e|, \quad (3.74)$$

where \mathcal{M} is symmetric positive-definite, H^e is an element-valued Hessian and the user-specified ε_u may be a spatially and temporally varying field (also referred to as the weight) (Piggott et al., 2009). Having calculated \mathcal{M} , it is decomposed into eigenvalues and eigenvectors:

$$\mathcal{M} = Q\Lambda Q^T, \quad (3.75)$$

where Λ is a diagonal matrix of eigenvalues and Q is an orthonormal matrix of eigenvectors. Geometrically Λ is a scaling and Q is a rotation. The eigenvalues λ_i represent the desired edge length h_i in the directions of the eigenvectors Q_i by the relationship

$$h_i = (\lambda_i)^{-1/2}. \quad (3.76)$$

Once the ideal element has been formed in metric space its edge lengths mapped back to physical space as shown in Figure 3.13.

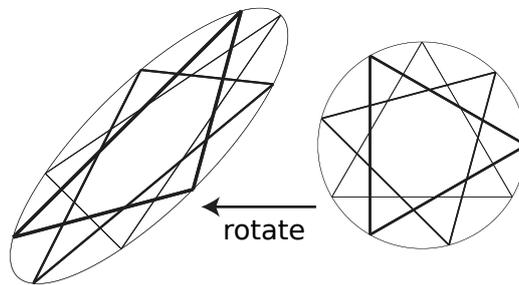


Figure 3.13: Mapping of an ideal element from metric space to an anisotropic element in physical space. (Courtesy of Piggott et al. (2009).)

3.6.1.2 Adapting to Multiple Fields and Constraining the Metric

The method described above forms a metric for a chosen scalar field (the interpolant). To form a metric for a vector field such as velocity, metrics are formed for each component and combined following the procedure developed in Pain et al. (2001). One metric is transformed to a sphere and the other metrics are transformed by the same operation. The metrics are then superimposed as shown in Figure 3.14. The largest ellipsoid which fits inside defines the combined metric, thus satisfying the most stringent interpolation error bounds on each field. Any number of fields can be combined in this way. In the results sections of this thesis, the fields are referred to as ‘target’ fields.

In practice it is necessary to constrain the optimisation, for example by the maximum number of mesh nodes and/or maximum and minimum edge lengths. A gradation algorithm can be applied to ensure sufficient smoothness in the mesh. Sudden changes in resolution could otherwise lead to undesirable effects such as reflecting pressure waves. These constraints are applied to the metric by modifying the eigenvalues as described in Piggott et al. (2009).

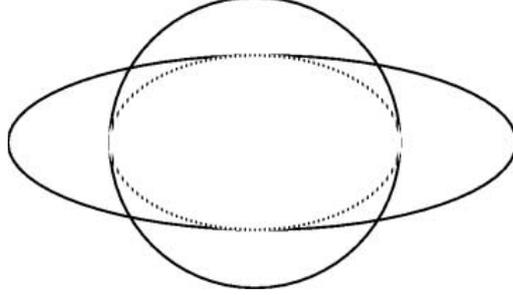


Figure 3.14: Metric superposition technique for adapting to error in multiple fields, showing two metrics and the combined metric. Individual metrics are solid lines and the combined metric is the dotted line. (Courtesy of Pain et al. (2001).)

3.6.1.3 Hessian Recovery

Hessian-based *a posteriori* interpolation error estimates are ideal for mesh adaptivity. However, considering a piecewise linear discretisation of the velocity, the second derivatives are zero in the element interior and undefined at the element boundaries because the first derivatives are piecewise discontinuous constant functions (Farrell, 2009). We have to recover the Hessian in some way from the information available. Pain et al. (2001) used a double-lumped Galerkin projection, which has the lowest error in the \mathcal{L}^∞ norm of any of the recovery methods analysed in Vallet et al. (2007):

$$\mathcal{H}_{ij}^e = \frac{1}{M_L^e} \int_{\Omega} \varphi_e \frac{\partial q_i}{\partial x_j} dV, \quad (3.77)$$

where e is the element under consideration, M_L^e is the lumped mass matrix, Ω is the domain and q_i is a piecewise linear projection of the first derivative given by

$$q_i = \frac{1}{M_L^e} \int_{\Omega} \varphi_e \frac{\partial \bar{u}}{\partial x_i} dV. \quad (3.78)$$

The resulting Hessian is made symmetric by averaging the off-diagonal entries. The mass matrix is lumped to reduce computational overheads. For other discretisations such as piecewise constant (P_0 -DG) or quadratic (P_2) the above procedure also holds.

3.6.2 Goal-Based Adaptivity and Adjoint Methods

Often in practical applications we ask:

If this is the answer we want, what are the initial and boundary conditions which give us this answer?

For example, what is the shape of the aerofoil which has the lowest drag? Where should the wind turbines be placed to extract the maximum amount of energy? This kind of ‘reversed’ questioning can be difficult to answer thanks to nonlinear physics and interactions between different phenomena. Turbulent flows are classic examples, e.g. the wake of one wind turbine affects the power output of other turbines downstream.

Conventional methods of solving these problems include sensitivity studies, mentioned in Chapter 2, by which the effects of changes to the initial and boundary conditions on the dependent variable are methodically assessed. Statistical techniques for reducing the number of trials required such as Design of Experiments (DoE) or Monte Carlo techniques are well-established.

Interpolation error-based mesh adaptivity is a form of reverse questioning, i.e. what is the topology of the mesh which results in the lowest interpolation error? However, in many applications, the interpolation error or energy norm does not necessarily provide useful bounds for errors in quantities of real physical interest (Becker and Rannacher, 2001). For example, Hill et al. (2012) found that the accuracy of ocean biology simulations varied depending on the choice of interpolant used for mesh adaptivity.

A major advance was made by Becker and Rannacher (2001) in the form of goal-based error estimation, in which the user selects a goal or functional \mathcal{J} based on the computed solution of the form

$$\mathcal{J}(\mathbf{u}) - \mathcal{J}(\mathbf{u}^h), \tag{3.79}$$

where \mathbf{u} is the exact solution and \mathbf{u}^h is a finite-dimensional Galerkin approximation. For example, one might choose \mathcal{J} as the integral of surface shear stress over a wing, in which case the functional of the (unknown) exact solution might be an experimentally determined value of wing drag. To solve this problem it is expressed as a linearised adjoint problem and run forwards and backwards in time over short periods: in effect, an iterative procedure for the optimal mesh to calculate the functional to a given accuracy (Farrell, 2009).

The goal-based optimisation framework determines the contribution of each element to the error in the functional. In other words, the spatial variation of the desired interpolation error ε_u is given to us by the method.

Therefore, it is ideally combined with mesh adaptivity in order to meet this error target. The method's appeal lies in its ability to both measure the amount of computational effort to achieve the goal and to save effort over an interpolation-error-based method (Farrell, 2009). Meeting the goal might not require the interpolation error to be minimised uniformly across the domain. Furthermore, the user might not know in advance where it should be minimised. Perhaps only a small region upstream of the wing needs to be resolved in order to calculate the correct drag. Fang et al. (2006) used goal-based adaptivity to provide optimal mesh resolution in 2D ocean simulations with sparse forcing data assimilated into the model. In this case the functional was composed of solution values at a set of locations and times.

In Chapter 5 these possibilities are explored in greater depth. Strategies for using interpolation error-based adaptivity with large eddy simulation are developed. The possibility of combining goal-based adaptivity with large eddy simulation is discussed. A method is outlined in which the goal functional is phrased in terms of the resolution of turbulence. In principle, achieving this goal demonstrates that a simulation was well-resolved and that its results can be trusted.

4 Weak Boundary Conditions for Turbulent Flow and Verification and Validation of the RANS $k - \varepsilon$ Model

This chapter moves on to implement a RANS turbulence model for use in industrial CFD problems. The aim is to investigate the properties of a standard model when combined with sophisticated discretisation, boundary and meshing techniques. The implementation, verification and validation of the oft-used standard $k - \varepsilon$ model discretised by the control volume-based finite element (CV-FE) method is described. A scale-adaptive version of the $k - \varepsilon$ model, the VLES model, is implemented to provide a bridge between RANS and LES in terms of turbulence resolution. A log-law wall function in variational form, suitable for any model on unstructured finite elements, is developed.

Contents

4.1	Introduction	93
4.2	RANS Model Formulation	95
4.2.1	Mixing-Length Model	95
4.2.2	One-Equation Model	95
4.2.3	$k - \varepsilon$ Model	96
4.2.4	$k - \omega$ Models	98
4.2.5	Further RANS Models	99
4.3	Boundary Conditions for Turbulent Flow	99
4.3.1	Wall Shear Stress Boundary Conditions	100
4.3.2	Improved Wall Functions	101
4.3.3	Weakly Enforced Boundary Conditions	102
4.3.4	Wall Functions for the $k - \varepsilon$ Model	103
4.4	Model Implementation	104
4.4.1	Discretisation	104
4.4.2	Solution Algorithm	106
4.4.3	$k - \varepsilon$ Model With Mesh Adaptivity	109
4.4.4	Scale-Adaptive $k - \varepsilon$ Model (VLES)	109
4.5	Verification	110
4.6	Validation	113
4.6.1	Simulation Setup	114
4.6.2	Results	115
4.7	Discussion	125

4.1 Introduction

In engineering applications the most popular models for the Reynolds stress tensor τ_{ij}^R in (3.15) are known as Reynolds-averaged Navier-Stokes (RANS) models based on the Boussinesq eddy-viscosity hypothesis (3.30) introduced in Chapter 3. This class of turbulence model is simple in formulation but limited in the level of detail representable in the solution. Advantages include rapid convergence to a steady-state solution and convergence to an ‘ideal’ model solution as mesh size is refined. RANS models are still popular for

industrial simulations as a quick way of establishing the performance of a fluid system.

A brief history of RANS models is presented with attention focused on two-equation closures. Several related two-equation models, all using the evolution of two quantities to derive an eddy viscosity, are described. The development of these models has been driven by the needs of engineers to simulate commonly occurring phenomena. For example, the $k-\omega$ SST model was developed to improve prediction of flow separation over an aerofoil at high stall angles (Menter, 2003). Limitations of these models and various attempts to fix them are also discussed.

A weakly applied velocity boundary condition for use with RANS or LES is described. Details of the implementation of the standard $k-\varepsilon$ model are given, including control volume-based finite-element discretisation method and solution algorithm. Of particular note is that Fluidity is a transient flow solver using a time-marching algorithm, therefore this implementation is categorised as unsteady RANS (U-RANS). Whether a simulation converges to a steady state or not may depend on the problem characteristics and the choice of timestep and time discretisation.

Results of verification tests using the method of manufactured solutions are presented followed by validation using the 2D backward-facing step problem. Validation efforts concentrate on the use of the standard $k-\varepsilon$ model with standard wall functions and the weak velocity boundary condition, as well as usage of the model with adaptive meshes. Results are compared against experimental data and numerical results published in Ilinca and Pelletier (1997) using a finite-element $k-\varepsilon$ model with mesh adaptivity. A scale-adaptive extension to the model which falls under the category of VLES methods is also described. This model is sensitive to the mesh resolution and allows a certain proportion of transient turbulent scales to be resolved, depending on the ability of the mesh to represent them. Its behaviour on a range of fixed meshes is assessed. Initial 3D results with the VLES model are presented in Chapter 6.

4.2 RANS Model Formulation

The evolution of the two-equation models most commonly used in industry is presented here. Simplifying assumptions and limitations are explained.

4.2.1 Mixing-Length Model

The simplest model for the eddy viscosity is the mixing length model, introduced by Prandtl (1925) for a 2D boundary-layer flow and modified by Smagorinsky (1963) to fit any flow:

$$\nu_T = l_m^2 \bar{S}_{ij}, \quad (4.1)$$

where l_m is the mixing length and \bar{S}_{ij} is the strain rate tensor defined for the resolved velocity $\bar{\mathbf{u}}$. Since it is an algebraic expression for the Reynolds stresses, i.e. not involving the solution of an equation, the mixing-length model is known as a zero-equation turbulence model. Specification of l_m is flow-dependent and usually requires *a priori* information about the flow, which limits its usefulness and accuracy (Pope, 2000). However, the model has been used extensively by Cebeci and Smith (1974) and Baldwin and Lomax (1978) in simulations of boundary layers over aerofoils for which l_m was well-known.

4.2.2 One-Equation Model

An improvement on the mixing-length method is obtained by replacing \bar{S}_{ij} in (4.1) with the turbulent kinetic energy, as suggested by Kolmogorov (1941) and Prandtl (1945):

$$\nu_T = ck^{1/2}l_m, \quad (4.2)$$

where $c \approx 0.55$ in order to yield the correct values in the log-law region. k is found by solving an advection-diffusion-reaction equation whose exact form is (Pope, 2000):

$$\frac{Dk}{Dt} \equiv \frac{\partial k}{\partial t} + \bar{u}_j \cdot \nabla k = \Pi - \varepsilon - \nabla \cdot \mathbf{T}, \quad (4.3)$$

$$\Pi = -\overline{u_i u_j} \cdot \nabla \bar{\mathbf{u}}, \quad (4.4)$$

$$T_i = \left[\frac{1}{2} \overline{u'_i u'_j u'_j} + \overline{u'_i p'} / \rho - \overline{\nu u'_j \left(\frac{\partial u'_i}{\partial x_j} + \frac{\partial u'_j}{\partial x_i} \right)} \right]. \quad (4.5)$$

This equation contains unknown terms – kinetic energy production Π , kinetic energy flux \mathbf{T} and dissipation rate of turbulent kinetic energy ε –

which have to be modelled. Π is a function of the Reynolds stress tensor $\tau_{ij} = -\overline{u_i u_j}$ and is modelled by invoking the eddy-viscosity hypothesis:

$$\Pi = (\tau_{ij} - \frac{1}{3}\tau_{ii}) \cdot \nabla \overline{\mathbf{u}} = \left[\nu_T \left(\nabla \overline{\mathbf{u}} + (\nabla \overline{\mathbf{u}})^T - \frac{2}{3}k\mathbf{I} \right) \right] \cdot \nabla \overline{\mathbf{u}}, \quad (4.6)$$

where \mathbf{I} is the identity matrix. The dissipation rate of kinetic energy ε is modelled as

$$\varepsilon = C_D k^{3/2} / l_m, \quad (4.7)$$

based on the observed dissipation rate in high-Reynolds-number flows, where $C_D \approx c^3$. The third term on the right-hand side is the kinetic energy flux \mathbf{T} , which includes transport of turbulence by itself and propagation by fluctuating pressure. It is modelled using the gradient-diffusion hypothesis (3.31) as

$$\mathbf{T} = -\frac{\nu_T}{\sigma_k} \nabla k, \quad (4.8)$$

where $\sigma_k = 1$. This assumption simply states that on average, k diffuses in the direction of the gradient of k . Wilcox (1998) presents comparisons of the one-equation model vs. the mixing-length model, demonstrating a slight advantage of the former. The problem with the one-equation model is that it still requires the mixing length to be specified.

4.2.3 $k - \varepsilon$ Model

To circumvent the problem of specifying mixing length l_m it is possible to solve a second equation for the dissipation rate ε , from which a different model lengthscale l can be derived by dimensional arguments:

$$l = k^{3/2} / \varepsilon, \quad (4.9)$$

which provides an estimate of the integral lengthscale of the flow. An eddy viscosity can also be derived:

$$\nu_T = C_\mu k^2 / \varepsilon = C_\mu k^{1/2} l, \quad (4.10)$$

where $C_\mu = c^4 = 0.09$.

It is important to define exactly what is meant by dissipation. For the purposes of turbulence modelling it is the energy flow rate in the turbulent cascade: a function of the large scales. However, the true physical meaning of dissipation is a sink for kinetic energy: a function of the dissipative (smallest) scales. An exact advection-diffusion equation exists for the latter definition but it is not an appropriate starting point for modelling the former (Pope,

2000). Instead an analogous form to (4.3) is used, bringing two benefits. Boundary conditions can be imposed on ε on the entire boundary and the gradient-diffusion form of the diffusivity term permits smoothly-varying solutions in inhomogeneous flows (Pope, 2000). Consequently, the two equations are written

$$\frac{\partial k}{\partial t} + \bar{u}_i \frac{\partial k}{\partial x_i} = \Pi - \varepsilon + \frac{\partial}{\partial x_i} \left[\left(\nu + \frac{\nu_T}{\sigma_k} \right) \frac{\partial k}{\partial x_i} \right], \quad (4.11a)$$

$$\frac{\partial \varepsilon}{\partial t} + \bar{u}_i \frac{\partial \varepsilon}{\partial x_i} = C_{\varepsilon 1} \frac{\varepsilon}{k} \Pi - C_{\varepsilon 2} \frac{\varepsilon^2}{k} + \frac{\partial}{\partial x_i} \left[\left(\nu + \frac{\nu_T}{\sigma_\varepsilon} \right) \frac{\partial \varepsilon}{\partial x_i} \right], \quad (4.11b)$$

where Π is given by (4.6). Coefficients C_μ , $C_{\varepsilon 1}$, $C_{\varepsilon 2}$, σ_ε and σ_k were empirically determined from examination of unbounded turbulent flows by Launder and Spalding (1974), and have the values 0.09, 1.44, 1.92, 1.3 and 1.0 respectively. These values were found by Launder and Spalding (1974) to be appropriate for plane jets and mixing layers although not for axisymmetric jets.

There is some disagreement over whether the diffusivity terms in (4.11) should contain just the eddy viscosity (scaled by a Prandtl number) or the sum of molecular and eddy viscosities. The former version is given by the original authors of the model (Launder and Spalding, 1974) as well as Pope (2000) and Ferziger et al. (1999) while the latter is given by Wilcox (1998), Lacasse et al. (2004), Lew et al. (2001) and Turgeon et al. (2000) among others. In this thesis the more common version containing both viscosities is used, motivated by numerical considerations: if the eddy viscosity is zero then we do not want a zero diffusivity term, otherwise the equations might become numerically unstable. Zero diffusivity is equivalent to an infinite Peclet number (cf. §3.5.6).

4.2.3.1 Limitations

The standard $k - \varepsilon$ model suffers from certain drawbacks. One particular problem is its behaviour near walls. The model does not correctly describe the variation of turbulent stress in the viscous sublayer of the boundary layer, so it should be used with wall functions to ensure correct scaling of the eddy viscosity (Mathieu and Scott, 2000). These are described in §4.3. The model often incorrectly predicts the mean flow in complex problems containing strong flow anisotropy because of the arbitrary form of the ε equation and the assumptions made by the eddy-viscosity hypothesis (cf. §3.3.4) (Pope, 2000). Some examples in which the model is known to be inaccurate are (NEA, 2007a):

1. impinging jets;
2. flow separation under strong adverse pressure gradients;
3. reattachment regions in separated flows;
4. strongly swirling flows/high streamline curvature;
5. buoyancy-driven flows;
6. laminar and transitional flows;
7. secondary flows (development of 3D motions from 2D mean flow);
8. jets with round cross-section.

Many modifications to the standard $k - \varepsilon$ model have been proposed over the years, which more or less successfully cure the shortcomings above, but none have been as widely used as the basic model (Pope, 2000). Alternative two-equation models may be used instead for particular classes of flow where the standard $k - \varepsilon$ model is inaccurate. Some of these are presented below for completeness, but in this thesis only the $k - \varepsilon$ model has been implemented and tested.

4.2.4 $k - \omega$ Models

The Wilcox $k - \omega$ Model simply replaces the dissipation rate ε with the turbulence frequency $\omega = k/\varepsilon$ in (4.11b), keeping the constants the same (Wilcox, 1998). Non-linear wall functions as used with the high-Re $k - \varepsilon$ model are not required because the typical near-wall resolution required is much larger ($y^+ < 2$ vs. $y^+ < 0.2$), although in industrial flows this still cannot be guaranteed, and so a wall treatment was developed (ANSYS, 2011). It is reported to be superior in boundary-layer flow in accounting for the viscous sublayer and adverse pressure gradients, but is sensitive to laminar free-stream boundary and inlet conditions for ω (Wilcox, 1998). The model also fails to predict the onset of separation from smooth surfaces because eddy viscosity is over-predicted near surfaces (Menter, 2011).

To improve upon these deficiencies, Menter (1994) developed the Baseline $k - \omega$ model, which blends between $k - \omega$ near surfaces and $k - \varepsilon$ in the outer flow using a blending function, and with an extra term in the ω equation to account for inhomogeneity. Equations for both ε and ω have to be solved. Despite being a combination of advantages of the two component models, the

baseline model fails to predict flow separation from smooth surfaces (ANSYS, 2011).

A further development of the $k - \omega$ model is the Shear Stress Transport (SST) model, which was specifically designed to predict the onset of flow separation from smooth surfaces by including the transport of turbulent shear stress (Menter, 1994). The eddy viscosity is limited by a different, more complicated, empirical blending function but the model is otherwise identical to the baseline model. Unlike the Wilcox and baseline $k - \omega$ models, it does not over-predict eddy viscosity near walls (ANSYS, 2011). Good predictions of flow separation under adverse pressure gradients were obtained by Bardina et al. (1997). Results are presented in Chapter 7 using this model in the commercial code CFX.

4.2.5 Further RANS Models

More sophisticated RANS models can be obtained by modelling the Reynolds stress τ_{ij}^R directly, rather than via an eddy viscosity. Simply called Reynolds stress models (RSM), they necessitate the solution of six or more additional equations for the components of the Reynolds stress. They are beyond the scope of this thesis, but are mentioned here for completeness.

4.3 Boundary Conditions for Turbulent Flow

To reduce computational requirements in wall-bounded turbulent flows, a class of near-wall turbulence models are usually applied. This section describes several near-wall treatments, or wall functions, for simulations of turbulent flow with RANS, VLES or LES. Many turbulence models have been designed for and calibrated to free-stream turbulence, i.e. in the absence of solid-wall effects. As a result, they do not ensure the correct variation of modelled quantities near walls so near-wall modifications or additional near-wall models are introduced (Wilcox, 1998). Furthermore, to resolve the viscous sublayer in high Reynolds number flow demands an extremely fine mesh; wall functions reduce the resolution requirements by prescribing the variation of velocity. The universal log law of the turbulent boundary layer in zero-pressure-gradient flow has been introduced in §3.4.1.1. This forms the basis of all the functions described here.

The notation of a 2D channel is adopted: the wall and flow are parallel to the x -direction and the wall normal is in the y direction. Streamwise velocity is the u component and wall-normal velocity is the v component. Since these wall functions only prescribe the streamwise velocity parallel to

the wall, they are intended to be used in combination with a no-normal-flow condition.

4.3.1 Wall Shear Stress Boundary Conditions

The idea of the wall function approach, proposed by [Lauder and Spalding \(1972\)](#), is to apply a boundary condition on velocity away from the wall with its value determined by the universal log law and wall distance (cf. §3.4.1.1). Then the model equations (RANS or LES) are not solved between the wall and the distance y_p from the wall of the node where the conditions are applied ([Pope, 2000](#)). This discussion applies to finite-difference and finite-volume methods where setting nodal values is appropriate. Wall functions for finite-element methods are discussed in §4.3.3. Setting the distance in wall units $y_p^+ \geq 30$ ensures that the log-law wall functions are valid, but to calculate y_p^+ one has to know the wall shear stress τ_w (3.33). Wall shear stress is not known *a priori* in any but the simplest of flows so the mesh may have to be corrected following an initial estimation run, unless a scalable method is employed (see §4.3.2).

In the log law region of a high Reynolds number zero-pressure-gradient boundary layer ($30 < y^+ < 0.3\delta$ in a channel of width 2δ (cf. §3.4.1.1)) it can be shown that the momentum equation (3.32) neglecting viscous, pressure and advection terms in the log-law region reduces to:

$$\frac{\partial}{\partial y} \left(\nu_T \frac{\partial u}{\partial y} \right) = 0, \quad (4.12)$$

leading to the solution

$$u = u_\tau \left[\frac{1}{C_\kappa} \ln(u_\tau y / \nu) + B \right], \quad (4.13)$$

where B is a model constant, y is the distance to the wall, u_τ is the friction velocity given by (3.34), u is the velocity parallel to the wall and $C_\kappa = 0.43$ is Von Kármán's constant ([Wilcox, 1998](#)). The expression for mean velocity is the 'law of the wall' introduced in §3.4.1.1. However, the friction velocity depends on the shear stress, which depends on the velocity gradient, creating a circular dependency.

A simple way of implementing these wall functions is given by [Pope \(2000\)](#). Rather than enforcing (4.13) as a Dirichlet condition on the velocity, the shear stress itself can be enforced, escaping the circular dependency. A nominal friction velocity is obtained from the kinetic energy instead of the shear stress, which avoids having to iterate for the shear stress value:

$$u_\tau^* = C_\mu^{1/4} k_p^{1/2}, \quad (4.14)$$

where the subscript p denotes evaluation at the distance y_p from the wall where the near-wall node lies. A nominal non-dimensional wall distance is given by $(y_p^+)^* = y_p u_\tau^* / \nu$ and the nominal mean velocity u_p^* is then given by (4.13) with u_τ^* inserted (Pope, 2000). Now the wall shear stress is found from

$$(\tau_w)_p = (u_\tau^*)^2 \frac{u_p}{u_p^*}. \quad (4.15)$$

This is applied in the modelled equations as a near-wall turbulence model: the Reynolds stress (cf. RANS) or subgrid stress (cf. LES) tensor is replaced by this form at the wall-adjacent nodes.

An alternative solution to the circular dependency is to estimate wall shear stress iteratively during a timestep, e.g. Bazilevs et al. (2007). Once a converged value of τ_w is reached, it is incorporated in the mean velocity and additional fields.

4.3.2 Improved Wall Functions

The wall function (4.13) is invalid when the near-wall resolution places the first node at $y_p^+ \leq 30$ or in separated or reattaching flows (Ferziger et al., 1999). Modified versions have been proposed to improve results in some of these situations. For example, the scalable wall functions (Grotjans and Menter, 1998) limit the $(y_p^+)^*$ value to $(y_p^+)^* = \max((y_p^+)^*, 11.06)$. 11.06 is the intersection between the logarithmic and the linear near wall profile. Therefore, scalable wall functions are only applied outside the viscous sublayer, avoiding the fine mesh inconsistencies.

Any significant physical effects can be incorporated into modified wall functions. Wilcox (1989) modified (4.13) to include the effect of pressure gradients by retaining the gradient term in the simplified boundary layer momentum equation (4.12). As observed by Lacasse et al. (2004), the standard wall functions predict zero wall shear stress at reattachment and therefore zero heat transfer, which is incorrect: experimental observations have shown that heat transfer is actually at a maximum at reattachment points (Ignat et al., 1998). To correctly account for this effect, Hutton and Szczypura (1987) proposed ‘two-velocity-scale’ wall functions. Deardorff (1970) and Schumann (1975) proposed a slightly different model to the log law for the velocity, modified to incorporate wall roughness:

$$\bar{u} = \frac{\bar{\tau}_w^{1/2}}{C_\kappa} \left(\log \left(\frac{\Delta y}{2} E \right) - 1 \right), \quad (4.16)$$

where $\bar{\tau}_w$ is the time mean wall shear stress, Δy is the wall-normal distance of the first off-wall node and E is the dimensionless wall roughness.

A useful approximation employed by [Schumann \(1975\)](#) is to relate time-mean to instantaneous shear stresses and velocities:

$$\frac{\tau_w}{u} = \frac{\bar{\tau}_w}{\bar{u}}. \quad (4.17)$$

This allows the mean wall shear stress (4.16) to be estimated from the instantaneous flow field. The approximate mean wall shear stress is applied as a boundary condition by writing the weak form of the momentum equation containing an integral of surface stress. More details on this method are given in §4.3.3.

4.3.3 Weakly Enforced Boundary Conditions

A method of wall functions is proposed here, which imposes a weak boundary condition for shear stress in the finite-element method where setting nodal values is not appropriate. It has been implemented in Fluidity for use with the $k-\varepsilon$, VLES and LES models. The benefits of weak imposition of boundary conditions have been discussed in §3.5.2.5 and §3.5.2.6. The shear stress is extracted from the above expression of the log law of the wall, (4.17), given by [Schumann \(1975\)](#). Using Green's Theorem for a computational domain Ω with solid walls Γ , the viscous term in (3.60) becomes:

$$\int_{\Omega} N \nabla \cdot \tau d\Omega = - \int_{\Omega} \nabla N \cdot \tau d\Omega + \int_{\Gamma} N \mathbf{n} \cdot \tau d\Gamma, \quad (4.18)$$

where N is the finite-element shape function, τ is the instantaneous stress tensor and \mathbf{n} is the unit normal to the wall. (4.18) can be rewritten using (4.16) and (4.17) and $\tau_w = \mathbf{n} \cdot \tau$:

$$\int_{\Omega} N \nabla \cdot \tau d\Omega = - \int_{\Omega} \nabla N \cdot \tau d\Omega + \int_{\Gamma} N \frac{C_{\kappa}^2 u^2}{(\log(\frac{\Delta y}{2} E) - 1)^2} d\Gamma. \quad (4.19)$$

Defining

$$q = \frac{C_{\kappa}^2}{(\log(\frac{\Delta y}{2} E) - 1)^2}, \quad (4.20)$$

(4.19) is written

$$\int_{\Omega} N \nabla \cdot \tau d\Omega = - \int_{\Omega} \nabla N \cdot \tau d\Omega + \int_{\Gamma} N q u^{**} u^{n+1} d\Gamma, \quad (4.21)$$

where u^{**} is the most recent approximation to the streamwise velocity component and u^{n+1} is the streamwise velocity component at the next time step.

Averaging over time steps in this manner linearises the term and improves stability. It is a similar concept to the Picard iteration employed in Fluidity’s solution algorithm (cf. §3.5.3), which iterates using the most recent value of a variable (Ford et al., 2004).

The second term on the right-hand side of (4.21) called a *penalty term* is a weakly enforced shear stress boundary condition. In practice it is combined with a weak no-normal-flow boundary condition, obtained by adding a boundary term to the discretised continuity equation:

$$\int_{\Omega} M \nabla \cdot u dV - \int_{\Gamma} M n \cdot u d\Gamma = 0, \quad (4.22)$$

where M is the pressure shape function.

4.3.4 Wall Functions for the $k - \varepsilon$ Model

A description of the standard wall functions for the $k - \varepsilon$ model has been postponed until this point so that the wall functions for velocity could be introduced. The $k - \varepsilon$ model in the form above (§4.2.3) is only valid for fully turbulent flows, i.e. high turbulence Reynolds number $Re_t \equiv k^2/\varepsilon\nu$ and dominance of inertial forces over viscous forces. Near walls the turbulence Reynolds number is small, viscous forces become more significant, tending towards a laminar regime, and the model over-predicts eddy viscosity (Launder and Spalding, 1974).

The following ‘standard high-Re’ wall functions are commonly used as Dirichlet boundary conditions for the $k - \varepsilon$ model (Ferziger et al., 1999) and have been implemented in Fluidity. Wall shear stress is specified by (4.17), providing a value of friction velocity u_{τ} for use here. In the log-law layer the k and ε equations can be simplified in the same manner as the boundary layer momentum equation (4.12), leading to the solutions:

$$k = \frac{u_{\tau}^2}{C_{\mu}^{1/2}}, \quad (4.23a)$$

$$\varepsilon = \frac{u_{\tau}^3}{C_{\kappa} y}. \quad (4.23b)$$

The value of ε , (4.23), follows from the fact that production and dissipation of kinetic energy balance in the log law region.

4.4 Model Implementation

The standard $k - \varepsilon$ model has been implemented in Fluidity using a CV-FE discretisation (cf. §3.5.5) and a time-marching algorithm which permits transient (U-RANS) solutions, both described here.

4.4.1 Discretisation

Conventionally, the RANS equations and $k - \varepsilon$ turbulence model have usually been implemented in finite-difference and finite-volume codes, perhaps because these methods are better-established amongst the CFD community. The emergence of staggered-grid techniques for velocity and pressure and upwind differencing for advection enabled stable simulation of high-Reynolds-number flows in finite-volume codes (Leschziner, 1989). Launder and Spalding (1974) reviewed several applications of finite-difference RANS modelling and developed their wall function technique within this framework, enabling wall function values to be set at nodes of the mesh. Such methods remain popular, for example Shih et al. (1995) used a conservative finite-volume procedure with the standard $k - \varepsilon$ model and a modified model with a new form of the ε equation to simulate a variety of flows including a 2D backward-facing step.

In the 1980s there was a shift towards finite-element and hybrid CV-FE simulations of turbulent flows, e.g. Patankar and Baliga (1983). Finite-element methods are simple to apply on unstructured meshes and they sometimes avoid the dissipative characteristics of upwind finite-volume discretisation (though upwind stabilisation such as SUPG is at times necessary in finite-element methods) (Donea and Huerta, 2003). Conversely, satisfying conservation of momentum in a finite-element method is more difficult than in the finite-volume method, which is derived from the conservation law expressed over a control volume (Schneider and Raw, 1987).

There are certain difficulties incorporating $k - \varepsilon$ -type models into finite-element methods (Leschziner, 1989). The $k - \varepsilon$ model typifies nonlinear advection-diffusion-reaction equations with strong reaction terms that destabilise the solution, leading to non-physical oscillations and negative values of kinetic energy. These models tend to be unstable when using the continuous Galerkin (CG) discretisation and have to be stabilised with, e.g. streamline upwind (SU) (Ilinca and Pelletier, 1997) or Petrov-Galerkin (SUPG) advective stabilisation methods (cf. §3.5.6) (Lew et al., 2001; Hachem, 2009). This is not ideal from a modelling point of view because turbulence models are akin to stabilisation methods and one would hope that an extra stabilisation method is unnecessary when using a turbulence model. Various other meth-

ods have been proposed to improve stability of the model itself (as opposed to the discretisation in general). For example, (Lacasse et al., 2004) solved the equations for the logarithm of k and ε ensuring positivity, although this method has the disadvantage that exponential terms appear in the equations (Hachem, 2009).

The control-volume-based finite-element method (CV-FE) has been briefly explained in §3.5.5. From an unstructured finite-element mesh, a control volume ‘dual’ mesh is constructed as shown in Figure 3.10. The CV-FE method retains the geometric flexibility of finite-element methods and the conservative formulation of control volume methods (Schneider and Raw, 1987). The CV-FE discretisation was employed by Lin and Ebadian (1999) for the $k - \varepsilon$ model for simulations of heated helical pipes.

A locally skewed upwinding scheme for CV-FE was developed for unstructured quadrilateral elements by Schneider and Raw (1987), which chooses the point at which upwind velocity is measured by taking element geometry into account and therefore does not retain the ‘false’ numerical diffusion of the conventional upwinding scheme while being second-order accurate. Flux limiting is a process of estimating whether surface fluxes will produce non-physical oscillations and, if so, limiting them so that the solution remains bounded. A new high-order flux-limited advection scheme was developed by Wilson (2009) for advection-dominated flow problems on tetrahedral elements, which retains the conservatism and stability of finite-volume method but lacks its numerical diffusion.

Control volume surface fluxes can be estimated in various ways from the underlying finite-element mesh; in this thesis they are found from a finite-element interpolation at the CV surface quadrature point, which is possible because the parent finite-element mesh and its dual CV mesh have collocated nodes (Wilson, 2009). The values may become unbounded, so the flux limiting method of Sweby (1984) is used, which is similar to the slope limiters in discontinuous Galerkin (DG) methods that limit inter-element fluxes. Upwind values are found by projecting the value from a point in the element just upwind of the current element, known as anisotropic limiting. It is only possible on simplex meshes, i.e. linear basis functions (AMCG, 2012). Time spent searching for upwind elements and quadrature can be saved by storing them in a matrix, although extra memory is needed for this operation.

With this CV-FE method the advection-diffusion-reaction equations in the $k - \varepsilon$ model are discretised using piecewise constant shape functions N_i on the dual control volume mesh. A CV is denoted Ω^e with faces $\partial\Omega^e$. The CV-discretised equations are defined by integrating by parts the advection and diffusion terms, so we write for a scalar φ , which can be either k or ε ,

and ignoring boundary conditions for simplicity:

$$\int_{\Omega^e} \frac{\partial \varphi}{\partial t} + \int_{\partial \Omega^e} [\mathbf{n} \cdot \mathbf{u} \varphi] - [\mathbf{n} \cdot \mathbf{K}^e \cdot \nabla \varphi] = \int_{\Omega^e} P_\varphi^e + R_\varphi^e, \quad (4.24)$$

where \mathbf{K} is the diffusion tensor, P_φ and R_φ are the production (source) and reaction (dissipation) terms, \mathbf{n} is the unit normal to a CV face and the square brackets $[\dots]$ denote fluxes across the CV faces. Since φ is discontinuous these flux terms do not have unique values, but the requirement for φ to be conserved prescribes that the fluxes between adjacent CVs are consistent with one another. Flux schemes therefore form a crucial part of the CV-FE method (Wilson, 2009). Note that unlike higher-order discretisations, the test function N_i can be dropped from the equation as it is one everywhere within the volume Ω^e . Terms involving the gradient of N_i or φ have been dropped because they are both constant functions, with the exception of the diffusion term. The boundary integral for the diffusivity \mathbf{K}^e is treated specially because of the need to define gradients on the CV surface. Here the element gradient method is applied, which takes advantage of the facts that where the CV boundaries intersect the parent element the parent FE basis functions are continuous and that the nodes are collocated. The gradients of N_i are estimated from the parent basis function gradients, similar to the standard finite-volume diffusion scheme (Ciarlet and Lions, 2000).

4.4.2 Solution Algorithm

Inserting the particular form of the production, dissipation and diffusion terms we obtain the CV equation for kinetic energy k :

$$\underbrace{\int_{\Omega^e} \frac{\partial k}{\partial t}}_{\text{evolution}} + \underbrace{\int_{\partial \Omega^e} [\mathbf{n} \cdot \mathbf{u} k]}_{\text{advection}} - \underbrace{\left[\mathbf{n} \cdot \left(\nabla N_i \cdot \left(\nu + \frac{\nu_T}{\sigma_k} \right) \nabla N_j \right) \cdot \nabla k \right]}_{\text{diffusion}} = \underbrace{\int_{\Omega^e} \nu_T \left(\nabla \bar{\mathbf{u}} + (\nabla \bar{\mathbf{u}})^T - \frac{2}{3} k \mathbf{I} \right) \cdot \nabla \bar{\mathbf{u}}}_{\text{production}} - \underbrace{\left(\frac{\varepsilon}{k} \right) k}_{\text{dissipation}}. \quad (4.25)$$

Similarly the CV equation for dissipation ε is written

$$\underbrace{\int_{\Omega^e} \frac{\partial \varepsilon}{\partial t}}_{\text{evolution}} + \underbrace{\int_{\partial \Omega^e} [\mathbf{n} \cdot \mathbf{u} \varepsilon]}_{\text{advection}} - \underbrace{\left[\mathbf{n} \cdot \left(\nabla N_i \cdot \left(\nu + \frac{\nu_T}{\sigma_\varepsilon} \right) \nabla N_j \right) \cdot \nabla \varepsilon \right]}_{\text{diffusion}} = \underbrace{\int_{\Omega^e} C_{\varepsilon 1} \left(\frac{\varepsilon}{k} \right) \nu_T \left(\nabla \bar{\mathbf{u}} + (\nabla \bar{\mathbf{u}})^T - \frac{2}{3} k \mathbf{I} \right) \cdot \nabla \bar{\mathbf{u}}}_{\text{production}} - \underbrace{C_{\varepsilon 2} \left(\frac{\varepsilon}{k} \right) \varepsilon}_{\text{dissipation}}. \quad (4.26)$$

4.4 Model Implementation

Note that the production and dissipation terms and the high-Reynolds-number boundary conditions (4.23) contain some or all of k , ε and $\bar{\mathbf{u}}$. As a result, the equations are nonlinear and coupled and like the advection term the production, dissipation and boundary terms need special treatment when it comes to time discretisation.

In the above equations an auxiliary variable ε/k has been introduced. This allows us to decouple and linearise the equations by calculating the auxiliary variable at the previous iteration as proposed in Gaston (1997) and used by Hachem (2009). Time discretisation is carried out using a classical θ scheme similar to the scheme described in §3.5.3, leading to the following fully discrete matrix equation for k :

$$\begin{aligned} M \frac{k^{n+1} - k^n}{\Delta t} + A(\bar{\mathbf{u}}^{n+\theta_{nl}})k^{n+\theta} + K(\nu_T^n, \bar{\mathbf{u}}^n)k^{n+\theta} \\ = P_k(k^n, \bar{\mathbf{u}}^n) + R_k(k^n, \varepsilon^n)k^{n+\theta} + B_k(k^n, \bar{\mathbf{u}}^n), \end{aligned} \quad (4.27)$$

where B_k are boundary conditions on k and $k^{n+\theta}$ indicates that k is evaluated at $\theta\Delta t$ after the time at timestep n . The advection term has a separate subcycling scheme with its own parameter θ_{nl} in which $\bar{\mathbf{u}}^{n+\theta_{nl}}$ indicates evaluation at time $\theta_{nl}\Delta t$. The fully discrete equation for ε is written

$$\begin{aligned} M \frac{\varepsilon^{n+1} - \varepsilon^n}{\Delta t} + A(\bar{\mathbf{u}}^{n+\theta_{nl}})\varepsilon^{n+\theta} + K(\nu_T^n, \bar{\mathbf{u}}^n)\varepsilon^{n+\theta} \\ = P_\varepsilon(k^n, \varepsilon^n, \bar{\mathbf{u}}^n) + R_\varepsilon(k^n, \varepsilon^n)\varepsilon^{n+\theta} + B_\varepsilon(\varepsilon^n, \bar{\mathbf{u}}^n). \end{aligned} \quad (4.28)$$

In (4.27) and (4.28) production and boundary terms are evaluated using values from the previous timestep making these terms fully explicit, whereas dissipation terms use a mixture of values from the previous timestep and previous iteration:

$$R_k = - \left(\frac{\varepsilon^n}{k^n} \right) k^{n+\theta}, \quad (4.29a)$$

$$R_\varepsilon = -C_{\varepsilon 2} \left(\frac{\varepsilon^n}{k^n} \right) \varepsilon^{n+\theta}. \quad (4.29b)$$

Linearising the term in this way may enhance robustness in complex 3D domains (Hachem, 2009).

(4.27) and (4.28) can be rearranged for the unknowns k^{n+1} or ε^{n+1} and solved using standard conjugate gradient methods with ‘BoomerAMG’ algebraic multigrid preconditioning, which offers robust convergence on meshes with arbitrary topology (Henson and Yang, 2002). A specified number of advective subcycles is carried out within a timestep to improve convergence of

the advection term. If a face value limiting scheme such as the Sweby finite-element limiter (cf. §3.5.5) is used, then the allowable timestep is severely limited. This limit can be overcome using a ‘pivot’ method (Leveque, 2002).

Numerically, the standard $k - \varepsilon$ model can be unstable and it is usual to limit or ‘clip’ the values of k and ε to be positive, which ensures that the eddy viscosity is always positive. However, changing nodal values of the solution during an iteration may cause the stability of the equation to deteriorate (Hachem, 2009). In Fluidity the values are limited to $\geq 1 \times 10^{-10}$ prior to calculating the eddy viscosity.

Whilst the discretisation of the k and ε equations is relatively straightforward, the resultant coupled system is much stiffer than the laminar momentum equation (i.e. without a turbulence model). Ferziger et al. (1999) recommended, in the context of finite-volume codes, first performing an iteration of the momentum and pressure correction equations (cf. §3.5.3) using the value of ν_T from the previous iteration followed by an outer iteration for the linearised k and ε equations (since they are highly nonlinear), and finally recalculating ν_T . A similar algorithm, with each field solved separately as in Ilinca and Pelletier (1997), is adopted in this work:

1. given initial conditions $\bar{\mathbf{u}}_0, k_0$ and ε_0 ,
2. compute ν_T from k_0 and ε_0 ,

begin timestep loop

begin nonlinear iteration loop

- (a) solve momentum and continuity for $\bar{\mathbf{u}}$ and \bar{p} ,
- (b) solve k equation,
- (c) solve ε equation,

end nonlinear iteration loop

3. update ν_T ,
4. update production and dissipation terms,

end timestep loop.

4.4.3 $k - \varepsilon$ Model With Mesh Adaptivity

Adaptive meshing has been described in §3.6.1. Its use with the $k - \varepsilon$ model is investigated here, in particular the choice of fields to form an error measure from, and model accuracy and stability compared to fixed-mesh simulations. Hill et al. (2012) performed ocean simulations with the generic two-equation turbulence model of Umlauf and Burchard (2003) and a biology model, finding that the results were sensitive to the choice of fields from which the metric was formed, with different choices performing better in different cases.

The model implementation described above is ready to be used with mesh adaptivity, barring a few details. When a mesh is adapted, all the prognostic fields have to be interpolated onto the new mesh from the old. First the eddy viscosity is updated from the k and ε fields interpolated on the new mesh. Then the production and dissipation terms are updated ready for the next calculation of k and ε . An interpolation error is incurred by using the consistent interpolation method, which is quick but dissipative and non-conservative. Alternatively, Galerkin interpolation is conservative and non-dissipative but a supermesh has to be constructed, which is more costly; see Farrell et al. (2009) for details and Hiester et al. (2011) for a comparison of the methods in a practical example. Consistent interpolation is used in the results presented in §4.6.

Care must be taken when using a model containing gradients with adaptivity. If the resolution changes in a region where k or ε are rapidly varying in space, then there may be a sudden change of gradient terms from pre- to post-adapted meshes because discretised gradient terms are sensitive to the location of mesh vertices. A sudden change in gradient causes a ‘kick’ in the gradient terms, which affects the evolution of the prognostic fields (Hill et al., 2012). This potentially destabilising effect can be reduced by reducing the timestep immediately after an adapt. Hill et al. (2012) found that the choice of interpolation method did not influence the magnitude of the kick.

4.4.4 Scale-Adaptive $k - \varepsilon$ Model (VLES)

The VLES method of Han and Krajinović (2012) has been implemented in Fluidity as an optional modification to make the $k - \varepsilon$ model scale-adaptive. The VLES model is tested in simulations of a vortex diode on a fixed mesh in Chapter 7. Eddy viscosity computed by the $k - \varepsilon$ model is damped by a function given by the following expression:

$$F = \min \left[1, \left(\frac{1 - \exp(-\beta \Delta / \eta)}{1 - \exp(-\beta l_0 / \eta)} \right)^n \right], \quad (4.30a)$$

$$\nu_{VLES} = F \nu_T, \quad (4.30b)$$

where the *ad hoc* model constants are $\beta = 0.002$ from numerical studies by [Speziale \(1997\)](#) and $n = 2$ was calibrated by [Han and Krajnoviç \(2012\)](#). This function interpolates between the largest and smallest lengthscales of the flow, namely the integral scale l_0 and the Kolmogorov scale η (cf. §3.2.2), with the mesh resolution Δ (here defined as the cube root of element volume) falling somewhere between these limits. At the DNS limit, $\Delta = \eta$, $F = 0$ and at the RANS limit, $\Delta = l_0$, $F = 1$. Consequently, the eddy viscosity is scaled to provide an appropriate amount of dissipation for the given resolution.

The scaling given by (4.30) is an improvement on the original form proposed by [Speziale \(1997\)](#), which did not behave well on coarse meshes and did not include l_0 as an upper lengthscale limit. This form is similar to the detached eddy simulation (DES) concept ([Spalart, 2009](#)). Sensitivity to the mesh resolution makes the method akin to an LES model and potentially a good turbulence model to use with mesh adaptivity.

4.5 Verification

To verify that the equations above have been correctly implemented in the CFD code Fluidity, a simple test has been constructed using the method of manufactured solutions (MMS) ([Roache, 2002](#)). In this method a forcing term is constructed and added to the right-hand side of the equation which we wish to force to a given solution. Implementation errors (e.g. code bugs) are discovered by subtracting the calculated solution from the manufactured solution. If the model equations have been correctly implemented then the errors should converge at the rate dictated by the order of accuracy of the numerical discretisation.

Tests were performed on a $\pi \times \pi$ domain with structured triangular meshes named A-D ranging from eight to 64 nodes per side in multiples of two. Figure 4.1 shows the manufactured solutions for $\bar{\mathbf{u}}$, \bar{p} , k , ε and ν_T as well as the production term P_k . The solutions are chosen to be:

$$\bar{u} = 0.6 \sin(y) + \cos(x) + 2.5, \quad (4.31a)$$

$$\bar{v} = y \sin(x), \quad (4.31b)$$

$$\bar{p} = \sin(xy/\pi) + \sin(x) + \cos(y) - 1, \quad (4.31c)$$

$$k = 0.6 \sin(0.7y) + 0.4 \cos(0.8xy/\pi) + 0.9 \cos(0.6x) + 0.9, \quad (4.31d)$$

$$\varepsilon = 1.7 \sin(0.6xy/\pi) - 3.8 \sin(0.7x) + 4.3 \cos(0.8y) + 8.2, \quad (4.31e)$$

$$\nu_T = \frac{[0.6 \sin(0.7y) + 0.4 \cos(0.8xy/\pi) + 0.9 \cos(0.6x) + 0.9]^2}{1.7 \sin(0.6xy/\pi) - 3.8 \sin(0.7x) + 4.3 \cos(0.8y) + 8.2}. \quad (4.31f)$$

4.5 Verification

These forms are similar to those used by Roy et al. (2007) to verify a RANS model, who advised that the magnitudes of the forced fields should be roughly the same if the MMS results are to be valid. Forcing terms for velocity, kinetic energy and dissipation were then obtained using the symbolic mathematics package Sage (Sage, 2011) from (4.31). For example, the momentum forcing term \mathbf{F} in the Reynolds-averaged momentum equation (3.32) is derived by writing (in the x -direction)

$$F_i = \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial}{\partial x_j} \left[(\nu + \nu_T) \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \frac{2}{3} k \delta_{ij} \right] + \frac{\partial \bar{p}}{\partial x_i}, \quad (4.32)$$

and inserting (4.31a), (4.31f) and (4.31c). The forcing terms \mathbf{F}_k and \mathbf{F}_ε for k and ε are derived the same way by rearranging (4.11a) and (4.11b) respectively. $\bar{\mathbf{u}}$, \bar{p} , k and ε are forced to converge to the prescribed manufactured solutions by adding the forcing terms to the right-hand side of their respective equations.

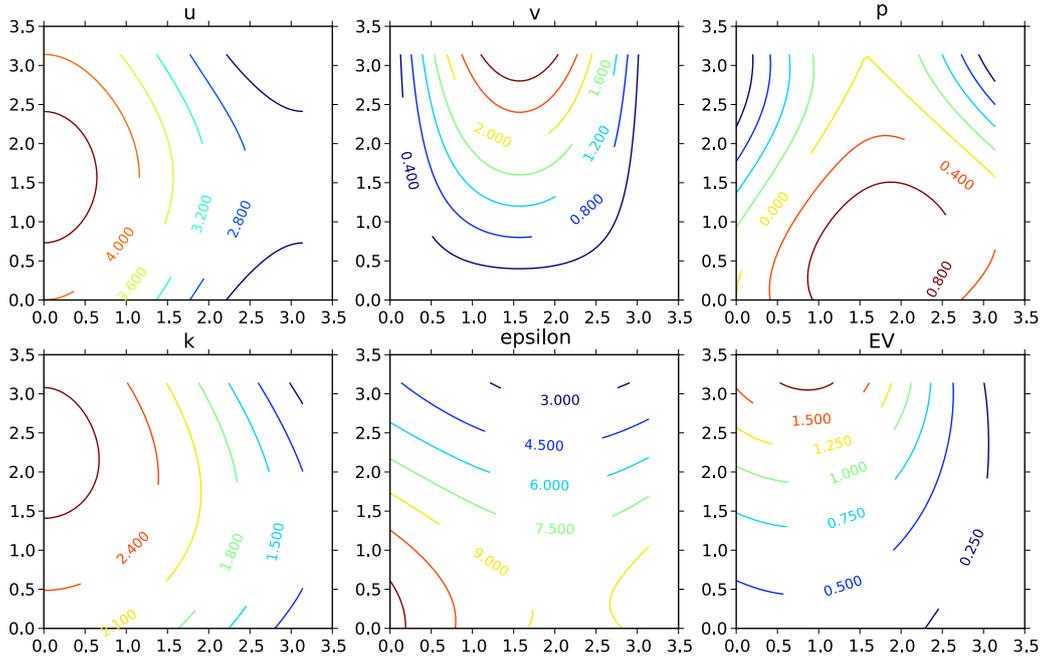


Figure 4.1: Contours of the manufactured solutions in the $k - \varepsilon$ MMS test. ‘EV’ is eddy viscosity.

The momentum equation was discretised using the continuous Galerkin (CG) discretisation with piecewise quadratic shape functions for velocity and piecewise linear shape functions for pressure (the Taylor-Hood element, P_2

4.5 Verification

CG – P_1 CG). k , ε and ν_T equations used P_2 CG. No stabilisation was required on k or ε because the solution is forced. The expected order of convergence for pressure is second order and third order for all other variables. By this we mean that the \mathcal{L}^2 norm of the difference between the exact solution (4.31) and the simulated solution is expected to vary as the square (second order) or cube (third order) of the mesh size. Halving the mesh size from mesh A to mesh B is predicted to reduce the error in the pressure solution by a factor of four. Table 4.1 lists the orders of convergence obtained in the test.

Field	Mesh	Order of Convergence
Velocity	A-B	3.004
	B-C	2.991
	C-D	2.995
		theoretical: 3.000
Pressure	A-B	2.180
	B-C	2.232
	C-D	2.137
		theoretical: 2.000
Kinetic Energy	A-B	3.251
	B-C	3.055
	C-D	2.990
		theoretical: 3.000
Dissipation	A-B	3.135
	B-C	3.016
	C-D	2.994
		theoretical: 3.000
Eddy Viscosity	A-B	2.870
	B-C	2.925
	C-D	2.961
		theoretical: 3.000

Table 4.1: Order of convergence obtained in the $k-\varepsilon$ MMS verification test. ‘A-B’ = order of convergence measured between mesh A and mesh B. ‘Theoretical’ = anticipated order.

As shown in Table 4.1, the expected orders of convergence have been achieved, which implies that the $k-\varepsilon$ model equations have been correctly implemented in Fluidity. An identical test was performed with CV-FE dis-

cretisation and the same orders of convergence were predicted and achieved. Now that the model is verified, it can be validated to find out how well it is able to simulate turbulent flow.

4.6 Validation

Validation of the $k - \varepsilon$ and VLES models has been carried out using a 2D flow past a backward-facing step problem with the geometry shown in Figure 4.2 against experimental data of Kim (1978) and numerical simulations of Ilinca and Pelletier (1997). This version of the backward-facing step is characterised by the expansion ratio (ratio of step height h to inlet height) of 2:1 and Reynolds number of 132,000 based on step height and maximum inlet velocity. Flow enters on the left and a boundary layer develops over the narrow inlet region before separating over the step, impinging on the bottom of the downstream section and re-developing a turbulent boundary layer towards the outlet. The length of the separation vortex behind the step, known as the reattachment length (R_L), is the defining characteristic of the backward-facing step flow.

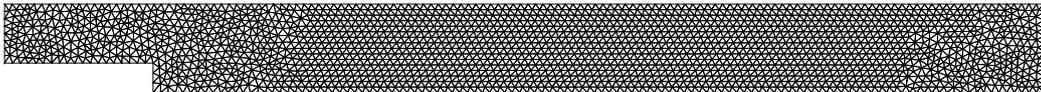


Figure 4.2: 2D backward-facing step geometry showing coarse mesh. Step height $h = 1$, inlet height = $2h$, inlet length = $5h$, total length = $35h$.

The backward-facing step is useful for investigating the behaviour of CFD models in separating and reattaching flows, which are common in engineering applications. Flow behaviour is quite different in two and three dimensions, with significant 3D effects being present in the turbulent mixing layer and reattachment region in the 3D case. In 2D these effects are not present, but complex effects including vortex shedding and boundary layer development make the reattachment length and velocity profiles sufficiently sensitive to the CFD model to be used as a validation exercise.

Many researchers have used the 2D problem to validate a variety of numerical schemes, for example Ercan and Erturk (2008) validated a method in which the streamfunction and vorticity are solved for in place of velocity and pressure. Lew et al. (2001) implemented a version of the $k - \varepsilon$ model in a finite-element code and validated it using the 2D backward-facing step. They found that the reattachment length was in good agreement with experimental data and consistent with the RNG version of the $k - \varepsilon$ model

of Speziale and Thangam (1992). The flow is an example of those given in §4.2.3.1 in which the $k - \varepsilon$ model is expected to perform poorly, but if the results obtained by other researchers with the model can be recreated then the model can be considered validated. Model behaviour on fixed and adaptive meshes is investigated here. The CV-FE discretisation is used with a P_1 -CG P_1 -CG discretisation of velocity and pressure; the velocity mesh is used as the parent mesh for the discrete k and ε equations.

Numerical results from Ilinca and Pelletier (1997) using a finite-element CFD code with the $k - \varepsilon$ model and adaptive meshing are selected for comparison of mean velocity profiles. A least-squares error estimate in the strain rate drove the mesh optimisation algorithm. In the algorithm a series of steady-state RANS simulations were run to iterate towards an optimised mesh by reducing the error estimate each time. It is not clear whether their method could be applied to a transient simulation. They used the Crouzeix-Raviart element ($P_2^{NC} - P_1$ DG), which consists of quadratic non-conforming (NC) basis functions for velocity (non-conforming in the sense that velocity is discontinuous across element boundaries), piecewise linear functions for pressure and a quadratic interpolant for k and ε . The quadratic non-conforming basis function acts like a bubble function, a stabilisation technique that is analogous to an LES model (Candy, 2008). In addition to the stable element, Ilinca and Pelletier (1997) used streamline upwinding (SU) which adds numerical dissipation to stabilise the discretisation (cf. §3.5.6). It was not stated whether this affects the dissipative contribution of the $k - \varepsilon$ model.

4.6.1 Simulation Setup

The setup was identical for $k - \varepsilon$ and VLES simulations. Three fixed unstructured meshes were generated in `gmsk` with edge lengths of approximately 0.2, 0.1 and 0.05, labelled ‘coarse’, ‘med’ and ‘fine’ with 2187, 11,144 and 44,235 nodes respectively. In addition, four adaptive meshes were generated, each adapting to the curvature of different solution fields. Vector fields are treated as two scalar fields and each component is equally weighted in the desired error ε_u . Interpolation error bounds could be applied as absolute or relative values. Absolute values were chosen to give a relative error bound of approximately 0.5%. An upper limit of 20,000 nodes was applied as a constraint on the optimisation, in order to limit the computational workload. Homogeneous anisotropic minimum and maximum edge lengths were specified to reduce resolution in the main flow direction and increase it in the perpendicular direction, preventing excessive resolution. The details are given in Table 4.2 including the actual number of nodes in the final meshes.

All simulations used an adaptive timestep with the stability condition

4.6 Validation

mesh	A1	A3	A4	A5
adapt to error in...	$\bar{\mathbf{u}}$	k, ν_T	k, ν_T	$\bar{\mathbf{u}}, \bar{p}, \varepsilon, k, \nu_T$
desired error ε_u (absolute or relative (%))	0.01, 0.01	0.5%, 0.5%	0.001, 0.0005	0.01, 0.001, 0.001, 0.001, 0.0005
period (timesteps)	50	50	50	50
final no. of nodes	8600	19300	21700	21100
min. size (x, y)	(0.03, 0.01)	(0.03, 0.01)	(0.03, 0.01)	(0.03, 0.01)
max. size (x, y)	(0.5, 0.2)	(0.5, 0.2)	(0.5, 0.2)	(0.5, 0.2)
gradation parameter	1.5	1.5	1.5	1.5

Table 4.2: Adaptivity parameters in 2D backward-facing step simulations. Error bounds are relative if given as a percentage and absolute if not. Gradation parameter controls the ratio of sizes of adjacent elements.

$CFL \leq 1$, i.e. the timestep is determined by the minimum mesh size and velocity across it. In adaptive simulations a very small fixed timestep was set at the first timestep after an adaptation to damp oscillations as explained in §4.4.3. Simulations were run until either 300 seconds or steady state was reached, measured by the convergence of the \mathcal{L}^2 norms of both velocity magnitude and scalar eddy viscosity with a tolerance of 1×10^{-7} (Ilinca and Pelletier (1997) specified 1×10^{-8}). The inlet velocity profile was flat in the middle with ramps up from zero to 2.3 between $y=0$ and $y=0.1$ and between $y=2$ and $y=1.9$ (measured from the bottom of the inlet). An outlet condition on velocity was necessary to stabilise the VLES simulations: an absorption or ‘sponge’ region was specified close to the end to prevent negative velocity on the outflow plane. No such condition was required for the $k - \varepsilon$ simulations.

4.6.2 Results

4.6.2.1 Reattachment Length

All $k - \varepsilon$ simulations converged to steady-state solutions, therefore reattachment length (R_L) is calculated at the end of the simulation. VLES simulations converged in a statistical sense: time-averaged velocity, where averaging began after an initial 50 seconds of ‘spin-up’ time, attained a steady state. R_L was measured by searching along a line a distance of 0.01 above the bottom boundary for the change of streamwise velocity direction from backwards to forwards. Table 4.3 lists the R_L predictions for all simulations against those from Kim et al. (1987) and Ilinca and Pelletier (1997).

4.6 Validation

Mesh	Model	Velocity BCs	$k - \varepsilon$ BCs	R_L/h
coarse	$k - \varepsilon$	no-slip	$k = 0, \varepsilon = 0$	4.96
			high-Re	4.74
		weak log law	$k = 0, \varepsilon = 0$	4.47
			high-Re	5.67
med	$k - \varepsilon$	no-slip	$k = 0, \varepsilon = 0$	5.52
			high-Re	5.29
		weak log law	$k = 0, \varepsilon = 0$	5.62
			high-Re	6.38
fine	$k - \varepsilon$	no-slip	$k = 0, \varepsilon = 0$	5.91
			high-Re	4.82
		weak log law	$k = 0, \varepsilon = 0$	5.94
			high-Re	5.66
A1	$k - \varepsilon$	weak log law	high-Re	5.25
A3			5.36	
A4			5.33	
A5			5.27	
coarse	VLES	weak log law	high-Re	2.29*
med				2.21*
fine				2.30*
Experiment (Kim, 1987)				7.0 ± 1
Adaptive RANS (Ilinca and Pelletier, 1997)				6.21

Table 4.3: Reattachment length predictions in the 2D backward-facing step.
*Measured from time-averaged velocity contours.

In general, R_L predictions improved with increasing mesh refinement. The spread of predictions on each mesh indicates the sensitivity of R_L to simulation setup. All were under the experimental value of $R_L = 7$ and most were less than the value of 6.21 obtained by [Ilinca and Pelletier \(1997\)](#). These findings are consistent with the known under-prediction of R_L by the $k - \varepsilon$ model: the seven different $k - \varepsilon$ model results quoted in [Lacasse et al. \(2004\)](#) were between $R_L = 5.2$ and $R_L = 6.2$. On the coarse and medium fixed meshes the standard combination of weak log law and high-Re wall functions for k and ε (4.23) led to an improvement in R_L compared to all other permutations. Conversely, on the fine fixed mesh the best predictions were obtained with homogeneous Dirichlet boundary conditions for k and ε and either the weak log law or no-slip conditions on velocity.

Using mesh adaptivity with the $k - \varepsilon$ model, weak log law and high-Re boundary conditions results in reasonable R_L predictions while at least halving the number of nodes (80% reduction in A1 case) compared to the same model on the fine fixed mesh. Results on all four adaptive meshes are very similar to each other. Minimum element sizes were reached in all adaptive meshes (see Figure 4.6, page 121), so the reason for the shorter R_L cannot be that there was insufficient resolution. Rather, it is the fault of the log-law wall function: near-wall resolution is very fine in these meshes and consequently the near-wall node is within the viscous sublayer instead of the log law region. This may have spoilt the fine fixed mesh results too. Two remedies are apparent. Firstly, a correction to the wall function is required such that it scales across a wider range of y^+ values, for example by incorporating the Reichardt law (Hachem, 2009) or scale-adaptive wall functions such as those described in §4.3.2. Secondly or alternatively, it would be advisable to calculate y^+ as a diagnostic field, potentially to be used as a constraint for adaptivity to ensure valid use of wall functions.

The VLES model under-predicts R_L by a considerable margin, but R_L is fairly constant across the range of mesh resolution demonstrating that the VLES model does indeed compensate for mesh resolution. No comparable VLES results are available from the literature to say whether this is expected behaviour. It is possible that the model does not correctly account for the full range of scales in the flow, and that the dissipative effect of some scales is missing, resulting in insufficient dissipation of the large scales and shorter R_L . Further tests on adaptive meshes might show how the compensating effect manifests within a single simulation.

4.6.2.2 Velocity Profiles

Figure 4.3 compares velocity contours from the $k - \varepsilon$ model on the coarse, medium, fine and A5 meshes versus those from Ilinca and Pelletier (1997) and Kim (1978) at several points downstream of the step. As the mesh is refined from coarse to fine, the agreement with the reference data in the shear layer ($y \approx 1$) and in the top boundary layer ($y > 2.5$) improves. Resolution of the large velocity gradients very close to the bottom boundary at $x/h = 5.33$ and $x/h = 16$ is poor in both the Fluidity and Ilinca results, which suggests that the standard wall functions are inadequate in this type of separating and reattaching flow, as has been found by other researchers (Wilcox, 1998). The adaptive profiles are comparable to the coarse mesh profiles, being too rounded in the shear and top boundary layers. Very close to the top wall the A5 mesh has resolved the highest gradients of all the profiles shown. These phenomena are probably linked to the invalid use of the log-law wall function

4.6 Validation

which degraded the R_L predictions.

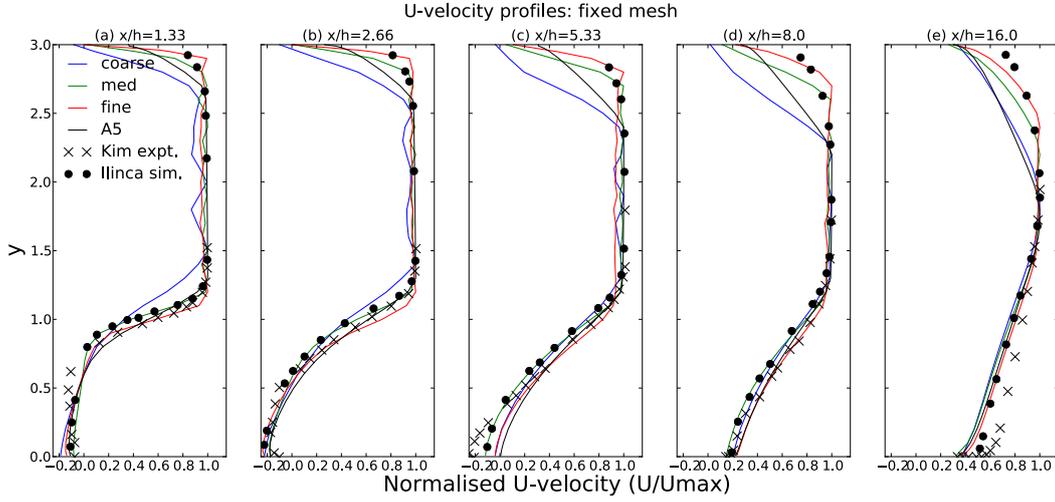


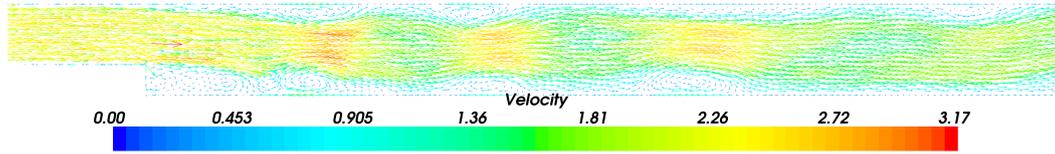
Figure 4.3: Normalised velocity profiles at five positions downstream of the step showing coarse/med/fine/A5 mesh results, Ilinca numerical results and experimental data.

4.6.2.3 VLES

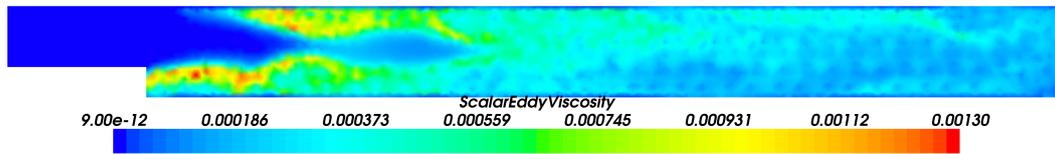
Figure 4.4 shows instantaneous velocity and eddy viscosity predicted by the VLES model on the coarse, medium and fine meshes. On the coarsest mesh, the velocity (Figure 4.4a) exhibits a wavy pattern indicating some transience. Eddy viscosity is patchy with a maximum in the shear layer near the step, decaying downstream. On the medium and fine meshes, smaller coherent structures are resolved by the model, particularly near the step, and the flow exhibits periodic vortex shedding from the step. In between vortices the flow is fairly uniform. The maximum value of eddy viscosity occurs near the step, its value reducing as the mesh resolution increases owing to the action of the VLES filter function. The period of vortex shedding is approximately the same on medium and fine meshes, suggesting that the solution may become mesh-independent at finer resolutions. Hachem (2009) observed similar patterns in this problem using a VMS approach (cf. §3.5.6.6).

Figure 4.5 shows the VLES filter function (4.30) on the coarse, medium and fine meshes. Red areas ($F = 1$) correspond to RANS solutions and the uniform low velocity and eddy viscosity areas in Figure 4.4. Low values of F correspond to larger values of eddy viscosity and transient behaviour such as vortex shedding. This is perhaps counter-intuitive, since the filter function multiplies the eddy viscosity in the model. The coupled nature of the model

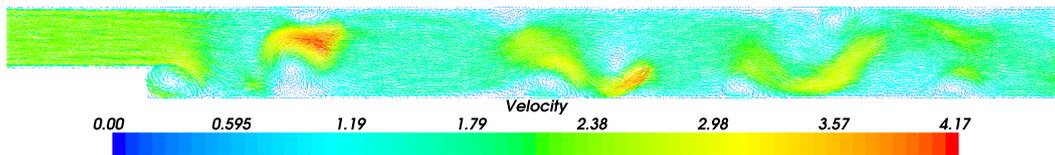
4.6 Validation



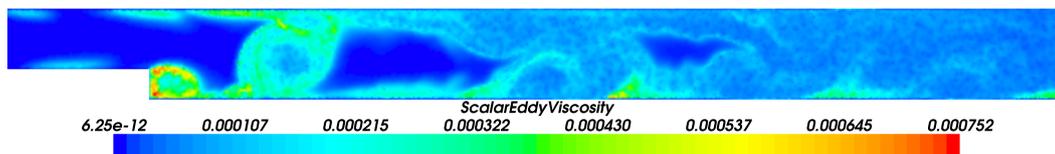
(a) Coarse mesh, \bar{u}



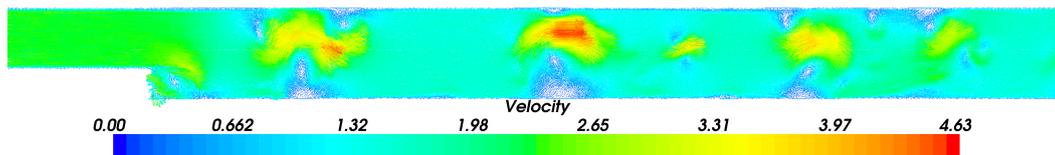
(b) Coarse mesh, ν_T



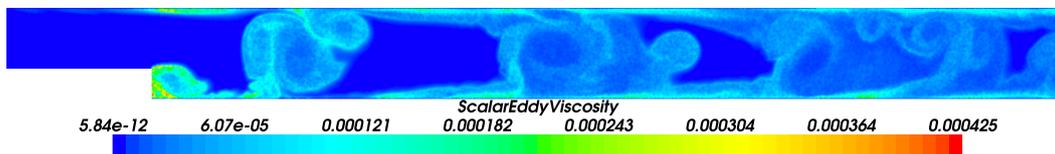
(c) Medium mesh, \bar{u}



(d) Medium mesh, ν_T



(e) Fine mesh, \bar{u}



(f) Fine mesh, ν_T

Figure 4.4: Snapshot of velocity and eddy viscosity predicted by VLES model in 2D backward-facing step simulations.

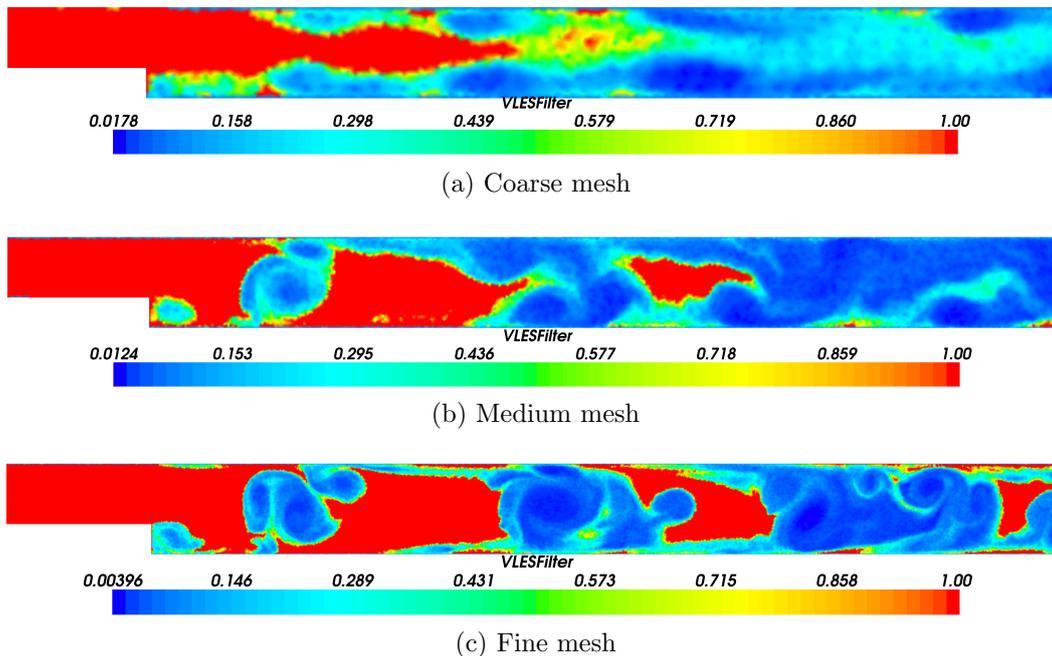
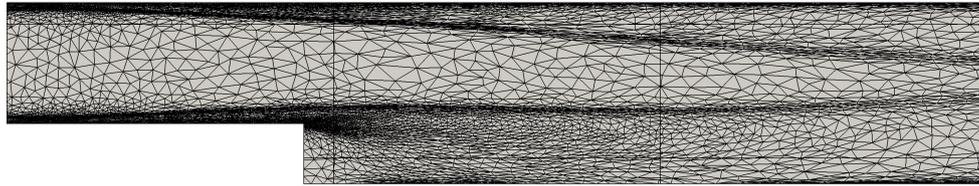


Figure 4.5: Snapshot of VLES model filter in 2D backward-facing step simulations.

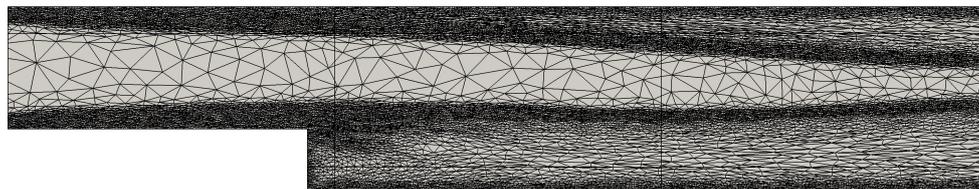
means that the velocity, kinetic energy, dissipation and lengthscale are all modified by the filter via the eddy viscosity, which in turn changes the eddy viscosity and leads to the observed behaviour. The demarcation between RANS and LES is sharp. These results suggest that the VLES model is able to resolve large coherent structures in 2D. Investigation of the model in 3D is essential to determine whether it is a valid turbulence parameterisation, since it is not possible to generate realistic turbulent flows in 2D.

4.6.2.4 $k - \varepsilon$ Model with Adaptivity

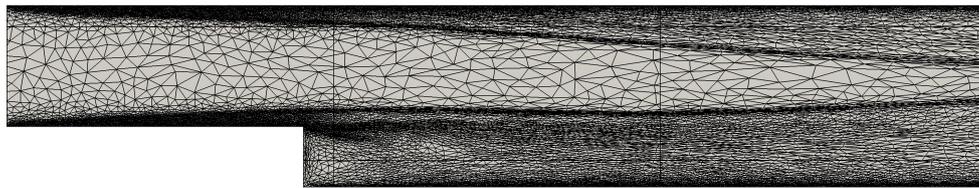
Figure 4.6 shows a portion of the final meshes generated in the A1, A3, A4 and A5 simulations compared to the adaptive mesh of [Ilinca and Pelletier \(1997\)](#). Their mesh adapted to every prognostic field, although using a different adaptivity technique to Fluidity, and resolution is clustered in the separation and reattachment zones. There is little difference between the Fluidity meshes or indeed the reattachment lengths predicted. Mesh A1 is the coarsest overall but resolves the step lip and inlet boundary layer very finely. Mesh A3, adapting to the curvature of k and ν_T with relative error bounds, exhibits smeared resolution in the boundary layers and shear layers. Mesh A4 adapts to the curvature of k and ν_T but with absolute error bounds; it has more localised and finer resolution in the inlet boundary layers and



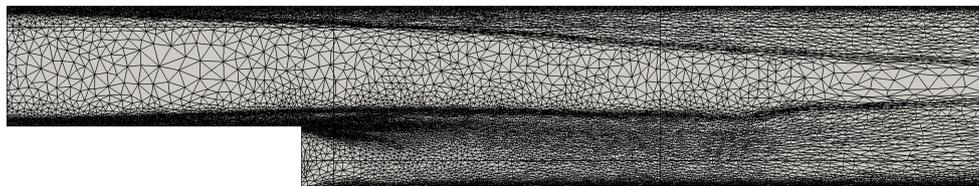
(a) A1 adaptive mesh



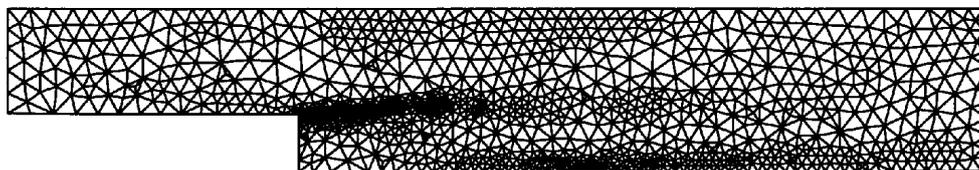
(b) A3 adaptive mesh



(c) A4 adaptive mesh



(d) A5 adaptive mesh



(e) Adaptive mesh of Ilinca and Pelletier (1997) with 7000 nodes

Figure 4.6: Portion of final adaptive meshes in 2D backward-facing step simulations showing the influence of the choice of target field.

step lip, and has also resolved the walls downstream of the step. Mesh A5, adapting to the curvature of every prognostic field, is similar to A1 and A4, which is expected considering that it combines the adaptivity parameters of those two meshes.

Figure 4.7 shows dissipation ε near the separation point on meshes A1, A3, A4 and A5. The maximum value of ε occurs at the separation point and varies considerably between the meshes; the maximum value is roughly correlated with the resolution at the step lip. A likely reason is that the production term in the equation for ε involves velocity gradients and is therefore sensitive to spatial resolution. The highest value is on mesh A5, which adapts to the error in ε and is therefore expected to yield the most accurate value.

Figure 4.8 compares contours of the solution fields obtained by Fluidity with mesh A5 and the adaptive finite-element $k - \varepsilon$ results of [Ilinca and Pelletier \(1997\)](#). The contour plots are very similar suggesting that the $k - \varepsilon$ model as implemented in Fluidity is behaving correctly in this test case. Slight differences are apparent at the top boundary: Fluidity has generated higher viscosity, dissipation and kinetic energy there. The wall functions may be responsible for this behaviour, suggesting that scale-adaptive wall functions may be necessary to ensure that the log law is only applied when the near-wall resolution is greater than $y^+ = 30$.

Apart from differences in the maximum values of ε at the step (Figure 4.7), the converged solutions $(\bar{\mathbf{u}}, \bar{p}, \varepsilon, k, \nu_T)$ on all adaptive meshes were nearly identical, suggesting that the use of adaptivity enabled the model equations to home in on a unique solution. The fact that it is not the most accurate solution (compared to the fixed fine mesh) is less important. Adapting to the curvature of any one, or combination, of the prognostic fields in the coupled system of equations led to the same solution.

The A4 mesh contained 21,700 nodes and the minimum edge lengths were reached throughout the domain. Accuracy of the solution was therefore limited by either the error bound, minimum edge lengths or maximum number of nodes. To test whether the adaptive solutions had truly converged, two further simulations were run: one using a setup identical to mesh A4 but with the maximum number of nodes also increased to 40,000, and the second the same as the first but with the error bounds reduced by half to 0.0005 on k and 0.00025 on ν_T . The first established whether total resolution was limiting accuracy and the second whether the error bound was limiting accuracy. Reattachment lengths of $R_L/h = 5.30$ and 5.32 respectively were obtained, while the meshes contained 27,500 and 41,500 nodes respectively. A further test with only the minimum edge lengths reduced by half returned $R_L/h = 5.28$ on 20,500 nodes. Therefore, the total number of nodes had limited the mesh but not the accuracy in terms of calculating reattachment length.

4.6 Validation

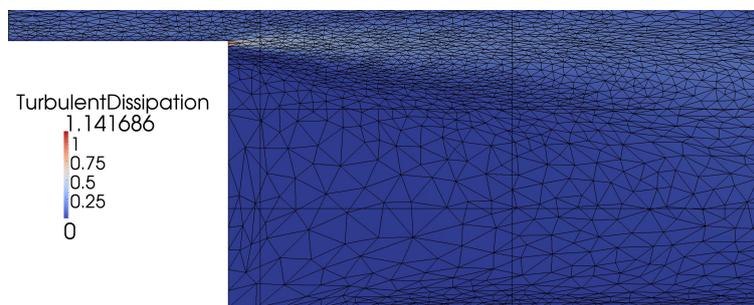
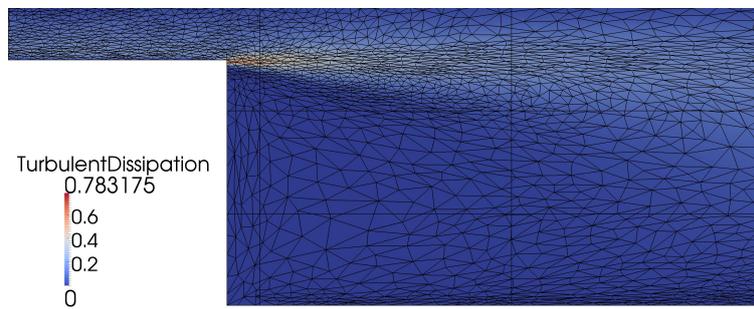
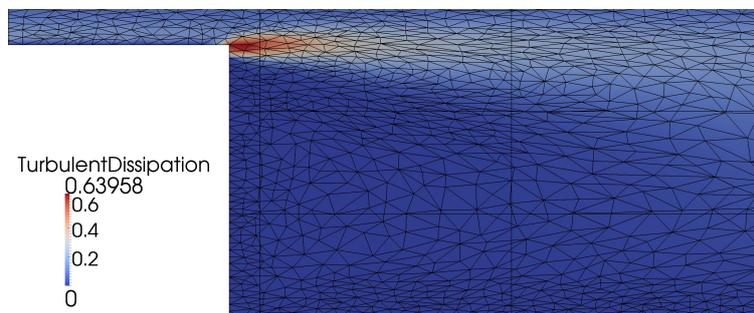
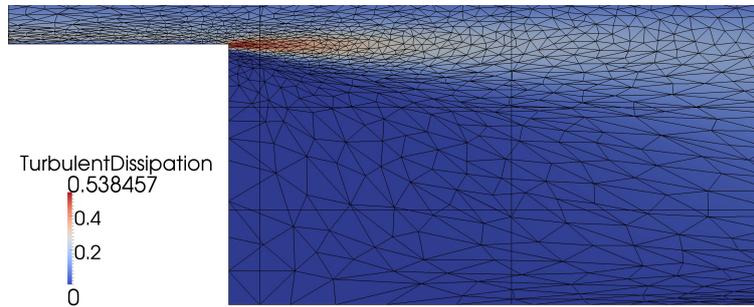


Figure 4.7: Turbulent kinetic energy dissipation ε in step region of 2D backward-facing step.

4.6 Validation

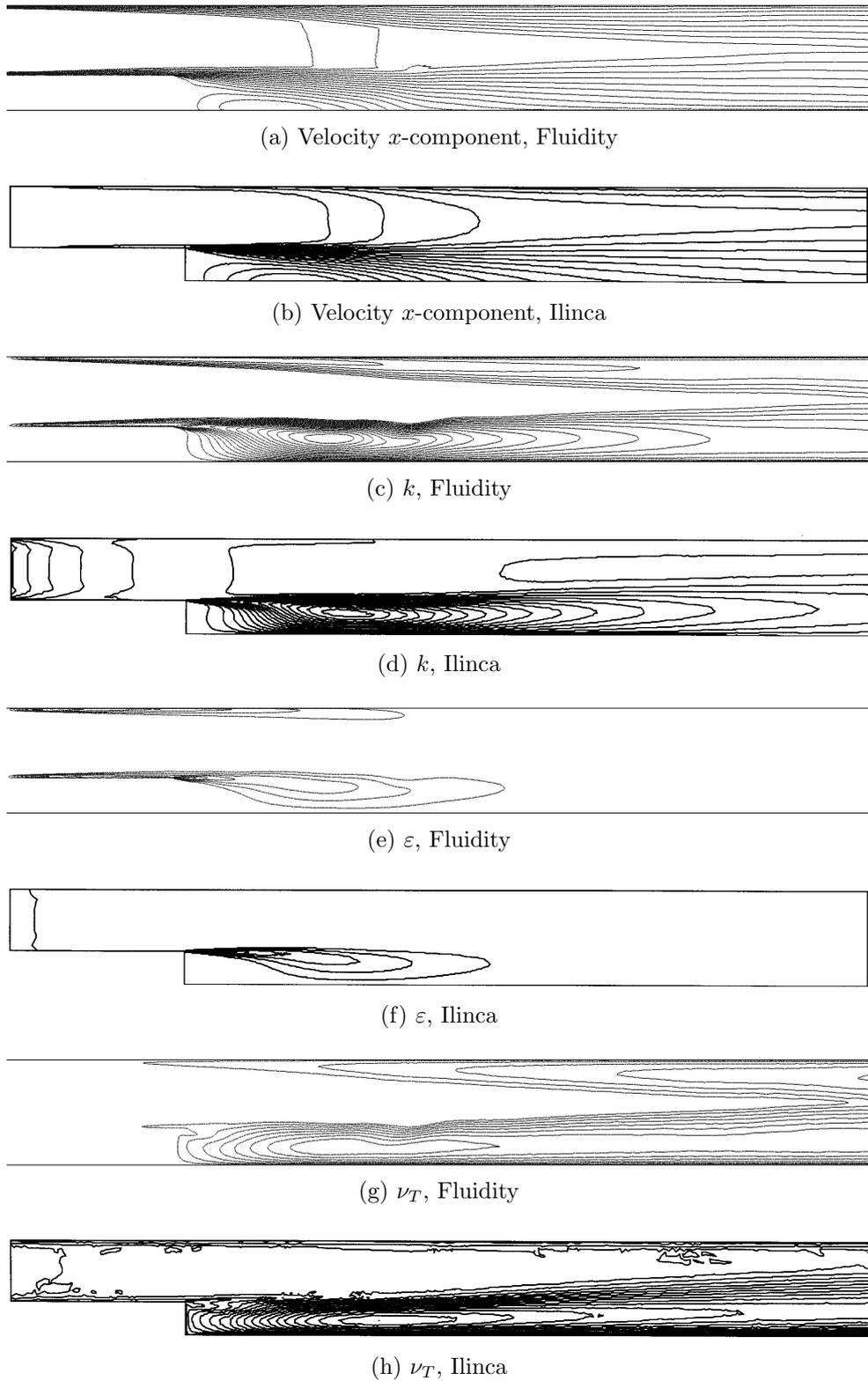


Figure 4.8: Contours of solution fields on A5 mesh comparing Fluidity to model of Ilinca and Pelletier (1997).

A similar experiment was conducted with mesh A1. In this case the error bound was limiting the mesh generated but again, reattachment lengths were comparable in all permutations. These results suggest that the solutions on meshes A1-A5 had indeed converged to a mesh-independent result.

4.7 Discussion

The standard $k - \varepsilon$ model has been implemented in Fluidity using the CV-FE discretisation. Convergence of numerical errors in the model has been quantified by the method of manufactured solutions with both CG and CV-FE discretisations of the model equations. A variational law-of-the-wall boundary condition for velocity enabled accurate predictions to be made with coarse near-wall resolution in high-Re flows. Any turbulence model can be used with this condition, but in this chapter it was used with the $k - \varepsilon$ model to simulate 2D flow over a backward-facing step with comparable results to other $k - \varepsilon$ simulations of this problem. Under-prediction of reattachment length by the $k - \varepsilon$ model, widely reported in the literature, was confirmed by the results presented above, although predictions on coarse meshes were improved by the action of the law-of-the-wall boundary condition.

A scale-adaptive modified $k - \varepsilon$ model, the VLES model of [Han and Krajinović \(2012\)](#), was implemented in Fluidity. Simulations of the 2D backward-facing step showed that the model behaved rather like a detached-eddy simulation (DES). The model generated transient solutions which included coherent large-scale turbulent structures and showed a distinct transition from a steady-state to a transient regime. More complexity and smaller-scale details were included with increasing mesh refinement. Reattachment length predictions were very poor but consistent across the range of meshes tested, showing that the model compensates for mesh resolution.

Since turbulence is a 3D phenomenon, 3D simulations should be performed to validate the VLES model. The presence of large areas in which no transient structures were visible suggests that the underlying $k - \varepsilon$ model is too dissipative. In future it would be worth testing the VLES modification on other more accurate RANS models, such as the $k - \omega$ SST model tested in Chapter 7, to find out whether accuracy can be improved by changing the behaviour in the RANS regime. Research reported in [Fröhlich and von Terzi \(2008\)](#) suggested that using the SST model in the DES method was beneficial, but it has not yet been tested with the VLES modification. Testing is also required to investigate the VLES model's behaviour on non-uniform and adaptive meshes.

Using mesh adaptivity with the $k - \varepsilon$ model resulted in similar reattach-

ment length predictions with up 50% to 80% fewer nodes compared to the fine fixed mesh. Resolution in the final converged mesh was concentrated in the boundary layers, near the step lip and in the shear layer. Velocity profiles were not reproduced as accurately as on the fixed fine mesh, possibly due to inappropriate use of the log law on very fine near-wall meshes. Finely resolving the region around the step lip was important to capture peak values of modelled quantities.

Apart from differences in peak values in a very few elements at the step lip, the solutions obtained on different adaptive meshes were almost identical. The choice of field or fields from whose curvature to form an error estimate did not make a significant difference to the mesh generated or the results obtained; reattachment lengths varied between 5.25 and 5.36. Choosing a combination of k and ν_T led to smeared resolution in these regions, but prediction of reattachment length was marginally improved. Relative versus absolute error bounds on k and ν_T made some difference to the mesh generated (A3 vs. A4) but not to the solutions or reattachment lengths. Further investigation of mesh adaptivity is required to establish whether the poor results were due to the wall functions and whether the $k - \varepsilon$ model with improved wall functions is better on adaptive than fixed uniform meshes.

Combining turbulence modelling with mesh adaptivity is a fertile area for exploration. The potential to reduce the number of mesh nodes and automatically arrive at a mesh-independent solution is an attractive option for industrial CFD. So far, a standard RANS model has been investigated, which only resolves large scales of motion. More work is needed to refine the technique of adaptive RANS simulation, including which fields to form the error measure from, how small the error should be and how fine the finest elements need to be in order to arrive at a mesh-independent solution. The method should be used with a more sophisticated and accurate RANS model and improved wall functions in order to show its true potential. In the next chapter, the method is applied to large-eddy simulation, where very small, rapidly evolving turbulence provides a more difficult challenge for the mesh to follow.

5 Development of An Adaptive Dynamic Tensorial Large Eddy Simulation Model

In this chapter, the challenge of performing large eddy simulation on unstructured inhomogeneous meshes is undertaken. A novel tensorial dynamic LES model is developed to address some of the shortcomings of standard LES modelling techniques. This robust model is designed specifically for use with unstructured mesh adaptivity and particular attention is paid to commutation errors arising from variable mesh resolution. A novel tensorial filter operator is developed in an attempt to reduce these errors and its properties are verified in simple tests. A discussion of strategies for combining LES with mesh adaptivity is presented.

This chapter expands upon the methods presented in [Bull et al. \(2012\)](#).

Contents

5.1	Introduction	129
5.1.1	Uses of LES	129
5.1.2	The Resolution Problem	130
5.1.3	Implicit and Explicit LES	131
5.1.4	Ideal LES Model Qualities	132
5.1.5	Smagorinsky LES Model	133
5.1.6	Avenues of Improvement	133
5.2	A New Tensorial Dynamic LES Model	142
5.2.1	Tensor Eddy Viscosity	142
5.2.2	Dynamic Method	143
5.2.3	Tensorial Dynamic Method	150
5.2.4	Improved Filter Width Definitions	151
5.3	A Tensorial Differential Filter For LES	152
5.3.1	Introduction to Filtering	152
5.3.2	The Inverse Helmholtz Filter	155
5.3.3	Filter Comparisons	156
5.3.4	Commutation Errors	158
5.3.5	Implementation of the Tensorial Inverse Helmholtz Filter	161
5.3.6	Discrete Filters	164
5.3.7	Verification of Filters	165
5.4	Adaptive LES Modelling	180
5.4.1	Errors in LES Solutions	180
5.4.2	Convergence of LES Solutions	181
5.4.3	Complete LES Modelling Strategies	183
5.4.4	Interpolation Error-Based Adaptivity with LES	184
5.4.5	Goal-Based Adaptivity with LES	186
5.4.6	Sub-Filter Kinetic Energy Estimation	188
5.4.7	Discussion	190

5.1 Introduction

This chapter describes sub-filter-scale (SFS) models for the SFS stress tensor τ_{ij} based on the Leonard or triple decomposition of the filtered Navier-Stokes equations (3.27) described in §3.3.3. We begin by looking at the uses of LES, the difference between implicit and explicit filtering (respectively LES) and the qualities of an ideal LES model, followed by the simplest and most popular LES model, the Smagorinsky model. After that several improvements are explained before moving on to the development, implementation and testing of a new tensorial dynamic LES model. Verification and validation (V & V) have been performed using the finite element code Fluidity; more details on the code and methods of V & V can be found in Farrell et al. (2010). In Chapter 6 results for the 3D backward facing step are presented and comparisons are made to the DNS data of Le et al. (1997) and to the commercial CFD code CFX.

Some terminology is clarified before continuing. An ‘LES model’ refers to a model for the SFS stresses. The ‘dynamic method’ is not an LES model but a generic method of calculating a flow-dependent quantity, into which an LES model is inserted. Methods refer to general calculational procedures rather than specific model forms. The ‘dynamic LES model’ is an LES model using the dynamic method.

5.1.1 Uses of LES

Many types of flow have been investigated with LES, from the simplest canonical flows including homogeneous isotropic turbulence (Fureby and Tabor, 1997) and channel flow (Moin and Kim, 1982), to flows with one direction of homogeneity such as a square cylinder (Sohankar et al., 2000) and backward-facing step (Toschi et al., 2006), to complex real-world flow problems from nuclear engineering such as flow over a fuel rod bundle (Takata et al., 2007) and thermal mixing in a T-junction (Kuczaj et al., 2010). Methods for simulating this last category of flow, which are characterised by turbulent structures that vary considerably from region to region, are the topic of this chapter.

LES generates much richer results, such as time series and energy spectra, than can be obtained with RANS models. This kind of information is required to solve key industrial problems, e.g. material fatigue due to temperature fluctuations (Takata et al., 2007), fluid-structure coupling (Sampaio et al., 2004) and aeroacoustics (Jarrin et al., 2009; Addad et al., 2003). However, automatically selecting LES for a problem in the hopes that it will be “more accurate than RANS” may be ill-advised, but provided that appropriate meshes and numerical solution algorithms are used, it can be superior

(Chabard and Laurence, 2009).

What is seldom reported in the literature is that significant trial and error often goes into finding the optimum configuration (Chabard and Laurence, 2009). Furthermore, determining the accuracy of an LES solution *a posteriori* is not easy. With LES usage broadening from validation test-cases where reference data is available, to actually making predictions where (high quality) reference data is not always available, there is greater focus on quality criteria and best practice guidelines so that we can have confidence in the solutions obtained, e.g. Meyers (2008), Knopp et al. (2010), Davidson (2011).

5.1.2 The Resolution Problem

According to Kolmogorov’s theory of the energy cascade (cf. §3.2.2), most of the kinetic energy and anisotropy of the flow are contained in the inertial range, and the smallest scales are isotropic and flow-independent. Many LES models, including the Smagorinsky model, are based on the assumption of local isotropy in the SFS, requiring that the filter width $\overline{\Delta}$ (which determines the scale separation into resolved and sub-filter scales cf. §3.3.3) is sufficiently far down in the inertial range (Pope, 2000). Filter width is bounded by the mesh size, $\overline{\Delta} \geq \Delta$, therefore the assumption of isotropy makes demands on mesh resolution.

As a rule of thumb, it can be shown by examination of the Kolmogorov energy spectrum that if $\overline{\Delta}$ is such that 80% of the turbulent kinetic energy is in the resolved motions $\overline{\mathbf{u}}$, then the LES solution can be called ‘well-resolved’ (Pope, 2000). In that case, simulation results are fairly insensitive to the LES model employed (Nieuwstadt and de Valk, 1987). Unfortunately, the mesh resolution requirement can be impractical with today’s computers in high-Re problems because the range of energy-containing (integral) scales increases with Re. Moreover, estimates of the amount of energy in the sub-filter scales depend on the LES model (Klein, 2005) (see §5.4.6 for a detailed discussion).

To compound the difficulties, in high-Reynolds-number boundary layer flows, eddy size tends to zero at the wall and it becomes impossible to resolve the entire energy-containing range (Mason, 1994). Then a large proportion of the turbulent stress (hence turbulent kinetic energy) near the walls is carried by the LES model, causing a strong dependence of near-wall dynamics on the choice of LES model (Porte-Agel, 2004). The physics of wall-bounded anisotropic turbulence is also more complicated than ‘free’ isotropic turbulence and poorly understood; many empirical near-wall models have been developed to parameterise the flow with varying levels of success.

5.1.3 Implicit and Explicit LES

Two alternative strategies can be considered for choosing the filter width (Geurts and Fröhlich, 2001). On the one hand, the filter width is tied to the mesh size by a fixed ratio:

$$\overline{\Delta} = \alpha \Delta, \quad \alpha \geq 1, \quad (5.1)$$

where the actual action of filtering is implicit in the discretisation method on a computational mesh, hence this method is known as *implicit filtering* (hence *implicit LES*)¹. Resolved flow information in LES is defined as those scales which are equal to or larger than the filter size, not the mesh. Therefore, implicit LES maximises the amount of resolved flow information on a given mesh because the filter size can be as small as the mesh size. Refining the mesh reduces the filter width, improving modelling accuracy.

In implicit LES, the role of α is simply to set the filter size in the LES model. Pope (2000) and Ferziger et al. (1999) recommended $\alpha = 2$ for the following reason. Consider a 1D uniform mesh using either finite elements with piecewise linear continuous shape functions or the control volume (CV) discretisation. By the Nyquist sampling theorem, the shortest resolvable wavelengths on that mesh require at least three degrees of freedom (i.e. two elements or CVs) to be represented. Any wavelengths shorter than that cannot be represented and are thus filtered out and relegated to the SFS. Hence the filter width has to be twice the mesh size: if less, then the largest SFS are missed and if more, then they are double-counted.

However, in 3D and on inhomogeneous unstructured meshes the filter width definition is not so clear, negating such a simple treatment. It is therefore a valid choice to set $\alpha = 1$ in order to maximise usable information. Results presented in Chapter 6 give weight to the idea that using $\alpha = 1$ is acceptable. Nevertheless, doing so may incur significant discretisation or ‘aliasing’ errors in the smallest resolved scales, but this may be offset by gains in modelling accuracy. Simulations by Klein (2005) and Meyers et al. (2003) show that discretisation and modelling errors interact in complex ways.

If the filter width is held constant and the mesh is refined it can be shown that the LES solution converges to some mesh-independent LES solution as the discretisation error (determined by mesh size) becomes relatively small compared to the modelling error (determined by the form of the SFS stress model) (Geurts and Fröhlich, 2001). In this case the filter width is no longer tied to the mesh size and *explicit filtering* must be performed using some

¹N.B. This is distinct from the Implicit LES (ILES) methods in §3.5.6 which use the truncation error in the discretised equations as an LES model, e.g. Hickel et al. (2007).

filter operator acting on the discrete solution, hence the LES method is called *explicit LES*².

Whilst it is useful from a numerical point of view to separate the modelling and discretisation errors, the amount of information in the filtered solution is much smaller than that which is representable on the mesh as the finest scales resolvable on the mesh are excluded. Explicit LES has been used successfully in a number of cases, e.g. Gullbrand (2003a) found that explicit filtering improved the prediction of turbulence intensity versus implicit LES. Implicit LES is more commonly used because it is deemed a more efficient use of resources, although the errors become intertwined if α is small (Geurts and Fröhlich, 2001). Some additional advantages and disadvantages of implicit versus explicit filtering are raised by the dynamic method, as discussed in §5.2.2.5.

5.1.4 Ideal LES Model Qualities

In practical or engineering terms, an ideal LES model is one that does not need to be tuned to a particular flow type or geometry, optimises use of computational resources, captures the inherent inhomogeneity and anisotropy of turbulence and has a well-defined convergent error. In order to be used in complex geometry, the model should be applicable to unstructured meshes. This chapter describes the steps taken towards such a model and the challenges encountered along the way.

In addition to these practical qualities, there are some basic mathematical qualities that any LES model is required to have³. These ensure that the model equations have the same properties as the exact Navier-Stokes equations and therefore that the simulated turbulence is a physically valid approximation to the real flow. For example, the fluid stress tensor τ_{ij} is a positive semi-definite matrix. The *realisability* condition requires that the modelled SFS stress tensor is also positive semi-definite. It can be proved that it is so, if and only if the LES filter kernel $G(\mathbf{x})$ is non-negative (cf. §5.1) (Geurts and Holm, 2004). However, if the LES filter is implicit, then $G(\mathbf{x})$ is not explicitly known so it is not clear whether the realisability condition is satisfied (Berselli et al., 2006).

The consistency condition states that the solution to the filtered Navier-Stokes equations, $\bar{\mathbf{u}}$, tends to the continuous solution \mathbf{u} in the limit of zero mesh size, $\Delta \rightarrow 0$. This ensures that a DNS is recovered as the mesh is

²Note that explicit filtering can be applied in the former case of filter width tied to mesh size, but that it doesn't have to be.

³For a complete list of properties that an LES model should obey, see Berselli et al. (2006).

refined. Furthermore, some level of accuracy is required such that $\|\bar{\mathbf{u}} - \mathbf{u}\| \rightarrow 0$ as $\Delta \rightarrow 0$ at some provable rate. Demonstrating this in practice is quite involved because $\bar{\mathbf{u}}$ changes considerably with reducing mesh size as more and more turbulent scales are included in the LES solution. A discussion of LES error estimation and control is presented in §5.4.

5.1.5 Smagorinsky LES Model

The most widely used LES model for τ_{ij}^R is that of Smagorinsky (1963). Like many LES models it is based on the Boussinesq eddy-viscosity hypothesis (cf. §3.30). The model is derived from the assumption of an instantaneous balance between the production and dissipation of SFS turbulent kinetic energy in isotropic turbulence (Germano et al., 1991). An eddy viscosity ν_τ is introduced to account for the dissipative effect of the SFS motions:

$$\nu_T = C_S^2 \bar{\Delta}^2 |\bar{S}|, \quad (5.2)$$

where C_S is the Smagorinsky coefficient, $\bar{\Delta}$ is the filter width and $|\bar{S}|$ is the rate of strain modulus:

$$\bar{S}_{ij} = \frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right), \quad (5.3)$$

$$|\bar{S}| = \sqrt{2\bar{S}_{ij}\bar{S}_{ij}}. \quad (5.4)$$

Then the deviatoric SFS stress tensor is defined as:

$$\tau_{ij}^R = \tau_{ij} - \frac{1}{3}\tau_{kk}\delta_{ij} = -2\nu_T\bar{S}_{ij}. \quad (5.5)$$

As outlined in §3.3.4, the isotropic or spherical stress tensor τ_{ii} is included implicitly in the pressure. This causes problems if we want to know the exact pressure in an LES solution, which are described in §5.1.6.5.

The constant C_S was originally tuned by Lilly (1967) to satisfy the $-5/3$ power law of the inertial range of the Kolmogorov energy spectrum in high-Re flows, assuming that $\bar{\Delta}$ is in the inertial range, resulting in the value $C_S = 0.17$. The generally used value is $C_S = 0.1$ for free shear and channel flows, though as will be seen in the next section, it should be a flow-dependent variable.

5.1.6 Avenues of Improvement

The Smagorinsky LES model may be the most widely used but it has several serious deficiencies. No allowance is made for flow inhomogeneity since the

Smagorinsky coefficient C_S does not vary spatially. The coefficient is in fact flow-dependent and therefore usually not known in advance, except in the simplest of flows (Sagaut and Ménéveau, 2006). The model can be too dissipative at walls, resulting in under-prediction of the Reynolds stresses at the wall, leading to over-predicted mean velocity further away from the wall (Meneveau et al., 2005). For example, the Smagorinsky model applied to the atmospheric boundary layer over-predicted mean shear in the surface layer, preventing the growth of instabilities (Chen et al., 2008). Additional near-wall models are often used to give the correct scaling of the coefficient (Pope, 2000).

Nor is there any provision for anisotropy of the SFS since the eddy viscosity is scalar (i.e. the same in all directions at each point). The model does not behave well in turbulent flows where anisotropy is strong, which is the condition of most real flows (Pope, 2000). The different turbulent lengthscales in different directions are smeared out by using a single ‘average’ value, effectively isotropising the SFS and breaking the cascade of anisotropy (Gallerano et al., 2005).

Criticisms are levelled at the Boussinesq eddy-viscosity hypothesis in general. The specification of a scalar eddy viscosity implies that the eigenvectors of the stress tensor τ_{ij} are aligned with the rate-of-strain tensor S_{ij} . It has been shown that they are not always aligned (Tao et al., 2000; Meneveau and Katz, 2000). In inhomogeneous flows the smallest scales contain two components: a rapid part explicitly dependent on the mean velocity gradient, and an independent slow part. The rapid part cannot be modelled by eddy-viscosity models (i.e. models based on the Boussinesq hypothesis), but is important in non-equilibrium turbulence when production is greater than dissipation, or with large filter size (Liu et al., 1999).

A further problem with the eddy-viscosity hypothesis is that it requires the SFS kinetic energy production to be positive, i.e. energy is constrained to transfer from resolved scales to SFS and not the other way. It has been shown that there is in fact a flow of energy in the opposite direction, known as *backscatter*, and eddy-viscosity-based models fail to account for this mechanism (Piomelli et al., 1991; Horiuti, 1989). In flows with a significant power input (i.e. energy flow from large to small scales is dominant) and over moderate time-scales, which are typical of industrial problems, this inability to model backscatter is not detrimental (Berselli et al., 2006). For problems in which the energy balance is more delicate, or in which we want to know the velocity and pressure over a long time interval, eddy-viscosity models may not be appropriate and one should look to alternatives which base the eddy viscosity on the finest resolved scales such as variational multiscale simulation (VMS) (Berselli et al., 2006).

Five strategies for creating improved eddy-viscosity-based LES models, which can be used together, are outlined below.

5.1.6.1 Dynamic Method

The ground-breaking dynamic method (Germano et al., 1991) addresses the problem of flow-dependence of the Smagorinsky coefficient by relating it dynamically to the local flow conditions. It has rapidly become a widely-used and trusted model for complex flows (Piomelli, 1999). Among others, Lilly (1992), Ghosal et al. (1995), Meneveau and Lund (1996) and Gallerano et al. (2005) have modified or extended the Germano dynamic method to resolve some of its mathematical inconsistencies and improve its behaviour. The method is described in full in §5.2.2.

5.1.6.2 Adaptive Meshing

Grid generation is one of the most difficult problems in simulating complex geometry, particularly if conventional structured grids are used (Cant, 2002). Unstructured meshes can be more simply designed to fit any arbitrary shape. To reduce the number of nodes in a mesh and thus the computational effort required to solve an LES, the mesh should be refined only where greater accuracy is needed, e.g. in shear layers, and coarsened where the solution is less sensitive to mesh size. Unstructured mesh adaptivity achieves this by adjustment of the sizes and shapes of elements in an unstructured mesh to meet a solution accuracy criterion. The method has been implemented in the general-purpose CFD code Fluidity (Pain et al., 2001) and has been used to reduce the effort required to solve complex problems such as bluff body flow with LES (Bentham, 2003) and gravity currents (Hiester et al., 2011). A full description of the method and its use with LES are given in §5.4.

Another method for generating flow-following meshes known as the Arbitrary Lagrangian-Eulerian (ALE) method has been used successfully in some problems (Darlington et al., 2002) but may be difficult to use in complex geometry.

5.1.6.3 Anisotropic Filtering

Let us assume that the implicit LES strategy is followed. The filter width $\overline{\Delta}$ is then a scalar multiple of element size Δ . We wish to obtain an accurate solution with the least amount of computational effort so we use an inhomogeneous unstructured mesh containing widely varying element sizes and aspect ratios. The question arises of what an appropriate measure of element size, hence filter size, actually is in such a mesh. The conventional

5.1 Introduction

approach is to take the element size as the cube root of the volume (in 3D) as proposed by [Deardorff \(1970\)](#), for Cartesian meshes:

$$\Delta = (\Delta_1 \Delta_2 \Delta_3)^{1/3}, \quad (5.6)$$

and for unstructured meshes:

$$\Delta = V^{1/3}, \quad (5.7)$$

where Δ_i signifies the edge length in a Cartesian direction and V is the element volume. In the finite element method the volume of an element can be found from the sum of the quadrature weights, i.e. integrating unity over the element:

$$V = \sum_{i=-n}^n W_i, \quad (5.8)$$

where n and W_i are determined by the quadrature rule.

High-aspect-ratio elements pose a problem because the cube root of the volume does not necessarily resemble the actual dimensions of the element in any direction. Therefore, the resolution capabilities of the mesh are not made best use of and furthermore, the implicit assumption is made that the SFS motions are isotropic and that the element is a cube. Secondly, basic geometry illustrates that the relationship between volume and characteristic size (edge length, diameter etc.) differs between shapes.

A *tensor* filter width accounts for mesh anisotropy. In the conventional case of anisotropic (stretched) Cartesian meshes, a tensorial ‘box’ filter was used by [Bardina et al. \(1980\)](#), who proposed that the moment of inertia I_{ij} was a convenient tensorial definition of the size of a parallelepiped element:

$$I_{ij} = \frac{1}{V} \int_V \Delta_i \Delta_j dV. \quad (5.9)$$

For a cuboid the tensor has only diagonal terms:

$$I_{ij} = \frac{2}{3} \begin{bmatrix} \Delta_1^2 & 0 & 0 \\ 0 & \Delta_2^2 & 0 \\ 0 & 0 & \Delta_3^2 \end{bmatrix}, \quad (5.10)$$

where Δ_i is the element size in direction i . The tensorial filter width (5.9) is decomposed into spherical (isotropic) and deviatoric (anisotropic) parts akin to the SFS stress tensor:

$$I_{ij} = I\delta_{ij} + I_{ij}, \quad (5.11)$$

$$I = \frac{1}{3} I_{kk} = \frac{1}{3} (\overline{\Delta_1^2} + \overline{\Delta_2^2} + \overline{\Delta_3^2}), \quad (5.12)$$

Alternatively, [Scotti et al. \(1993\)](#) proposed a filter width correction factor for anisotropic Cartesian meshes which, it was claimed, improved estimation of the SFS dissipation rate, but neither of these proposals is applicable to unstructured meshes. [Oshima et al. \(1997\)](#) found that on stretched Cartesian meshes the Deardorff definition of scalar filter width (5.7) gave similar predictions of $C_S^2 \overline{\Delta}^2$ to calculating independent filter widths in each Cartesian direction. From this result they rationalised the use of Deardorff's definition on unstructured meshes, but the validity of extending it to such a different mesh topology is questionable.

A tensorial filter width definition for tetrahedral elements based on a metric tensor was proposed in [Bentham \(2003\)](#) to form the basis of a tensorial Smagorinsky model (cf. §5.2.1). The metric tensor \mathcal{M} , used in the unstructured mesh adaptivity method in Fluidity (see §3.6.1), encodes mesh sizing and orientation:

$$\mathcal{M} = \mathbf{V}^T \mathbf{\Lambda} \mathbf{V} = \mathbf{V}^T \begin{bmatrix} \lambda_\zeta & 0 & 0 \\ 0 & \lambda_\eta & 0 \\ 0 & 0 & \lambda_\xi \end{bmatrix} \mathbf{V}, \quad (5.13)$$

where λ_i are the eigenvalues of \mathcal{M} , \mathbf{V} is a rotation matrix consisting of the normalised eigenvectors of \mathcal{M} , $\mathbf{\Lambda}$ is a diagonal matrix of the eigenvalues of \mathcal{M} and the transformation $\mathbf{V}^T \dots \mathbf{V}$ is the rotation from local (element) to global (Cartesian) coordinates. The eigenvalues λ_i correspond to h_i^{-2} where h_i are the edge lengths of a tetrahedral element. Inverting this metric defines a tensor, the squared size of a generic tetrahedral element $(\Delta^2)_{ij}$:

$$(\Delta^2)_{ij} = \mathcal{M}^{-1} = \mathbf{V}^{-1} \mathbf{\Lambda}^{-1} (\mathbf{V}^T)^{-1} = \mathbf{V}^T \mathbf{\Lambda}^{-1} \mathbf{V} = \mathbf{V}^T \begin{bmatrix} h_\zeta^2 & 0 & 0 \\ 0 & h_\eta^2 & 0 \\ 0 & 0 & h_\xi^2 \end{bmatrix} \mathbf{V}. \quad (5.14)$$

Then the tensorial filter size is given by

$$(\overline{\Delta}^2)_{ij} = \alpha^2 (\Delta^2)_{ij}. \quad (5.15)$$

5.1.6.4 Anisotropic, Nonlinear and Mixed Models

Even if an LES is well-resolved according to the definition of [Pope \(2000\)](#) given above, there is some evidence that the small scales of turbulence are not as isotropic as predicted by the Kolmogorov hypotheses ([Fureby and Grinstein, 2002](#)). In the case of an under-resolved LES it is to be expected that the SFS are anisotropic because the anisotropy of the largest scales cascades down to intermediate scales ([Sagaut, 2006](#)). Some way of accounting for this

in the model, i.e. for a departure from local isotropy, is desirable to improve model accuracy and practicality. While the dynamic method of [Germano et al. \(1991\)](#) captures the spatial variations (*inhomogeneity*) of the SFS by means of a spatially-varying coefficient, there is no commonly accepted way to properly account for anisotropy in an LES model. A discussion of how SFS anisotropy can be represented in the eddy viscosity is given by [Carati and Cabot \(1996\)](#).

The scale-similarity LES model of [Bardina et al. \(1980\)](#) is based on the *scale-similarity assumption* that the important interactions between the resolved scales and SFS involve the largest unresolved and smallest resolved scales, and that these scales are self-similar in shape. The model simply ignores the effects of the cross and Reynolds tensors C_{ij} and R_{ij} in (3.26), specifying that

$$\tau_{ij}^R = L_{ij} - \frac{1}{3}L_{kk}\delta_{ij}. \quad (5.16)$$

However, it has been found that the scale-similarity model is insufficiently dissipative, i.e. it may not provide an adequate energy drain from the resolved scales to SFS ([Gallerano et al., 2005](#)).

[Bardina et al. \(1980\)](#) also proposed a nonlinear Smagorinsky eddy-viscosity model based on the inertia-tensor filter given in (5.11):

$$\begin{aligned} \tau_{ij}^R = & C_1 I |\bar{S}| \bar{S}_{ij} \\ & + C_2 |\bar{S}| (I_{ik} \bar{S}_{kj} + I_{jk} \bar{S}_{ki} - \frac{1}{3} I_{lk} \bar{S}_{kl} \delta_{ik}) \\ & + C_3 |\bar{S}| / I (I_{ik} I_{kl} \bar{S}_{kl} - \frac{1}{3} I_{mk} I_{ml} \bar{S}_{kl} \delta_{ij}), \end{aligned} \quad (5.17)$$

where C_1, C_2, C_3 are constants ([Sagaut and Méneveau, 2006](#)). The first term is simply the Smagorinsky model with a scalar filter width given by $\bar{\Delta} = \frac{1}{3}(\bar{\Delta}_1^2 + \bar{\Delta}_2^2 + \bar{\Delta}_3^2)$, while additional terms account for anisotropy. The model was found to be superior to the conventional Smagorinsky model, resulting in less variation of best-fit Smagorinsky coefficients for different shapes of mesh. It has been combined with the Bardina scale-similarity model, (5.16), to form a mixed model which is reported by [Zang et al. \(1993\)](#) to combine the advantages of both models and provide a sufficient energy drain.

A simple anisotropic model, extending the original Smagorinsky model to three directions, was developed by [Zahrai et al. \(1995\)](#):

$$(\nu_T)_k = C_S (\bar{\Delta}_1 \bar{\Delta}_2 \bar{\Delta}_3)^{2/9} (\bar{\Delta}_k)^{4/3} \langle 2\bar{S}_{ij} \bar{S}_{ij} \rangle^{3/2}. \quad (5.18)$$

This model was tested successfully in turbulent channel flow but is only applicable to Cartesian meshes.

Bentham (2003) proposed a tensorial Smagorinsky-type model (cf. §5.2.1) for inhomogeneous unstructured tetrahedral meshes based on the anisotropic filter width given by (5.15). This tensorial LES model with unstructured mesh adaptivity produced slightly superior results to the isotropic Smagorinsky model in simulations of a cube in cross-flow. Good agreement with DNS and experimental data was obtained using the tensorial LES model with unstructured mesh adaptivity in simulations of plane channel flow and bluff body flow by Pavlidis et al. (2009). In the more complex case of flow at a street junction, accuracy was compromised by insufficient resolution and the use of no-slip boundary conditions (Aristodemou et al., 2009).

The problem of incorrect alignment of the stress and strain tensors can also be addressed by using a tensorial eddy viscosity. Multiplication of the strain tensor by an anisotropic viscosity tensor results in a stress tensor with wholly different eigenvectors, not necessarily aligned with the strain tensor. The required degree of rotation is not known from physical principles; experimental results (Tao et al., 2000) and DNS (Horiuti, 2003) have only provided empirical relationships although the latter found that the rotation was primarily due to a term proportional to $(\bar{S}_{ik}\bar{\Omega}_{kl} + \bar{S}_{jk}\bar{\Omega}_{kl})$, where S_{ij} is the strain rate tensor and $\Omega_{ij} = \left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i}\right)$ is the rotation rate tensor.

An alternative method for obtaining the correct alignment of stress and strain is via a *nonlinear eddy-viscosity* model (Pope, 2000). A general relationship between stress, strain rate and rotation rate was derived by Lund and Novikov (1992) in terms of 11 independent combinations of strain and rotation rates, of which the above term is one, although no more than six are retained to avoid excessive computational cost and bad numerical behaviour (Wang and Bergstrom, 2005). The scale-similarity model of Bardina et al. (1980) also included several of the 11 terms. For a full analysis of the angles between eigenvectors of the stress tensor and various components of a dynamic nonlinear model (DNM) incorporating four of the 11 terms, see Wang and Bergstrom (2005). DNM and standard dynamic LES model results in Couette flow were comparable. However, the DNM was more robust, did not require the stabilisation measures that the standard dynamic LES model does, and provided more information on the flow of energy to and from the SFS. Furthermore, the angles between the stress and strain tensors were measured and showed significant differences between the free-stream and near-wall regions, providing justification for an approach which allows the angles to be controlled.

It is more common to simply include terms up to fourth-order in $\bar{\mathbf{u}}$ from the scale-similarity model (5.17), e.g. Dantinne et al. (1997), Clark et al. (1979). Fourth-order dissipation has a localised effect in wavenumber space,

concentrating on the highest resolved wavelengths near the cut-off frequency (Cerutti et al., 2000). At Imperial College, Candy (2008) extended the work of Bentham (2003) to use the anisotropic inhomogeneous filter with a fourth-order *hyperviscosity* model defined by taking the difference between two second-order discretisations. Tests of the method in the backward-facing step and an ocean gravity current showed that it was more scale-selective in its dissipative properties than a second-order scheme: mesh-scale structures were dissipated more strongly and large-scale structures less so.

5.1.6.5 Subfilter-Scale Kinetic Energy Estimation

When an LES model is based on the Boussinesq eddy-viscosity hypothesis (cf. §3.3.4) and the flow is incompressible, the spherical or isotropic SFS stress is not modelled. The reason for this is that $\nabla\bar{\mathbf{u}} + (\nabla\bar{\mathbf{u}})^T$ has zero trace. It has been mentioned in §3.3.4 that the spherical stress is added implicitly to the pressure which begs the question “How do we know its value?” The expression below is often cited (Pope, 2000):

$$\frac{1}{3}\tau_{ij}\delta_{ij} = \frac{2}{3}k, \quad (5.19)$$

where k is the sub-filter kinetic energy – but in an LES we do not know k unless we have an additional model for it such as the one-equation model (cf. §4.2).

In actual fact it is not necessary to know τ_{ii} but the reason why is seldom explicitly stated. When the Poisson equation for pressure is solved, subject to the constraint that the filtered velocity has zero-divergence, the pressure solution must implicitly include the spherical stress in order to satisfy the zero-divergence constraint (Vreman et al., 1994). In other words, if the deviatoric stress is added to the momentum equation then the pressure must be the modified pressure (3.14) in order to satisfy the filtered continuity equation (3.20b). The same applies to RANS eddy-viscosity models.

Therefore, in cases where we want to know the exact pressure, LES models based on the eddy-viscosity hypothesis are deficient (Vreman et al., 1994). It has been shown in channel flow simulations that the magnitude of the modification can be non-negligible when the ratio α is very large or when very close to the wall (Kaneda and Leslie, 1983). A method of estimating k might then be desirable. Vreman et al. (1994) derived a lower bound on k for the SFS stress τ_{ij} to be positive semi-definite and therefore realisable (cf. §5.1.4) if the Smagorinsky model and a non-negative filter (cf. §5.1) are used:

$$k \geq \frac{1}{2}\sqrt{3}C_S^2\bar{\Delta}^2|\bar{S}|. \quad (5.20)$$

The Yoshizawa model for k arises from this lower bound:

$$k = C_k \overline{\Delta}^2 |\overline{S}|, C_k \geq \frac{1}{2} \sqrt{3} C_S^2, \quad (5.21)$$

where Yoshizawa (1986) proposed the value $C_k = 0.0886$. Other models for the SFS turbulent kinetic energy are available; see, e.g. Berselli et al. (2006).

Further advantages can be gained by modelling k , including the ability to determine the global energy balance and the local flux of energy to *and from* the SFS (i.e. backscatter) (Gallerano et al., 2005). The Smagorinsky model by itself is unable to account for this process, as it assumes that energy only flows to the SFS from the resolved scales. The reverse direction is known as backscatter and is accounted for in methods such as EDQNM modelling which accounts for the quantity and spectrum of the kinetic energy fluxes (Sagaut and Méneveau, 2006). The localised dynamic LES model of Ghosal et al. (1995) also accounts for energy balance and fluxes.

5.1.6.6 Boundary Conditions

LES models generally incur errors near the wall in turbulent boundary layers because the integral scale of the flow becomes very small (Georgiadis et al., 2009). Provision of adequate mesh resolution becomes impractical beyond moderate Reynolds numbers. The Smagorinsky model is usually supplemented with wall functions similar to those used with RANS models (see §4.3.1) to damp the eddy viscosity near walls. The design of wall functions is a vast subject; Piomelli and Balaras (2002) provide a comprehensive review of the various options.

More sophisticated LES models can avoid the need for empirical near-wall adjustments. The dynamic method with the Smagorinsky model has been shown to give approximately the correct near-wall behaviour without wall functions (Germano et al., 1991). A more recent solution to the problem of insufficient near-wall resolution is hybrid LES-RANS modelling; see the review by Fröhlich and von Terzi (2008). LES is used in the free stream while the boundary layer flow is modelled with RANS, allowing grids to be coarser near the wall without loss of detail in the main flow. The main difficulty is matching the flow at the interface between the RANS and LES regions (Piomelli et al., 2003). Hybrid models have begun to be tested in nuclear engineering applications. Rolfo et al. (2010) simulated flow through a heated tube bundle with a hybrid RANS-LES model using the $v^2 - f$ RANS model of Laurence et al. (2005), finding that it over-predicted wall shear stress and was sensitive to the choice of blending function.

A different hybrid method, the hybrid zonal (two-layer) model of Balaras et al. (1996), uses simplified equations on a separate mesh near the wall with

weak interaction at the interface. The velocity term in the wall region comes from the outer flow and the resulting wall shear stress is fed back to the outer region. Results were good for a square duct and a rotating channel compared to the log law. [Diurno et al. \(2001\)](#) applied the model to a backward-facing step with good results.

In order to obtain correct flow statistics in a problem where turbulent flow enters the domain, some realistic turbulent behaviour should be prescribed as an inlet condition or enough of the upstream geometry should be included in the domain to allow the flow to develop turbulence. Various methods are available of differing complexity and sophistication as outlined in §3.4.2. The synthetic eddy method (SEM) of [Jarrin et al. \(2006\)](#) is adopted here for this purpose. SEM is a statistical reconstruction of a turbulent flow based on prescribed mean, Reynolds stress and lengthscale profiles obtained from DNS or experimental data. It has been implemented in Fluidity as described in [Pavlidis et al. \(2009\)](#).

5.2 A New Tensorial Dynamic LES Model

Several of the above improvements – the dynamic method, mesh adaptivity, anisotropic filtering and tensorial eddy viscosity – are here combined into one new model, the tensorial dynamic LES model. A tensorial filter width, related to the mesh size, is used in a tensorial eddy-viscosity model for the SFS tensor as proposed by [Bentham \(2003\)](#). It is also incorporated into the inverse Helmholtz filter operator as suggested in [Germano \(1986a\)](#) to be used as an explicit filter operator in the dynamic method. The result is a tensorial dynamic LES model which is designed for use with unstructured mesh adaptivity since it can represent the anisotropy of the elements. It is proposed that this method is ideal for high-Reynolds number flow in complex geometries owing to its ability to represent the anisotropy and inhomogeneity of the flow. By constructing anisotropic SFS stresses, the model is not bound by Kolmogorov’s isotropy hypothesis, and may therefore relax the resolution requirements of standard LES models which do assume isotropic SFS. Another consequence of the model is that the anisotropy cascade from resolved scales to SFS is maintained.

5.2.1 Tensor Eddy Viscosity

The tensor eddy viscosity LES model developed by [Bentham \(2003\)](#) is a variation of the standard Smagorinsky model (5.2) that can be applied consistently to unstructured triangular or tetrahedral meshes with finite element

discretisations. The filter width $\bar{\Delta}$ is redefined as a tensor $\bar{\Delta}_{ij}$ so that the model diffusion is scaled independently in each spatial direction and at each point in the domain. This is particularly useful in combination with unstructured mesh adaptivity where the mesh is refined or coarsened and stretched according to the local flow conditions. The model is written

$$\nu_{T,ij} = C_S^2 (\bar{\Delta}^2)_{ij} |\bar{S}|, \quad (5.22)$$

where $\nu_{T,ij}$ is a tensor eddy viscosity, C_S is the Smagorinsky coefficient and $(\bar{\Delta}^2)_{ij}$ is the squared tensor filter width in an unstructured mesh given by (5.15). Brackets have been used to clarify that $(\bar{\Delta}^2)_{ij}$ is different from $(\bar{\Delta}_{ij})^2 = \bar{\Delta}_{ij} \bar{\Delta}_{ij}$.

In order to include this tensor eddy viscosity in the Navier-Stokes equations the stress-strain relationship given by (5.5) has to be modified. Instead of writing the deviatoric SFS stress tensor directly, the divergence of the SFS stress, or *subfilter force*, which is the term actually included in the filtered momentum equation (3.27), is given by (Bentham, 2003):

$$\frac{\partial \tau_{ij}^R}{\partial x_j} = \frac{\partial}{\partial x_i} \left[\nu_{T,jk} \circ \frac{\partial \bar{u}_j}{\partial x_k} \right], \quad (5.23)$$

where \circ indicates a pointwise (Hadamard) product. This is equivalent to treating each component of the mesh size tensor as a separate scalar. Unlike the eddy-viscosity hypothesis, it does not constrain the stress and strain tensors to be aligned.

Unfortunately, the Hadamard product is not invariant. Given two tensors A and B and a transformation matrix R (e.g. a rotation or reflection), it can be shown by finding simple counter-examples that, in general,

$$R(A \circ B)R^T \neq (RAR^T) \circ (RBR^T). \quad (5.24)$$

That is, the transformed Hadamard product of A and B is not the same as the Hadamard product of two transformed tensors. This means that the LES model thus derived is not invariant under rotations, reflections and Galilean transformations, while the original Navier-Stokes equations are. In contrast, an LES model based on a scalar eddy viscosity and scalar filter width is invariant, since no Hadamard product is required. Whether this has bearing on the model accuracy has not been ascertained and remains a matter for further research.

5.2.2 Dynamic Method

The concept of the dynamic method is to derive an SFS stress term from the finest resolved scales, which are elucidated by twice filtering the velocity field

and comparing the first- and second-filtered fields (Germano et al., 1991). A spatially varying coefficient is calculated from this extra information, addressing the problem of the Smagorinsky model's constant coefficient. The idea actually predates the dynamic method: Bardina et al. (1980) described the difference between two filtered fields as a natural way to estimate the SFS by invoking the scale-similarity assumption (cf. §5.1.6.4) and also introduced the idea of a tensorial eddy viscosity.

The first or 'mesh' filtering $\overline{G}(\mathbf{u}) \rightarrow \overline{\mathbf{u}}$, with filter width given by (5.1), is either an unknown filtering operation implicit in the choice of discretisation and mesh, i.e. implicit LES, or an explicit operation on the discrete field as explained in §5.1.3. The second or 'test' filtering is an explicit operation (described in §5.1) on the first filtered field $\overline{\mathbf{u}}$ resulting in a twice-filtered velocity field $\widetilde{\mathbf{u}}$: $\widetilde{G}(\overline{\mathbf{u}}) \rightarrow \widetilde{\mathbf{u}}$, with filter width $\widetilde{\Delta}$ given by

$$\widetilde{\Delta} = \beta \overline{\Delta}, \quad \beta \geq 1. \quad (5.25)$$

The overbar notation is used for first-filtered quantities, the tilde for test-filtered quantities and the combination of overbar and tilde for twice-filtered quantities. Sequential application of filters, written as the convolution $\widetilde{\overline{G}} = \widetilde{G}\overline{G}$, can be treated as a single filter operation with a combined width of

$$\widetilde{\overline{\Delta}}^2 \approx \widetilde{\Delta}^2 + \overline{\Delta}^2 = (1 + \beta^2)\overline{\Delta}^2. \quad (5.26)$$

5.2.2.1 Modelled Terms

For each filter level an LES model is written. The mesh- and twice-filtered deviatoric SFS stresses are, respectively:

$$\tau_{ij}^R = \tau_{ij} - \frac{1}{3}\tau_{kk}\delta_{ij} = \overline{u_i u_j} - \overline{u_i}\overline{u_j}, \quad (5.27)$$

$$T_{ij}^R = T_{ij} - \frac{1}{3}T_{kk}\delta_{ij} = \widetilde{\overline{u_i u_j}} - \widetilde{\overline{u_i}}\widetilde{\overline{u_j}}. \quad (5.28)$$

Any eddy-viscosity model may be used to model the stress tensors, but in the original form the Smagorinsky model (5.2) is used:

$$\tau_{ij}^R = -2C_S^2 \overline{\Delta}^2 |\overline{S}| \overline{S}_{ij}, \quad (5.29)$$

$$T_{ij}^R = -2C_S^2 \widetilde{\overline{\Delta}}^2 |\widetilde{\overline{S}}| \widetilde{\overline{S}}_{ij}. \quad (5.30)$$

Intermediate turbulent stresses (i.e. between the two filter levels), known as the Leonard tensor⁴, are defined by the *Germano identity* (Pope, 2000):

$$L_{ij} = T_{ij} - \widetilde{\tau}_{ij} = \widetilde{\overline{u_i u_j}} - \widetilde{\overline{u_i}}\widetilde{\overline{u_j}}. \quad (5.31)$$

⁴A different but related form to the Leonard tensor in the triple decomposition, (3.26).

The deviatoric part of the Leonard tensor is written

$$L_{ij}^R = L_{ij} - \frac{1}{3}L_{kk}\delta_{ij} = T_{ij}^R - \widetilde{\tau}_{ij}^R. \quad (5.32)$$

The benefit of this identity is that the terms are known, whereas not all the terms of (5.27) and (5.28) are known. Substitution of (5.29) and (5.30) into (5.32) and contraction of both sides with the rate of strain \overline{S}_{ij} as proposed by Germano et al. (1991) gives

$$L_{ij}^R \overline{S}_{ij} = L_{ij} \overline{S}_{ij} = -2c_S \left(\overline{\Delta}^2 |\widetilde{S}| \widetilde{S}_{ij} \overline{S}_{ij} - \overline{\Delta}^2 |\widetilde{S}| \widetilde{S}_{kl} \overline{S}_{kl} \right), \quad (5.33)$$

where c_S and $\overline{\Delta}$ have been written outside of the filtering operation $\widetilde{\cdot}$ in the second term by assuming that they do not vary rapidly in space, and furthermore the coefficient has been redefined as $c_S = C_S^2$ to reflect the possibility of it being negative. Note that the isotropic part of L_{ij} has disappeared because $\overline{S}_{kk} = 0$ in an incompressible flow.

The coefficient c_S is then found from the following expression:

$$c_S(\mathbf{x}, t) = -\frac{1}{2} \frac{L_{ij} \overline{S}_{ij}}{\overline{\Delta}^2 |\widetilde{S}| \widetilde{S}_{kl} \overline{S}_{kl} - \overline{\Delta}^2 |\widetilde{S}| \widetilde{S}_{mn} \overline{S}_{mn}}, \quad (5.34)$$

and finally the eddy viscosity is given by the standard Smagorinsky formula:

$$\nu_T(\mathbf{x}, t) = c_S(\mathbf{x}, t) \overline{\Delta}^2 |\widetilde{S}|. \quad (5.35)$$

5.2.2.2 Inconsistencies

There are some difficulties with the dynamic method as written above. (5.34) is actually five independent equations for one unknown: the first inconsistency of the dynamic method as originally conceived. The second inconsistency is the assumption in (5.33) that c_S does not fluctuate strongly in space, which is mathematically inconsistent and not always true (Meneveau et al., 2005). However, Lilly's method for determining c_S (see below) renders this assumption consistent (Ghosal et al., 1995). The assumption that $\overline{\Delta}$ does not vary rapidly is valid on inhomogeneous meshes only if the filter width can be controlled by careful mesh design (the implications for mesh adaptivity are considered later).

A third inconsistency is that the denominator of (5.34) can become very small which causes numerical instability. To stabilise the calculation Germano et al. (1991) locally averaged the quantities along homogeneous planes

in the fluid (e.g. parallel to the wall in fully-developed channel flow), so that the dynamic Smagorinsky coefficient is calculated from

$$c_S(\mathbf{x}, t) = -\frac{1}{2} \frac{\langle L_{ij} \bar{S}_{ij} \rangle}{\widetilde{\Delta}^2 \langle |\widetilde{S}| \widetilde{S}_{kl} \bar{S}_{kl} \rangle - \bar{\Delta}^2 \langle |\bar{S}| \bar{S}_{mn} \bar{S}_{mn} \rangle}, \quad (5.36)$$

where $\langle \dots \rangle$ denotes local averaging. However, homogeneous directions may not be known *a priori*. To address this last point, Meneveau and Lund (1996) came up with a Lagrangian path averaging technique which made the Germano dynamic method applicable to arbitrary geometry in which no homogeneous directions exist. Neither of these *ad hoc* methods is trivial or cheap to compute and it is advantageous to avoid averaging procedures if possible.

Germano et al. (1991) used (5.35) and (5.36) in a turbulent channel flow simulation with filtering performed only over the wall-parallel directions with good results. Balaras et al. (1995) found that the dynamic method was able to recreate the asymptotic behaviour of SFS stress near walls, observed in experiments and DNS, without resolving into the viscous sublayer and without additional damping functions.

5.2.2.3 Lilly's Modified Dynamic Method

An alternative to averaging for removal of the numerical instability was introduced by Lilly (1992). The Leonard tensor is written

$$L_{ij} = 2c_S M_{ij}, \quad (5.37)$$

$$M_{ij} = \widetilde{\Delta}^2 |\widetilde{S}| \widetilde{S}_{ij} - \bar{\Delta}^2 |\bar{S}| \bar{S}_{ij}. \quad (5.38)$$

The coefficient is found by a least-squares error minimisation. Defining the error as

$$Q = (L_{ij} - 2c_S M_{ij})^2, \quad (5.39)$$

and setting $\partial Q / \partial c_S = 0$, a modified expression for c_S is obtained:

$$c_S(\mathbf{x}, t) = \frac{1}{2} \frac{L_{ij} M_{ij}}{M_{kl} M_{kl}}. \quad (5.40)$$

Even with this improved estimation method, c_S has been shown to vary rapidly in space and to become locally negative representing backscattering of turbulent kinetic energy from the SFS to the resolved scales (Lund and Novikov, 1992). However, negative values of c_S (hence eddy viscosity) remained negative for long periods of time which caused exponential growth

of the local velocity field. Whilst some mechanism to account for energy backscatter is desirable, this long-lasting negativity simply causes simulations to blow up (Metais and Lesieur, 1996).

To fix this problem, Lilly (1992) ‘clipped’ the coefficient so that $c_S \geq 0$ and $c_S \leq c_{S,max}$, stating that locally unbounded values could be truncated without losing the conceptual advantage of the formulation. Enforcing $c_S \geq 0$ prevents backscatter and the associated problems. A maximum value of $c_{S,max} = C_S^2 = 0.17^2 \approx 0.03$ is obtained theoretically from the Kolmogorov energy spectrum (Lilly, 1967), and has been confirmed by experiments (Clark et al., 1979; Kwak et al., 1975), although Deardorff (1970) and Schumann (1975) found $c_{S,max} = 0.01$.

Neither clipping nor averaging are ideal solutions since they are *ad-hoc* operations which weaken the robustness and physical validity of the dynamic method. Clipping has the side effect of making the model irreversible, whereas the SFS stresses in the exact Navier-Stokes equations are reversible, meaning that the evolution of the velocity is reversed if its sign is changed in the absence of molecular viscosity (Carati et al., 2001). In this work, the clipping approach is taken for its simplicity in comparison to averaging. A value of $c_{S,max} = 0.04$ is selected to allow for some local deviation from the theoretical maximum value, and backscatter is prevented by clipping $c_S \geq 0$.

5.2.2.4 Backscatter Modelling

Backscatter is a potentially desirable property of the model as long as the model still satisfies the ideal model properties given in §5.1.4, i.e. it is bounded and realisable. Germano et al. (1991) reported that the stresses were closer to DNS as a result of including backscatter. The Smagorinsky model by itself has no knowledge of the amount of SFS energy so it is unable to halt the reverse flow of energy once the SFS energy reserves are depleted, therefore an alternative LES model is needed if backscatter is to be incorporated properly.

Ghosal et al. (1995) proposed modifications to the dynamic method to account for energy fluxes, among other improvements which addressed the inconsistencies in the dynamic method, but at the expense of model programmability (Sagaut, 2006). An equation was solved for the SFS kinetic energy k_{SFS} which allowed c_S to become negative. With an eddy-viscosity LES model dependent on k_{SFS} , their method allowed stable energy flow in both directions while the total energy monotonically decreased due to viscous effects. However, the exact nature of backscatter is difficult to model and sensitive to the filter employed (Ghosal et al., 1995).

5.2.2.5 Self-Similarity and Scale Invariance

The dynamic method assumes self-similarity in the scales of motion in which the first and second filter widths lie, i.e. both scales lie within the inertial range of the turbulent cascade, where the Kolmogorov -5/3 gradient (3.5) is expected to apply (Carati and van den Eijnden, 1997). This allows the same c_S to be used in both LES models. A consequence of the assumption of self-similarity is that the dynamic method is restricted to fine meshes in high Reynolds number flows because of the requirement to have both filter widths sufficiently far down in the inertial range. Another resolution constraint is placed by the Smagorinsky models, (5.29), which assume isotropy of the SFS. Obviously these constraints are somewhat limiting when simulating complex high-Reynolds number flows. Avoiding having to make either the scale-similarity or local isotropy assumptions may allow coarser meshes to be used. In this case it is sensible to use a technique such as mesh adaptivity in order to control the solution error. A full discussion of error estimation for LES is given in §5.4.

Various methods have been developed to avoid the assumption of scale-similarity in the dynamic method, usually by making c_S scale-dependent. One way of addressing this is to assume a power-law variation: $c_{S,\Delta} = c_{S,\alpha\Delta}(\Delta/\alpha\Delta)^n$, with an additional equation for the power n (Porte-Agel et al., 2000). Meneveau et al. (2005) found that this scale-dependent dynamic method was superior to the conventional scale-invariant dynamic Smagorinsky model, and to the non-dynamic Smagorinsky LES model with wall damping functions, in reproducing the log-law region in high-Reynolds number boundary layers.

Assumption of self-similarity of the turbulent scales requires the filters to be self-similar too. In turn this requires knowledge of the shape of the filters but if the mesh filter is implicit then its shape is generally hard to define. Carati and van den Eijnden (1997) proposed a way to avoid this conundrum by formulating a dynamic method that was independent of the filter type using special sets of self-similar filters.

Either implicit filtering on the grid and test filtering with another filter, or explicit filtering using two (known) similar filter kernels is possible and the choice has bearing on the self-similarity assumption. In the first approach the shape of the first filter is implicit in the discretisation and the mesh construction. In the special case of a Cartesian grid with central differencing, the implicit numerical filter is identical to the box filter of width 2Δ where Δ is the grid size, and applying the top hat filter as a test filter satisfies filter self-similarity (Tejada-Martinez and Jansen, 2004).

In the second approach, both filters are explicit operations which can be

chosen to satisfy self-similarity (Tejada-Martinez and Jansen, 2004). This latter method was used by Germano et al. (1991) in turbulent channel flow with the sharp spectral cut-off filter at both levels. Two different definitions of filter width were tested and the dynamic model constant c_S was reported to be slightly sensitive to the choice in channel flow tests. Tejada-Martinez and Jansen (2003) tested the model on regularly connected (structured) meshes and computed filter widths from the average radial wavenumber associated with an isosurface of the filter's transfer function (cf. §5.3.1.2). They found that the exact choice of filter shape did not significantly affect the value of c_S and that the ratios α and β were more significant. Tejada-Martinez and Jansen (2006) used the top-hat filter in the dynamic method on hexahedral, tetrahedral and wedge-shaped finite elements, finding again that c_S was not sensitive to the filter shape, as long as its width was calculated consistently by this method. The effects of implicit vs. explicit first filter, filter shape and filter width ratios on the results of LES of a 3D backward-facing step are investigated in Chapter 6.

5.2.2.6 Filter Width Ratio

The ratios of filter widths α and β may significantly affect results and the optimal values may differ for a particular simulation and/or discretisation. Tejada-Martinez and Jansen (2003) found, by trial and error, that $\alpha = 1.5$ gave the best fit to experimental data of decaying isotropic turbulence when using regularly-connected hexahedral elements, piecewise trilinear basis functions, one-point quadrature and the top hat filter. Germano et al. (1991) found that $\alpha = 2$ gave the best prediction of SFS shear stress and dissipation in turbulent channel flow at $Re=3300$. However, in LES of higher- Re channel flow, the mean and rms velocities were insensitive to changes in α in contrast to their expectations. Tejada-Martinez and Jansen (2004) tested different values of α , finding that values of three or four led to over-prediction of energy in the inertial range. Oshima et al. (1997) used Deardorff's definition of scalar filter width (5.7) in finite-difference simulations of decay of homogeneous turbulence and showed that the optimum values of the filter width ratios were⁵ $\alpha = \sqrt{2}$ and $\beta = \sqrt{3}$. It is desirable to find a less *ad hoc* method of establishing the optimal value of α and β , the only input variables to the dynamic method. For example, Tejada-Martinez and Jansen (2004) developed a method for dynamically estimating the filter width ratio when using an implicit first filter.

An alternative view of the dynamic method is that it tunes the dimen-

⁵ $(\tilde{\Delta}/\bar{\Delta})^2 = 4$ and $(\tilde{\Delta}/\Delta)^2 = 6$ in the notation used in Oshima et al. (1997).

sionless product $c_S\alpha^2$, implying that a decrease in α leads to an increase in c_S and the LES solution is only sensitive to the product (Laurence, 2013). Therefore, there should be no reason to not select $\alpha = 2$ to satisfy the Nyquist theorem, and let the model choose a corresponding value of c_S . However, the results presented in Chapter 6 contradict this conclusion, finding that $\alpha = 1$ is better than $\alpha = 2$ with both implicit and explicit first filters. This suggests that the reciprocal relationship of α and c_S may not be strictly true, if only for the model developed here. The reasons for this are as yet unknown and the matter requires further investigation.

5.2.2.7 Interaction with Numerical Dissipation

Najjar and Tafti (1996) found that the dynamic method had greater influence when used with a non-dissipative numerical method (central difference scheme) versus a stabilised method (upwinding), though the authors think that this phenomenon is a “numerical shortcoming [rather] than a reflection on the model itself”. Tejada-Martinez and Jansen (2005) found that the dynamic method was sensitive to numerical dissipation introduced by the SUPG scheme (cf. §3.5.6) simply because numerical dissipation lowered the strain rate magnitude, though the adjustment was not perfect. They redefined the dynamic Smagorinsky coefficient based on the difference between the SUPG and LES model dissipation rates. Garmann et al. (2012) performed high-resolution simulations of aerofoils with and without a dynamic LES model, finding that numerical dissipation inherent in the high-order discretisation scheme provided sufficient stability in this case.

5.2.3 Tensorial Dynamic Method

It is logical to combine the benefits of the dynamic method with an LES model capable of representing SFS anisotropy. One way this has been achieved is through a dynamic mixed model, e.g. using the dynamic method with the sum of Bardina scale-similarity and Smagorinsky models (Zang et al., 1993; Liu et al., 1994). Alternatively, Vreman et al. (1996) applied the dynamic method to the Clark model (another tensorial model), and Winckelmans et al. (1998) applied it to the nonlinear model of Leonard (1974) (similar to the Bardina scale-similarity model).

In the standard dynamic method with the Smagorinsky model a scalar $\overline{\Delta}^2$ is used, resulting in a scalar eddy viscosity (5.35). Here the dynamic method is implemented with the tensor eddy viscosity Smagorinsky model of Bentham (2003) related to a tensorial definition of filter width $(\overline{\Delta}^2)_{ij}$

suitable for unstructured tetrahedral meshes (cf. §5.2.1). Certain changes to the dynamic method are necessary to accommodate a tensor filter width.

The SFS stress tensors at both filter levels are redefined using (5.23):

$$\frac{\partial \tau_{ij}^R}{\partial x_j} = \frac{\partial}{\partial x_i} \left[c_S |\overline{S}| (\overline{\Delta}^2)_{jk} \circ \frac{\partial \overline{u}_j}{\partial x_k} \right], \quad (5.41a)$$

$$\frac{\partial T_{ij}^R}{\partial x_j} = \frac{\partial}{\partial x_i} \left[c_S |\widetilde{S}| (\widetilde{\Delta}^2)_{jk} \circ \frac{\partial \widetilde{u}_j}{\partial x_k} \right]. \quad (5.41b)$$

The deviatoric Leonard tensor (5.37) is redefined as

$$L_{ij} - \frac{1}{3} L_{kk} \delta_{ij} = -2c_S (|\widetilde{S}| \widetilde{S}_{ij} \circ (\widetilde{\Delta}^2)_{ij} - |\overline{S}| \overline{S}_{kl} \circ (\overline{\Delta}^2)_{kl}). \quad (5.42)$$

The dynamic model coefficient in the least-squares form of Lilly (1992) is then given by

$$c_S = \frac{1}{2} \frac{L_{ij} M_{ij}}{M_{kl} M_{kl}}, \quad (5.43a)$$

$$M_{ij} = |\widetilde{S}| \widetilde{S}_{ij} \circ (\widetilde{\Delta}^2)_{ij} - |\overline{S}| \overline{S}_{kl} \circ (\overline{\Delta}^2)_{kl}. \quad (5.43b)$$

With (5.22) for the tensorial eddy viscosity and (5.23) for the SFS stress gradient we have an LES model capable of representing anisotropic inhomogeneous sub-filter scales on an unstructured mesh.

5.2.4 Improved Filter Width Definitions

The tensor definition of filter width described in §5.2.1 has been implemented and tested with the tensorial dynamic LES model in the code Fluidity. For comparison purposes a scalar definition has also been implemented. It is defined as the cube root of the element volume (5.7), though this has no physical basis in unstructured mesh. Validation of the model in the 3D backward-facing step (cf. Chapter 6) found that the tensor filter width led to larger values of Smagorinsky coefficient than the scalar width, causing excessive damping of turbulence, possibly because the scalar filter width is smaller than the tensor magnitude.

As a result, it was decided to scale the tensor filter width so that it had equal magnitude to the scalar width by some appropriate definition of magnitude. The definition chosen was the Frobenius norm, which for an $m \times n$ matrix A with components a_{ij} is

$$\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2}. \quad (5.44)$$

Then the new scaled-down tensor width is given by

$$(\overline{\Delta}^2)_{ij} = \alpha^2 \left(\frac{V^{2/3}}{\|\mathbf{V}^T \mathcal{M}^{-1} \mathbf{V}\|_F} \right) \mathbf{V}^T \mathcal{M}^{-1} \mathbf{V}. \quad (5.45)$$

Results obtained with the normalised tensor filter width were improved and compared well with the scalar filter, as seen in Chapter 6. Normalisation may also isolate the effect of tensorial filter anisotropy. However, different scalar and tensor definitions could be considered; in Chapter 8 this topic is considered for future research. It is also possible that different definitions will be superior in different flow conditions.

5.3 A Tensorial Differential Filter For LES

Filtering for LES has already been introduced in §3.3.3 where the idea of an integration operation on the solution field to remove small scales beyond a threshold was outlined. In §5.2 the filter width was used extensively in the formation of LES models and was related directly to the mesh size. Distinctions were made between an implicit filter operation (equivalent to discretisation of a continuous field on a computational mesh) and an explicit filtering operation (actually performing a weighted averaging or equivalent operation).

In this section filtering is described in detail, including commutation errors induced by filtering with spatially varying filter width. Then the conventional inverse Helmholtz filter is introduced, modified to accept the tensorial filter width definition defined above, and implemented in finite-element form before verifying its properties in a series of tests.

5.3.1 Introduction to Filtering

5.3.1.1 Types of Filter

Three types of filter operator denoted are in use in LES codes: *integral* filters defined by (3.17), *differential* filters which solve a differential equation for the filtered variable, and *discrete* or *implicit* filters which project the solution field onto a finite set of basis functions. All are associated with a filter kernel G in terms of filter width $\overline{\Delta}$ and distance from the origin $\mathbf{x} - \xi$. It is possible to describe both integral and differential forms in terms of discrete projection; see for example Manica (2006).

Integral filter operators $\mathcal{G}(\mathbf{u}) \rightarrow \overline{\mathbf{u}}$ are defined such that:

$$\overline{\mathbf{u}}(\mathbf{x}, t) = \mathcal{G}(\mathbf{u}(\mathbf{x}, t)) = \int_{-\infty}^{\infty} G(\mathbf{x} - \xi) \mathbf{u}(\xi, t) d\xi, \quad (5.46)$$

in which the filter kernel G is normalised, that is, $Gc = c$ for any constant solution $\mathbf{u} = c$. The kernel is assumed to be localised as a function of $\mathbf{x} - \xi$, and it is associated with a filter width $\bar{\Delta}$. The simplest integral filter kernel is the top-hat or box:

$$G(\mathbf{x}) = \begin{cases} \bar{\Delta} & \text{if } |x_i| < \bar{\Delta}_i/2 \\ 0 & \text{otherwise} \end{cases}, \quad (5.47)$$

where $\bar{\Delta}_i$ is the filter width in direction x_i and $\bar{\Delta}$ is the average filter width. On structured Cartesian meshes it is easy to implement with the average filter width given by, for example, the Deardorff definition (5.7), i.e. $\bar{\Delta} = (\bar{\Delta}_1\bar{\Delta}_2\bar{\Delta}_3)^{1/3}$. Other integral kernels include the Gaussian, used by e.g. John (2006):

$$G(\mathbf{x}) = \left(\frac{6}{\pi\bar{\Delta}^2} \right)^{1/2} \exp\left(-\frac{6r^2}{\bar{\Delta}^2} \right), \quad r = |\mathbf{x}|, \quad (5.48)$$

and the sharp spectral (a top-hat in wavenumber space), used by e.g. Germano et al. (1991).

5.3.1.2 Effect in Wavenumber Space

Fourier transformation of the filter function clearly demonstrates its effect on the wavenumber space, or energy spectrum, of the solution. If the solution \mathbf{u} has the discrete Fourier transform

$$\hat{\mathbf{u}}(\kappa) = \mathcal{F}(\mathbf{u}(\mathbf{x})) = \sum_{n=0}^{N-1} \mathbf{u}(\mathbf{x}_n) e^{-2\pi n\kappa i/N}, \quad (5.49)$$

where κ is the wavenumber, n is the sampling point, $i = \sqrt{-1}$, N is the number of sampling points and $\mathbf{u}(\mathbf{x}_n)$ are the sampled values of \mathbf{u} at equally spaced positions \mathbf{x}_n , then the transfer function $H(\kappa)$ of \mathbf{u} is

$$H(\kappa) = \int_{-\infty}^{\infty} G(\mathbf{x}) e^{-i\kappa\mathbf{x}} d\mathbf{x} = 2\pi\mathcal{F}(G(\mathbf{x})), \quad (5.50)$$

and the filtered solution $\bar{\mathbf{u}}(\mathbf{x}, t)$ has the Fourier transform

$$\hat{\bar{\mathbf{u}}}(\mathbf{x}, t) = \mathcal{F}(\bar{\mathbf{u}}(\mathbf{x}, t)) = H(\kappa)\hat{\mathbf{u}}(\kappa). \quad (5.51)$$

Transfer functions are also functions of filter width $\bar{\Delta}$. In general, a filter transfer function has the value one where $\kappa = 0$ and diminishes to zero as κ increases. Thus small scales (those with large wavenumber) are filtered out whereas large scales are preserved. Mid-range scales are damped to varying degrees depending on the shape of the transfer function.

5.3.1.3 Moments

Kernels are analogous to probability density functions (PDFs) and the methods used to analyse filters are the same as for statistical distributions. Only *symmetric* filter kernels are considered for LES, i.e. those for which $G(-x) = G(x)$ (zero first moment M_1 , analogous to the mean of a PDF), because these are purely dissipative and non-dispersive, i.e. the spectrum amplitude is modified, not its frequency. Furthermore, only filters with strictly positive transfer functions are considered in this thesis because these prevent *spurious* energy backscatter (as opposed to intentional backscatter by letting $c_S \leq 0$) from the SFS to resolved scales (cf. §5.1.6); Figure 5.2 demonstrates that the top-hat does not belong to this class.

The second moment M_2 of a filter kernel is the first nonzero moment of the filters, analogous to the variance (square of standard deviation) of a PDF:

$$M_2 = - \left(\frac{\partial^2 H(\kappa)}{\partial \kappa^2} \right)_{\kappa=0}. \quad (5.52)$$

The top-hat, Gaussian and inverse Helmholtz filters have $M_2 = \mathcal{O}(\Delta^2)$. This result is important because it partially determines the convergence rate of the commutation error with respect to mesh size, as will be seen shortly.

5.3.1.4 Taylor Expansion

The Taylor series expansion of a filtered field about the centroid of the filter kernel x_0 is given by (in 1D):

$$\bar{u}(x) = u(x) + \sum_{n=1}^{\infty} \frac{M_n}{n!} \frac{\partial^n u(x)}{\partial \xi^n}, \quad (5.53)$$

where M_n are the filter moments and $\xi = x - x_0$ is a local coordinate. By this method Love (1980) showed that the top-hat and Gaussian filters have equivalent form up to second order in filter width. The Taylor expansion of a field filtered by the top hat filter is written (Sagaut, 1999) (up to fourth order terms in $\bar{\Delta}$):

$$\bar{u}(x) = u(x) + \frac{\bar{\Delta}^2}{24} \frac{\partial^2 u(x)}{\partial \xi^2} + \frac{\bar{\Delta}^4}{1920} \frac{\partial^4 u(x)}{\partial \xi^4} + \mathcal{O}(\bar{\Delta}^6), \quad (5.54)$$

and by the Gaussian filter (Sagaut, 1999):

$$\bar{u}(x) = u(x) + \frac{\bar{\Delta}^2}{24} \frac{\partial^2 u(x)}{\partial \xi^2} + \frac{\bar{\Delta}^4}{1152} \frac{\partial^4 u(x)}{\partial \xi^4} + \mathcal{O}(\bar{\Delta}^6). \quad (5.55)$$

5.3.1.5 Differential Filters

While the top-hat and Gaussian filters have been introduced starting from the kernel shape, the expansion (5.53) is a general form of filter in its own right, known as a differential filter. Differential filters were introduced in the context of LES in Germano (1986a). They solve an elliptic PDE for the filtered field $\bar{\mathbf{u}}$ supplemented with boundary conditions and are an extension of the Gaussian filter to bounded domains (Manica, 2006). The generic second-order differential filter equation for a vector is

$$\mathbf{u}(\mathbf{x}) = \bar{\mathbf{u}}(\mathbf{x}) + \varepsilon_i(\mathbf{x})\nabla\bar{\mathbf{u}}(\mathbf{x}) - \varepsilon_{ij}(\mathbf{x})\nabla^2\bar{\mathbf{u}}(\mathbf{x}), \quad (5.56)$$

where $\varepsilon_i, \varepsilon_{ij}$ are coefficients. Useful properties of differential filters include ease of implementation in integral form in a finite element code (see §5.3.5.2).

5.3.2 The Inverse Helmholtz Filter

The inverse Helmholtz filter is a differential operator introduced by Germano (1986a), and since used by many others with a variety of LES models, e.g. Geurts and Holm (2004), Vreman et al. (1996). The operator is defined via the Helmholtz equation

$$\mathbf{u} = \bar{\mathbf{u}} - \varepsilon\nabla^2\bar{\mathbf{u}}, \quad (5.57)$$

where the choice of $\varepsilon = \bar{\Delta}^2/24$ gives the inverse Helmholtz filter the same second moment as the Gaussian and top-hat filters (Geurts and Holm, 2004) (cf. Table 5.1). It is in fact an approximation of the Gaussian on bounded domains (Dunca and John, 2003) and was recommended by Mullen and Fischer (1999) for use in complex 3D domains.

The filtered field $\bar{\mathbf{u}}$ is found by inverting (5.57) and applying Dirichlet boundary conditions \mathbf{u}_D :

$$\bar{\mathbf{u}} = \left(\mathbf{I} - \frac{\bar{\Delta}^2}{24}\nabla^2 \right)^{-1} \mathbf{u} \quad \text{in } \Omega, \quad (5.58a)$$

$$\bar{\mathbf{u}} = \mathbf{u}_D \quad \text{on } \Gamma, \quad (5.58b)$$

where \mathbf{I} is the identity matrix, Ω is the solution domain, Γ is the domain boundary and $\bar{\Delta}$ is the scalar filter width given by the cube root of element volume (5.7). A tensor filter width $\bar{\Delta} = \bar{\Delta}_{ij}$ can be incorporated in the coefficient ε as suggested by Germano (1986a), thus arriving at a *tensorial inverse Helmholtz operator*:

$$\bar{\mathbf{u}} = \left(\mathbf{I} - \frac{1}{24}\bar{\Delta}^2 \cdot \nabla^2 \right)^{-1} \mathbf{u}. \quad (5.59)$$

Note that inclusion of the tensorial filter width in the inverse Helmholtz filter does not violate frame invariance, unlike the tensorial eddy viscosity LES model, (5.23).

A scalar filter width is used in the analysis that follows for simplicity. The inverse Helmholtz-filtered field has the Taylor series expansion

$$\bar{u}(x) = u(x) + \frac{\bar{\Delta}^2}{24} \frac{\partial^2 u(x)}{\partial \xi^2} + \frac{\bar{\Delta}^4}{576} \frac{\partial^4 u(x)}{\partial \xi^4} + \mathcal{O}(\bar{\Delta}^6), \quad (5.60)$$

which is equivalent to the top-hat and Gaussian kernels up to second order. This supports the finding of [Tejada-Martinez and Jansen \(2003\)](#) that the type of test filter used in the dynamic method did not significantly influence results because generally numerical discretisations are only second-order accurate. Despite its similarity to other filters in discrete form, the inverse Helmholtz filter has the distinct advantage that boundary conditions can be applied and that it is easily solved by the finite element method, as will be shown in §5.3.5.2. Only if fourth order terms are included does the difference between the filters appear; the inverse Helmholtz filter has the largest fourth-order contribution which may explain why it has the widest support in Figure 5.1.

5.3.3 Filter Comparisons

Table 5.1 lists filters and their kernels $G(r)$ in terms of radius r and transfer functions $H(\kappa)$, and we can see that the Gaussian and inverse Helmholtz are similar. The top-hat is written in terms of the Heaviside step function $H(r)$ (distinct from the transfer function $H(\kappa)$), an alternative form to (5.47).

Name	Filter kernel	Transfer function	2 nd moment
General	$G(r)$	$\int_{-\infty}^{\infty} e^{-i\kappa r} G(r) dr$	$-\left(\frac{\partial^2 H(\kappa)}{\partial \kappa^2}\right)_{\kappa=0}$
Top-hat	$\frac{1}{\bar{\Delta}} H(\frac{1}{2}\bar{\Delta} - r)$	$\prod_{i=1}^3 \frac{\sin(\frac{1}{2}\kappa_i \bar{\Delta}_i)}{\frac{1}{2}\kappa_i \bar{\Delta}_i}$	$\frac{\bar{\Delta}^2}{12}$
Gaussian	$(\frac{6}{\pi\bar{\Delta}^2})^{1/2} \exp(-\frac{6r^2}{\bar{\Delta}^2})$	$\exp(-\kappa^2 \bar{\Delta}^2 / 24)$	$\frac{\bar{\Delta}^2}{12}$
Inverse Helmholtz	$(\frac{1}{4\pi\bar{\Delta}^2}) \frac{1}{ r } \exp(-\frac{ r }{\bar{\Delta}})^*$	$1/(1 + \kappa^2 \bar{\Delta}^2 / 24)$	$\frac{\bar{\Delta}^2}{12}$

Table 5.1: Top-hat, Gaussian and inverse Helmholtz filter kernels, transfer functions and second moments. H in the top-hat kernel is the Heaviside step function.

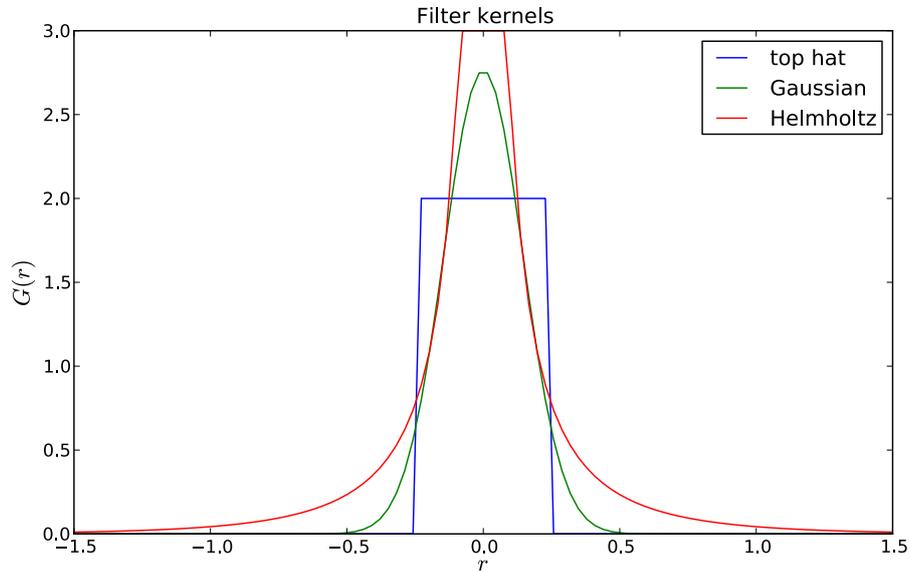


Figure 5.1: Top-hat, Gaussian and inverse Helmholtz filter kernels $G(r)$ in terms of radius r , for width $\overline{\Delta} = 0.5$. The inverse Helmholtz kernel has a value of ∞ at $r = 0$. The shape of the kernel determines the degree to which information is filtered in relation to its distance r from the origin.

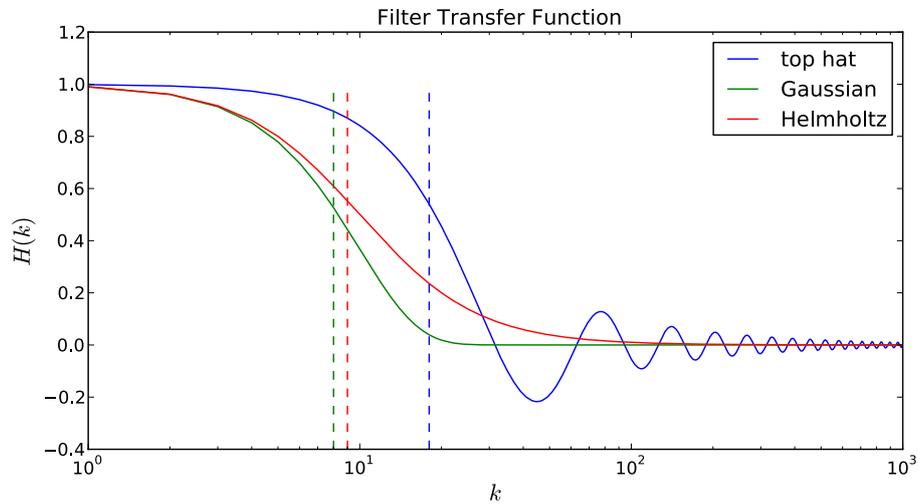


Figure 5.2: Plot of filter transfer functions with width $\overline{\Delta} = 0.1$. The top-hat is distinctly non-local in spectral space, with an oscillatory tail which causes spurious damping ($H(\kappa) > 0$) or enhancement ($H(\kappa) < 0$) of high wavenumbers. The Gaussian and Helmholtz are quasi-local, smooth and non-negative, completely damping out high wavenumbers. Dotted lines show average wavenumbers κ^* where $H(\kappa) = 0.5$, sometimes used to calculate filter width, showing that different definitions of width give differing widths.

Figure 5.1 shows the top-hat, Gaussian and inverse Helmholtz filter kernels with width $\overline{\Delta} = 0.5$ in 1D in terms of radius $r = x - x_0$, demonstrating that the top-hat is local in physical space while the Gaussian and Helmholtz are quasi-local in physical space (i.e. they have long tails). Figure 5.2 shows the three transfer functions against wavenumber κ in 1D. The plot shows that the top-hat is distinctly non-local in spectral space while the Gaussian and Helmholtz are quasi-local, and that different filters of the same width can have quite different effects on the energy spectrum. The top-hat is poor at damping high-wavenumber modes and is shifted to the right of the others, indicating that it is dissipative at higher wavenumbers, i.e. smaller scales than the others shown, while the Gaussian is the least dissipative of the three at high wavenumbers.

The dotted lines in Figure 5.2 represent the average wavenumber κ^* , which is the value of κ at which $H(\kappa) = 0.5$. An alternative definition of filter width, representing the effect of filtering on the spectrum, is given by $\overline{\Delta}^* = \pi/\kappa^*$. In 3D this is $\overline{\Delta}^* = \pi/\kappa_r^*$, where $\kappa_r = |\kappa|$ is the average radial wavenumber (Tejada-Martinez and Jansen, 2004). As explained in §5.2.2.5, the results of the dynamic method were found to be invariant to filter shape if the width was computed by this method (Tejada-Martinez and Jansen, 2003). However, in physical space $\overline{\Delta}^*$ is not a trivial quantity to compute, particularly on unstructured meshes where calculating a discrete Fourier transform of the solution is not possible; it is better suited to spectral discretisation and solution methods, e.g. Winckelmans et al. (1998). For our purposes, a more practical definition is $\overline{\Delta}$ based on element size Δ .

5.3.4 Commutation Errors

A crucial but oft-neglected consequence of filtering the Navier-Stokes equations is the appearance of an error. Adopting the notation $\partial_j u = \partial u / \partial x_j$ for clarity, the incompressible Navier-Stokes equations (3.1) in the absence of body forces are written

$$\partial_t u_i + \partial_j (u_i u_j) + \partial_i p - \nu \partial_{jj} u_i = 0, \quad (5.61a)$$

$$\partial_j u_j = 0. \quad (5.61b)$$

On applying a generic explicit filtering operator \mathcal{G} with non-uniform filter width $\overline{\Delta}(\mathbf{x}, t)$ to (5.61), we obtain the filtered Navier-Stokes equations (van der Bos and Geurts, 2005):

$$\begin{aligned} \partial_t \bar{u}_i + \partial_j (\bar{u}_i \bar{u}_j) + \partial_i \bar{p} - \nu \partial_{jj} \bar{u}_i = \\ -\partial_j \tau_{ij} - \mathcal{C}_t(\bar{u}_i) - \mathcal{C}_j(\bar{u}_i \bar{u}_j) - \mathcal{C}_i(p) + \nu \mathcal{C}_{jj}(\bar{u}_i), \end{aligned} \quad (5.62a)$$

$$\partial_j \bar{u}_j = -\mathcal{C}_j(\bar{u}_j). \quad (5.62b)$$

Extra terms, the *commutation errors* \mathcal{C} , have appeared compared to (3.27); in the earlier expression they were left out for simplicity. If the filter width is uniform, then these errors disappear. If so, the first term on the r.h.s. of (5.62a), the SFS force $\partial_j(\overline{u_i u_j} - \overline{u_i} \overline{u_j})$, is the only term remaining (van der Bos and Geurts, 2005). The commutation errors contribute to the transfer of energy between the SFS and resolved scales in a particular way which depends on the inhomogeneity and anisotropy of the filter width (Geurts and Leonard, 2000).

Commutation errors arise because the non-uniform filter operator \mathcal{G} applied to the gradient of a function ∇f does not give the same answer as the reverse operation, the gradient of a filtered field, i.e. the operations do not commute:

$$\mathcal{G}(\nabla f) \neq \nabla(\mathcal{G}(f)). \quad (5.63)$$

The commutation error in the gradient of a filtered function f is then

$$\mathcal{C}(f) = \mathcal{G}(\nabla f) - \nabla(\mathcal{G}(f)). \quad (5.64)$$

A commutation error also appears in the continuity equation if explicitly filtering, (5.62b), hence zero divergence is not guaranteed in this case (Fureby and Tabor, 1997). This error is included in the modified pressure (3.14) when using an eddy-viscosity model and is therefore impossible to quantify. Manica (2006) suggested using the Stokes equation – (5.58) with an added pressure gradient term – as the basis for a differential filter to preserve the incompressibility condition but it had a larger commutation error. Langford and Moser (2001) described strategies for optimising the continuity equation to minimise the error, but in general it is ignored.

In a bounded domain the commutation error has two components: an interior term and a boundary term (van der Bos and Geurts, 2005). The boundary error, which arises because the support of the filter cannot extend beyond the domain boundary, is zero, if and only if the boundaries exert zero force on the fluid – in other words, in almost no problems of practical interest (Dunca and John, 2003). Otherwise it is non-zero within a distance δD from solid boundaries determined by the support of the filter (the region of integration in (3.17)). It may be possible to apply different boundary conditions to the filtered and unfiltered fields to cancel out the error (Fureby and Tabor, 1997).

For any symmetric filter with a well-defined nonzero second moment in physical space, including the top hat, Gaussian and inverse Helmholtz, the interior error term can be expressed as a Taylor series (in 1D for simplicity) in terms of the moments M_n of the filter kernel, the gradient of the filter size

$\frac{d\bar{\Delta}}{dx}$ and the derivatives of the filtered function $f(x)$:

$$\mathcal{C}_x(f(x)) = -\frac{d\bar{\Delta}}{dx} \sum_{n=1}^{\infty} \frac{1}{(n-1)!} \bar{\Delta}^{n-1} M_n \frac{d^n f}{dx^n}, \quad (5.65)$$

$$M_n = \int_{-\infty}^{\infty} s^n G(s) ds, \quad (5.66)$$

where $s = (y - x)/\bar{\Delta}(x)$ is a local coordinate (van der Bos and Geurts, 2005). Kollmann et al. (2002) confirmed this result in an analysis of the commutation error for the top-hat filter in a finite element discretisation. The dominant commutation error term in 1D is proportional to $\bar{\Delta} \frac{d\bar{\Delta}}{dx}$ or $\bar{\Delta} \cdot \nabla \bar{\Delta}$ in higher dimensions, which is why the error disappears when filter width is constant (Germano, 1986b). Note that the second moment in terms of x is proportional to $\bar{\Delta}^2$ but the coordinate transformation cancels it out.

One method of controlling the error is to increase the order of accuracy of the filter kernel by controlling the moments M_n . A class of filters with zero moments up to order $n - 1$ was developed by Vasilyev et al. (1998) which commute up to order n for varying filter width, although they should be used with higher-order numerical methods to take advantage of the increased accuracy (van der Bos and Geurts, 2005). For example, Gullbrand (2003b) used a fourth-order numerical scheme with a fourth-order commuting filter. The extension of Vasilyev's class of filters to unstructured meshes was considered by Vasilyev et al. (2002).

It is also possible to model the commutation errors in (5.62a) directly and include the models in the filtered equations. For example, the last term on the r.h.s. of (5.62a), the second-derivative commutation error, is equivalent to the Germano identity (5.31), suggesting a possible modelling strategy (van der Bos and Geurts, 2005). Guermond et al. (2004) proposed an LES model whose accuracy does not rely on the commutation error of the filter. It was also noted in Germano (1986b) that the commutation error could be expressed in terms of $\bar{\mathbf{u}}$, opening up error modelling possibilities, and Ioveno et al. (2004) found that explicit modelling of commutation errors made a noticeable improvement to accuracy in channel flow simulations.

The error expression (5.65) demonstrates the clear need for some control over the gradient of $\bar{\Delta}$ in an unstructured mesh (van der Bos and Geurts, 2005). Careful mesh design, particularly when using mesh adaptivity, is crucial in this regard. Provided that $\nabla \bar{\Delta} = \mathcal{O}(\bar{\Delta})$, which is true if the filter width varies smoothly enough, the error convergence rate is (Kollmann et al., 2002)

$$\mathcal{C}_x(f(x)) = \mathcal{O}(\bar{\Delta}^2). \quad (5.67)$$

For rapidly varying filter width, $\nabla \bar{\Delta} = \mathcal{O}(\bar{\Delta}^\gamma)$, $\gamma < 1$, the order of convergence is reduced to

$$\mathcal{C}_x(f(x)) = \mathcal{O}(\bar{\Delta}^{n+\gamma-1}), \quad (5.68)$$

where n is the order of the first nonzero moment (Vasilyev et al., 2002). A generic anisotropic unstructured mesh has independently varying length-scales in all directions. For second-order symmetric filters the commutation error (5.65) in direction x on an anisotropic 3D mesh is (Vasilyev et al., 2002):

$$\mathcal{C}_x(f(\mathbf{x})) = \mathcal{O} \left[\frac{\partial}{\partial x} \left(\bar{\Delta}_x^i(\mathbf{x}) \bar{\Delta}_y^j(\mathbf{x}) \bar{\Delta}_z^k(\mathbf{x}) \right) \bar{\Delta}_x^i(\mathbf{x}) \bar{\Delta}_y^j(\mathbf{x}) \bar{\Delta}_z^k(\mathbf{x}) \right], \quad i+j+k = 1. \quad (5.69)$$

5.3.5 Implementation of the Tensorial Inverse Helmholtz Filter

5.3.5.1 Usage in the Tensorial Dynamic Method

In the tensorial dynamic method the inverse Helmholtz filter is used as an explicit test filter to calculate $\tilde{\mathbf{u}}$:

$$\tilde{\mathbf{u}} = \left(\mathbf{I} - \frac{\tilde{\Delta}^2}{24} \cdot \nabla^2 \right)^{-1} \bar{\mathbf{u}}, \quad (5.70a)$$

$$\tilde{\Delta} = \tilde{\Delta}_{ij} = \beta \bar{\Delta}_{ij}. \quad (5.70b)$$

Note that setting $\beta = 0$ implies $\tilde{\mathbf{u}} = \bar{\mathbf{u}}$, while $\beta = 1$ causes some filtering. The first filtered velocity can also be obtained from the discrete velocity $\bar{\mathbf{u}}$ by an explicit filtering operation with tensor width $\bar{\Delta}$.

In the expressions for the dynamic Smagorinsky coefficient (5.43b) and Leonard tensor (5.42) a combined filter width $\tilde{\Delta}$ appears. This width (scalar or tensor) is the convolution of mesh and test filter widths $\tilde{\Delta}$, given by the triangular relationship (5.26), which is derived from the Fourier transform of the inverse Helmholtz filter:

$$\begin{aligned} H(k, \tilde{\Delta}) &= H(k, \bar{\Delta}) H(k, \tilde{\Delta}) \\ &= \frac{1}{1 + k^2/24\tilde{\Delta}^2} \times \frac{1}{1 + k^2/24\bar{\Delta}^2} \\ &= \frac{1}{1 + k^2/24(\tilde{\Delta}^2 + \bar{\Delta}^2) + (k^2/24)^2(\tilde{\Delta}^2\bar{\Delta}^2)} \\ \Rightarrow \tilde{\Delta}^2 &= \tilde{\Delta}^2 + \bar{\Delta}^2 + (k^2/24)\tilde{\Delta}^2\bar{\Delta}^2. \end{aligned} \quad (5.71)$$

The fourth-order term, third on the r.h.s., is neglected in the triangular expansion but it may be non-negligible for large wavenumbers k . Since a second-order-accurate discretisation scheme is employed here it is neglected.

5.3.5.2 Discretisation

Filtering has thus far been limited to continuous functions. In the CFD setting a discrete filter is applied to a discrete field which limits the extent of the filter's support. For example, in the finite volume method the integration region is limited to a patch of control volumes surrounding a node, and in the finite element method support may be limited to a single element. Higher-order discretisations with wider stencils might widen the support but they are not considered here.

In this thesis, the inverse Helmholtz filter operator is discretised by the finite element method, i.e. as an integration over each element followed by a global solve of the discretised equation. Employing the Galerkin finite element method, the inverse Helmholtz equation (5.58) applied to a field $\mathbf{u}^h \in \mathcal{U}^h$, weighted with a function $\mathbf{w}^h \in \mathcal{W}^h$ and integrated over a bounded domain Ω is written

Given $\mathbf{u}^h \in \mathcal{U}^h$, find a filtered field $\bar{\mathbf{u}}^h \in \mathcal{U}^h$ such that

$$\int_{\Omega} \mathbf{u}^h \mathbf{w}^h d\Omega = \int_{\Omega} \bar{\mathbf{u}}^h \mathbf{w}^h d\Omega - \int_{\Omega} \frac{\bar{\Delta}^2}{24} \nabla^2 \bar{\mathbf{u}}^h \mathbf{w}^h d\Omega, \quad \forall \mathbf{w}^h \in \mathcal{W}^h. \quad (5.72)$$

Integrating the second term on the right by parts:

$$\int_{\Omega} \mathbf{u}^h \mathbf{w}^h d\Omega = \int_{\Omega} \bar{\mathbf{u}}^h \mathbf{w}^h d\Omega + \int_{\Omega} \nabla \bar{\mathbf{u}}^h \cdot \frac{\bar{\Delta}^2}{24} \cdot \nabla \mathbf{w}^h d\Omega - \int_{\Gamma} \frac{\bar{\Delta}^2}{24} \cdot \mathbf{n} \cdot \nabla \bar{\mathbf{u}}^h \mathbf{w}^h d\Gamma, \quad (5.73)$$

where an integral over the domain boundary Γ has appeared allowing the imposition of a weak Neumann boundary condition. Note that the minus sign in front of the second derivative term has become a plus after integrating by parts. In inner product notation (5.72) is written

$$(\mathbf{u}^h, \mathbf{w}^h) = (\bar{\mathbf{u}}^h, \mathbf{w}^h) + \frac{\bar{\Delta}^2}{24} (\nabla \bar{\mathbf{u}}^h, \nabla \mathbf{w}^h), \quad (5.74)$$

where a homogeneous Neumann condition is imposed implicitly by not including the boundary term in the CFD code. The equation is supplemented with strong Dirichlet boundary conditions $\bar{\mathbf{u}} = \bar{\mathbf{u}}_D$ following the method

in §3.5.2.5. Appropriate choices of Dirichlet conditions are discussed below. Choosing identical test and trial spaces \mathcal{U}^h and \mathcal{W}^h and linear shape functions N_i we write the matrix equation over an element Ω^e :

$$\begin{aligned}\bar{\mathbf{u}}^e &= (\mathbf{M}_{\text{Helm}}^e)^{-1} \mathbf{u}^e, \\ \mathbf{M}_{\text{Helm}}^e &= \int_{\Omega^e} N_i N_j d\Omega + \int_{\Omega^e} \nabla N_i \cdot \frac{\bar{\Delta}_{ij}^2}{24} \cdot \nabla N_j d\Omega.\end{aligned}\quad (5.75)$$

Bearing in mind the above results for convergence of the commutation error due to non-uniform filtering, we must show that the results still hold for the tensor filter operator. A necessary condition for symmetry of the filter operator ($M_1 = 0$) is that $\mathbf{M}_{\text{Helm}}^e$ is symmetric. Noting that the tensor filter width $\bar{\Delta}_{ij}$ is equated to the inverse metric tensor \mathcal{M}^{-1} (see §5.2.1), and that the metric tensor is symmetric positive definite for any size and shape of element (Farrell, 2009), then its inverse is symmetric and so is $\mathbf{M}_{\text{Helm}}^e$ and its inverse. Pre- and post-multiplying by the shape function gradients preserves the symmetry since the shape functions N_i and N_j are identical. The first term in (5.75) is the mass matrix which is symmetric. Setting the Neumann boundary term to zero preserves the symmetry of $\mathbf{M}_{\text{Helm}}^e$. The sum of symmetric terms is also symmetric, so $\mathbf{M}_{\text{Helm}}^e$ is symmetric. The same result can be derived for the scalar filter width.

In the finite element setting the inverse Helmholtz-filtered field has been proven by Dunca and John (2004) to converge to the source field at second order in the \mathcal{L}^2 norm, provided that appropriate boundary conditions are applied:

For every $\mathbf{u} \in \mathcal{H}_0^1(\Omega) \cap \mathcal{H}^2(\Omega)$ and $\bar{\Delta} > 0$ holds

$$\|\bar{\mathbf{u}} - \mathbf{u}\|_{\mathcal{L}^2} \leq \bar{\Delta}^2 \|\nabla^2 \mathbf{u}\|_{\mathcal{L}^2}.\quad (5.76)$$

This result will be tested for the tensorial inverse Helmholtz filter in §5.3.7. The determination of appropriate boundary conditions follows.

Discretising the equation in variational form, such as by finite elements, improves the convergence rate of the commutation error on non-uniform meshes. Dunca and John (2003) proved that the commutation error induced by inverse Helmholtz filtering in a bounded domain in strong form, with constant $\bar{\Delta}$, converges at

$$\mathcal{C}_x = \mathcal{O}(\bar{\Delta}^{1/2}),\quad (5.77)$$

but in variational form it converges at higher order:

$$\mathcal{C}_x = \mathcal{O}(\bar{\Delta}^{1-\gamma}),\quad (5.78)$$

for arbitrarily small γ , i.e. close to order one. This result, and the effect of boundary conditions on the convergence rate, will also be tested in §5.3.7.

5.3.5.3 Boundary Conditions

Boundary conditions present both an opportunity and a problem with this filter. As seen above, the commutation error converges at second order, provided that appropriate boundary conditions are applied. However, no such conditions are known and therefore imposition of Dirichlet conditions will most likely incur an error on the domain boundaries (Dunca and John, 2004). Different conditions for source field and filtered field were suggested by Iovenio and Tordella (2003) on the basis that we expect the fields to differ on the boundary.

Bounded domains incur an extra commutation error because the filter width changes abruptly over the boundary. A method of reducing this error is to gradually reduce the filter width towards the boundary, but then extra resolution is required and the interior error term due to non-uniform filter width is emphasised. In this case we are solving a wall-resolving LES, and homogeneous Dirichlet conditions $\bar{\mathbf{u}} = 0$ are an appropriate choice (Dunca and John, 2003).

In the case of abruptly changing filter width at the boundary (little or no gradation), it is unclear what the best choice of Dirichlet conditions on $\bar{\mathbf{u}}$ is. There may exist appropriate conditions which do not incur an extra error, but they are not known (Dunca and John, 2003). If weakly enforced Dirichlet boundary conditions could be imposed, these might increase the order of convergence, but a method of doing so is unclear⁶. In this work the strong Dirichlet condition $\bar{\mathbf{u}} = \mathbf{u}$ and a weak homogeneous Neumann condition are applied on all boundaries. The expected *minimum* order of convergence of the commutation error is given by (5.77) and will be verified numerically (see §5.3.7).

5.3.6 Discrete Filters

Returning to the introduction to filtering, §5.1, a third way of deriving filters is from the discretisation method itself. Jansen (1999) described several ways to do so using basis functions in the finite element method. Pope (2001) developed similar methods using orthogonal projections as filters. Tejada-Martinez and Jansen (2003) developed several different discrete filters on

⁶Weak Dirichlet conditions can be enforced on the advection-diffusion and momentum equations by integrating the advection term by parts and subtracting from the r.h.s., but elliptic equations do not have any such term.

tetrahedral and hexahedral element topologies and analysed the effect on the optimal value of filter width ratio α in the dynamic model. Test filtering in the dynamic method was achieved by [Petry and Awruch \(2006\)](#) by construction of a ‘super-element’ around each node in a hexahedral finite element mesh. Then the test-filtered quantities were evaluated element-wise by interpolating from the super-element, equivalent to constructing a coarser mesh from the connectivity of the parent mesh and projecting discrete fields on the coarse mesh back to the original.

A simple finite-element-based filter can be obtained from a Galerkin projection incorporating the lumped mass matrix:

$$\bar{\mathbf{u}} = \mathbf{M}_L^{-1} \mathbf{M} \mathbf{u}, \quad (5.79)$$

where \mathbf{M} is the mass matrix (3.60a) and \mathbf{M}_L is the lumped mass matrix: $M_{L,ii} = \sum_j M_{ij}$, $M_{L,i \neq k} = 0$. Lumping the mass matrix is an averaging operation over a local patch which behaves like a filter in that it reduces the amount of information in $\bar{\mathbf{u}}$ compared to \mathbf{u} , analogous to the top-hat filter.

[Jansen \(1999\)](#) implemented the dynamic method in a parallelised FE code using a discrete FE filter based on the inverse lumped mass matrix. The additional cost of the dynamic method was less than one-fifth of a nonlinear iteration, in part because of the triviality of inverting the mass matrix. However, it has the disadvantage when used in the dynamic method of not being able to choose the filter width. Moreover, an expression for the filter width is not readily obtained. The lumped-mass filter is compared to the inverse Helmholtz filter in the verification tests which follow.

5.3.7 Verification of Filters

The inverse Helmholtz filter in scalar and tensor form and the lumped-mass filter have been implemented in the CFD code Fluidity. Numerical properties are verified here by obtaining convergence of the difference between the filtered and unfiltered fields and convergence of the commutation error using the method of manufactured solutions (MMS) of [Roache \(2002\)](#). Some simple 1D tests of the effect in spectral space of implicit and explicit filtering with different filter widths on a sinusoidal signal are also presented.

5.3.7.1 Test 1: Signal Smoothing in Spectral Space

Some basic properties of the inverse Helmholtz filter are established by looking at its effect in spectral space on a 1D signal comprising a superposition of sine waves. The signal was chosen for its distinct spectral characteristics

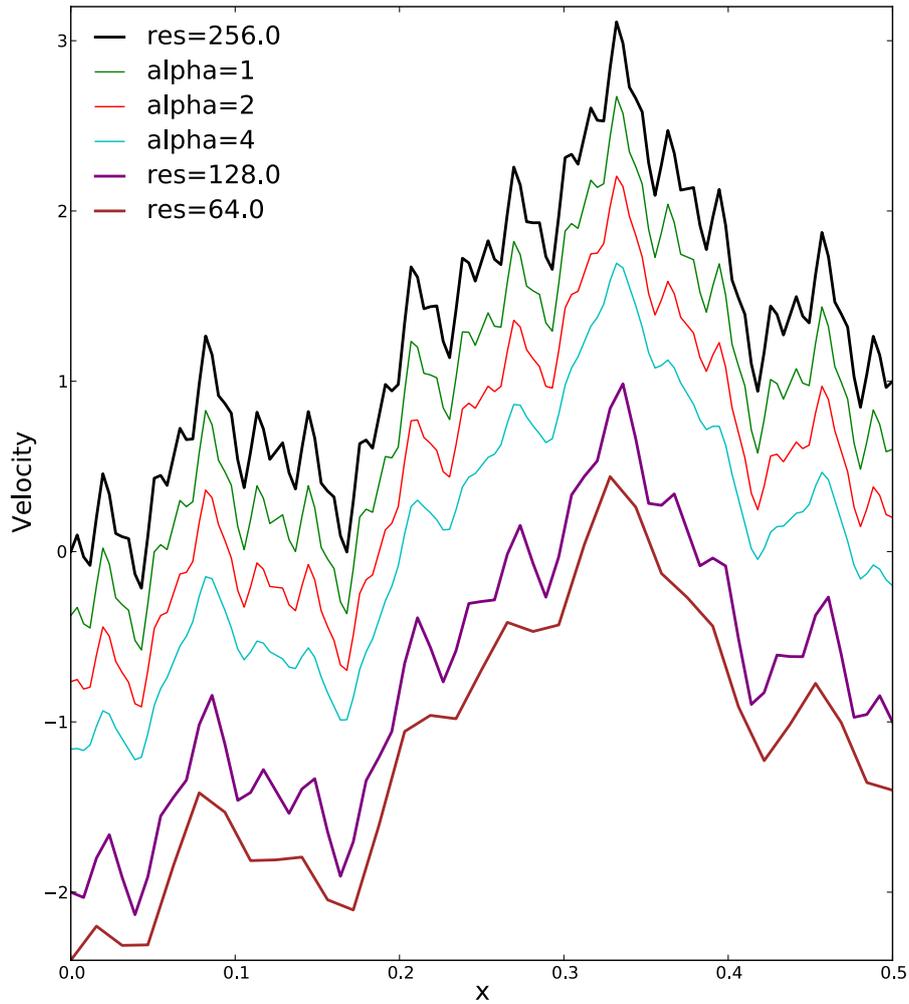


Figure 5.3: Effect of inverse Helmholtz filter on a signal in physical space. The filtered signals become smoother with increasing filter width. Coarse-mesh representations differ from filtered signals. From top to bottom: black: signal on 256-element mesh, green: filtered signal with $\alpha = 1$, red: filtered signal with $\alpha = 2$, blue: filtered signal with $\alpha = 4$, purple: signal on 128-element mesh, brown: signal on 64-element mesh. Origins have been shifted for each signal.

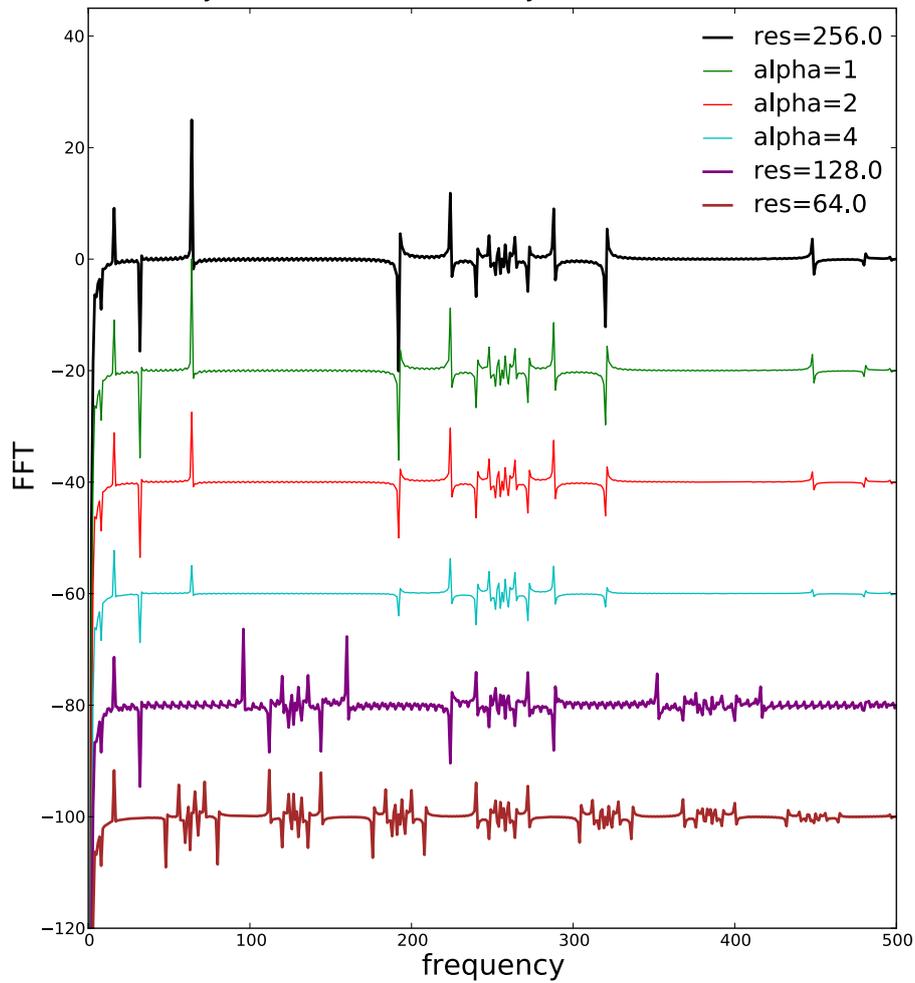


Figure 5.4: Effect of inverse Helmholtz filter on a signal in spectral space. Filtered signals display damping of wavemodes across the spectrum, while coarse-mesh representations show entirely different modes. From top to bottom: black: signal on 256-element mesh, green: filtered signal with $\alpha = 1$, red: filtered signal with $\alpha = 2$, blue: filtered signal with $\alpha = 4$, purple: signal on 128-element mesh, brown: signal on 64-element mesh. Origins have been shifted vertically for each signal.

and its visual similarity to a signal from a turbulent point probe:

$$u(x) = \sin(\pi x) + \sum_{n=1}^{10} \frac{(-1)^{n+1}}{n} \sin(2^n \pi x), \quad x = [0, 2]. \quad (5.80)$$

A minimum of 4096 uniform elements would be required to represent the smallest scales of this signal. The signal was prescribed on uniform 1D meshes with 256, 128 and 64 elements and piecewise linear continuous shape functions (P_1 CG). The 256-element mesh can represent up to $n = 6$; all scales smaller than this are in the subgrid scales. The coarser meshes were intended to represent the effect of projection of the signal of 256 elements onto meshes with two and four times larger elements, as if *implicitly filtering* the signal.

Because the test was in one dimension, it made no difference whether the filter was scalar or tensor. The filter width ratio α was set to one, two and four and the filtered signals compared to the unfiltered signal on meshes with 128 and 64 elements to see if explicit filtering differs from implicit filtering (discretisation on a coarser mesh). Figure 5.3 shows the effect of the inverse Helmholtz filter on the signal in physical space on the mesh of 256 elements. As α is increased, more fine scales are filtered out of the signal as expected. Compared to the coarser meshes, it appears that more fine scale information is retained in the explicitly filtered signal.

Figure 5.4 plots the fast Fourier transform of the filtered signals against frequency to demonstrate the effect of filtering in spectral space. Now it becomes apparent that the inverse Helmholtz filter has a broad-spectrum effect: all the peaks are reduced, with the degree of reduction depending on α , meaning that all scales of motion are damped. The frequencies are not shifted, so the filter is non-dispersive. The spectra of the unfiltered signals on coarser meshes are quite different: they have completely different dominant frequencies. Therefore, inverse Helmholtz filtering with a certain filter size is not the same as projection to coarse meshes with equivalent element size. Consequently, we would expect to see differences in the finest resolved scales between explicit and implicit LES.

The implication of the broad-band filtering observed in this test is that energy is drained from all resolved scales to the SFS in the dynamic LES model. This is not what was expected by looking at the filter's transfer function, Figure 5.2: in that plot the higher wavenumbers were damped much more than the lower. However, it is likely that the choice of P_1 CG has some influence over filter's performance, both in terms of the energy dissipation and spectral shape. Tests with different basis functions, e.g. P_2 CG or P_0 DG, would be revealing, since these discretisations would represent the finest scales more or less accurately.

5.3.7.2 Test 2: Pure Diffusion of a Scalar

A Gaussian bump scalar field $\varphi(x, y)$ shown in Figure 5.5a was prescribed on a $[0, 2]^2$ square with zero advection. The scalar and tensor inverse Helmholtz filters and lumped-mass filter were used to filter the field. The scalar field and its filtered counterpart were discretised with the continuous Galerkin finite element discretisation with linear basis functions. In this simplest of MMS tests there was no advection.

Results on five structured and five unstructured tetrahedral meshes were compared. Mesh resolution varied from 8 to 128 edges per side (approximate in the unstructured meshes) with edge lengths halved between successive meshes. Unstructured meshes were generated using the command

```
gmsh -2 -algo frontal -o MMS_A.msh src/MMS_A.geo.
```

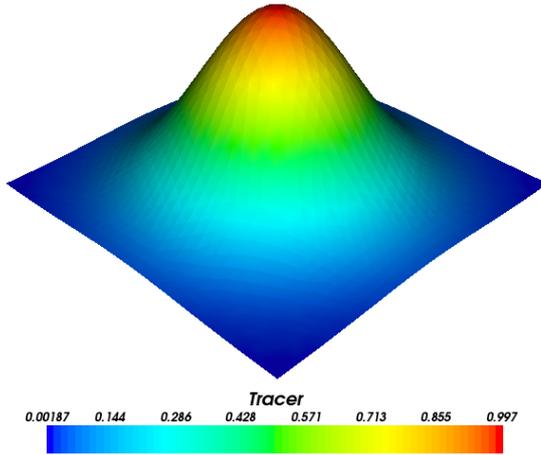
The `frontal` (advancing front) algorithm generated a smoother mesh than others such as Delaunay triangulation (the default for `gmsh`); it was found that Delaunay produced a ‘knot’ of elements which were smaller than the surrounding elements, causing a sudden jump in filter width and contaminating the commutation error on the finest meshes. These still appear in the unstructured meshes below but are less severe. Table 5.2 summarises the meshes.

Mesh	edges/side	elements	nodes
A structured	8	128	81
B structured	16	512	289
C structured	32	2048	1089
D structured	64	8192	4225
E structured	128	32768	16641
A unstructured	8	146	88
B unstructured	16	612	337
C unstructured	32	2446	1286
D unstructured	64	9686	4970
E unstructured	128	38660	19585

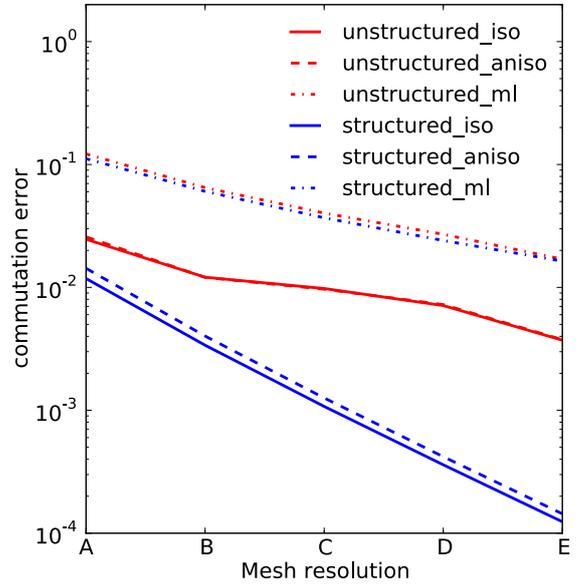
Table 5.2: MMS test two mesh characteristics.

The \mathcal{L}^2 norm of the difference between the filtered and original fields, $\|\varphi - \bar{\varphi}\|_{\mathcal{L}^2}$ was calculated and is henceforth referred to as the filter error norm. Second-order convergence of the filter error norm is expected on either mesh topology according to (5.76). A commutation error (a vector) inherent in the filtering operation was calculated by the difference between the filtered

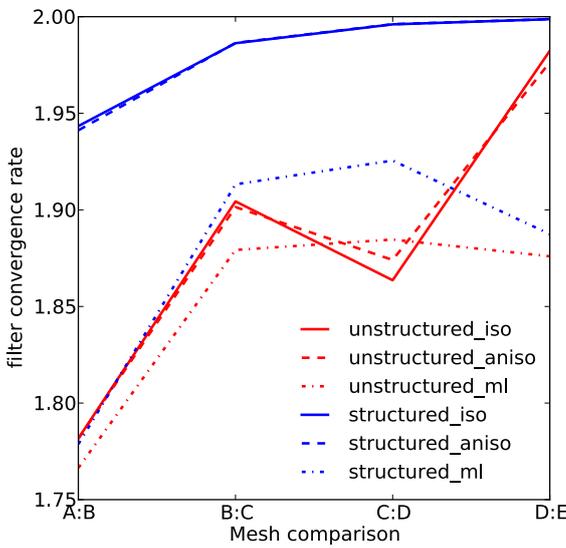
5.3 A Tensorial Differential Filter For LES



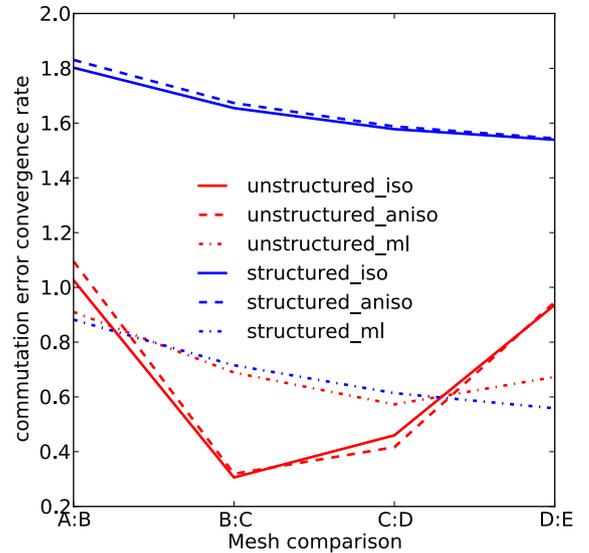
(a) Gaussian bump field φ on structured ‘C’ mesh



(b) Commutation error norm



(c) Filter error convergence rate



(d) Commutation error convergence rate

Figure 5.5: Tracer error convergence test results comparing different filters. Source field, commutation error magnitude, filter error convergence rate and commutation error convergence rate are shown. ‘iso’ = isotropic/scalar filter, ‘aniso’ = anisotropic/tensor filter, ‘ml’ = lumped-mass filter.

gradient of φ and the gradient of filtered φ :

$$\mathcal{C} = \overline{\nabla\varphi} - \nabla\bar{\varphi}. \quad (5.81)$$

The commutation error magnitude was measured in the \mathcal{L}^2 norm over the domain. According to (5.69) the commutation error norm should be zero on a uniform mesh but convergence is limited by boundary errors. The gradient of element size also limits convergence of the commutation error norm. Element size in the structured meshes is exactly constant across the domain (except at the boundaries) so commutation error norms are expected to converge at second order as the mesh is refined. Element size is approximately constant (except at the boundaries) in the unstructured meshes so commutation error norm convergence is expected to be slightly lower than second order.

Figure 5.5 shows the commutation error norm $\|\overline{\nabla(\varphi)} - \nabla(\bar{\varphi})\|_{\mathcal{L}^2}$ (5.5b), the convergence rate of the filter error norm $\|\varphi - \bar{\varphi}\|_{\mathcal{L}^2}$ (5.5c) and the convergence rate of the commutation error norm (5.5d) for both inverse Helmholtz filters and the lumped-mass filter on structured and unstructured tetrahedral meshes. On structured meshes the convergence rate of the filter error norm for both inverse Helmholtz filters tends to the expected second order. The commutation error norm convergence rate is slightly below second order, trending towards $\mathcal{O}(1.5)$, owing to the increasing influence of boundary errors on the finer meshes. Figure 5.5b shows that the scalar Helmholtz filter commutation error norm is smaller than its tensor counterpart on structured meshes. On unstructured meshes the convergence rate of the filter error norm for both inverse Helmholtz filters is lower but still tends towards second order, and the commutation error norm convergence rate for both inverse Helmholtz filters is much lower but still positive ($\approx \mathcal{O}(0.4 - 1.0)$). These results are at least partly down to differences in unstructured mesh generation at each level A – E as well as boundary errors. The commutation error convergence rate is approximately consistent with the bounds set by (5.77) and (5.78), i.e. less than one and greater than a half. The lumped-mass filter has a lower convergence rate in both errors and is far less sensitive to mesh topology (i.e. whether the mesh is structured or unstructured). The commutation error magnitude is also much larger than either inverse Helmholtz filter. These results may be due to the large error on the boundary (see Figures 5.7e and 5.7f).

Figure 5.6 displays the distribution of the filter error over the domain on structured and unstructured ‘B’ meshes. On the structured mesh there is some anisotropy in the error (dark blue patches framing the centre peak to left and right, circled) visible in the tensor-filtered field 5.6c aligned with the mesh anisotropy (directions of greatest and least element size), but not in the

5.3 A Tensorial Differential Filter For LES

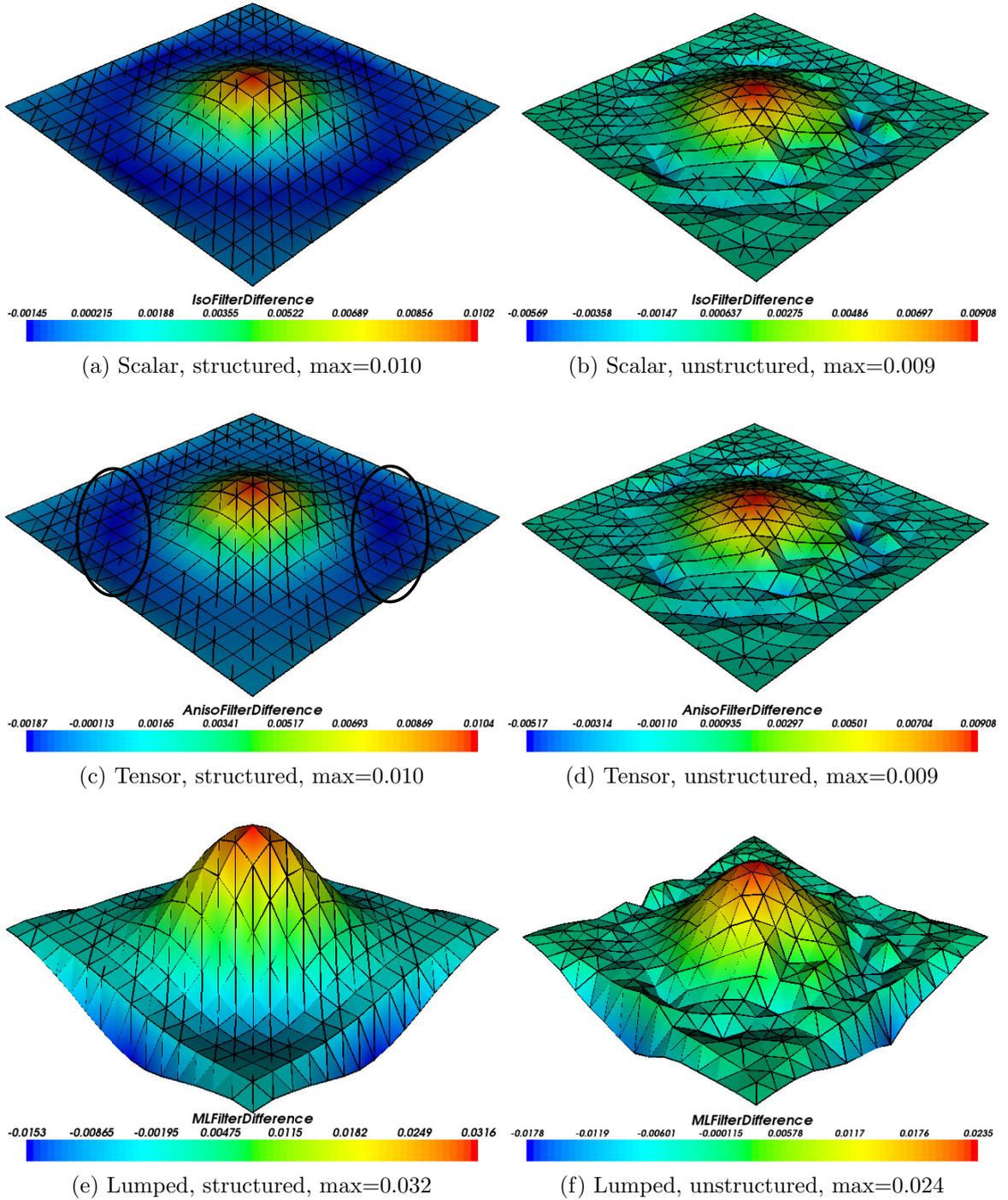


Figure 5.6: Difference between filtered and source tracer fields ($\varphi - \bar{\varphi}$) on structured and unstructured ‘B’ meshes comparing three filters. Maximum values are listed. Anisotropic effects circled in 5.6c.

5.3 A Tensorial Differential Filter For LES

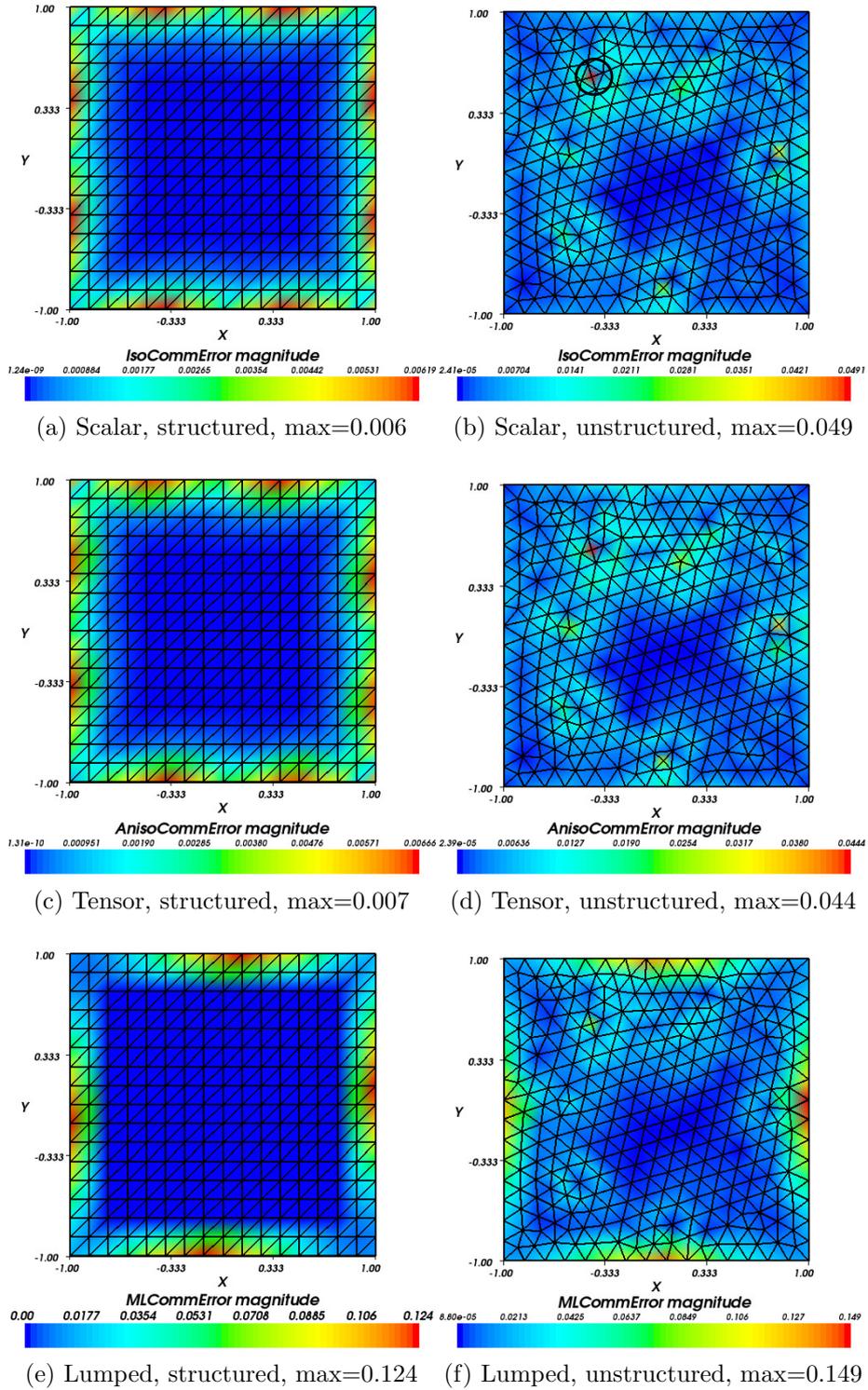


Figure 5.7: Tracer commutation error magnitude $|\overline{\nabla(\varphi)} - \nabla(\overline{\varphi})|$ on structured and unstructured ‘B’ meshes comparing three filters. Maximum values are listed. Mesh ‘knot’ circled in 5.7b. 173

scalar-filtered field 5.6a because it is not sensitive to mesh anisotropy. There is little difference between the filter error distributions returned by the scalar and tensor filters on the unstructured mesh (Figures 5.6b and 5.6d). Both have a central maximum where the field φ is changing most rapidly, with the error reducing roughly to zero towards the edges, though less smoothly than on the structured mesh. Both inverse Helmholtz filters return zero error on the boundary because of the application of Dirichlet conditions ($\bar{\varphi} = \varphi$). Absence of boundary conditions is visible in large error on the boundaries of the lumped-mass filtered field, 5.6e and 5.6f. So too is the larger filter error at the centre, which may be a result of the filter width being larger than the inverse Helmholtz filters (though this has not been confirmed by deriving the width of the lumped-mass filter since its kernel function is not known). Some anisotropy aligned with the structured mesh anisotropy is also evident.

Figure 5.7 displays the commutation error over the domain on structured and unstructured ‘B’ meshes. On the structured mesh the Helmholtz commutation errors are relatively small and solely due to boundary effects (Figures 5.7a and 5.7c): this is because the error is sensitive both to the inhomogeneity of the mesh, which is nonzero near the discontinuity at the edges, and to the field gradients. As expected, the commutation error is zero in the centre where the mesh is uniform. The unstructured Helmholtz commutation errors are dominated by values at points of greatest mesh inhomogeneity (Figures 5.7b and 5.7d), for example where connectivity is larger in one element than all of its neighbours. These ‘knots’ (circled in 5.7b) polluted the error as described above, but to a lesser degree than when the Delaunay mesh generation algorithm was used. The boundary error is still present but is roughly the same magnitude as the majority of the interior. Mass-lumped filtering incurs the largest commutation errors of all but again it is fairly insensitive to the topology of the mesh (Figures 5.7e and 5.7f).

These observations confirm that the commutation error is proportional to the gradient of filter width as explained in §5.3.4. Furthermore, on unstructured meshes and with the inverse Helmholtz filters, the error is largest in the interior, not the boundary. Whether this finding translates to other meshes and other functions will be investigated in the next verification test. Few differences were seen between the scalar and tensor inverse Helmholtz filters on the unstructured mesh because it did not contain high-aspect-ratio elements.

On structured meshes the boundary commutation error dominates while the interior error is zero. The boundary error limits the convergence rate to 1.5 – 1.8, possibly because the relative area covered by boundary elements halves with increasing mesh refinement. It might be worth experimenting with different boundary conditions to reduce the error. Anisotropy was ob-

served in the tensor inverse Helmholtz filter error on the structured mesh, showing that the filter width definition reflects the underlying mesh shape.

5.3.7.3 Test 3: Filtered Velocity

In the second MMS test, a zero-divergence velocity field given by $u = \sin(x) \cos(y)$, $v = -\cos(x) \sin(y)$ is prescribed on a $[0, \pi]^2$ mesh (Figure 5.8a). The structured and unstructured meshes both had 6, 12, 24, 48 and 96 nodes per side. The purpose of this test is to establish whether a field with large values on the boundary compared to the previous test affects the resulting commutation error norms inherent in the filtering operations, and whether treatment of a vector is any different to a scalar field. The filter error norm $\|\mathbf{u} - \bar{\mathbf{u}}\|_{\mathcal{L}^2}$ (a vector) and commutation error norm $\|\overline{\nabla(\mathbf{u})} - \nabla(\bar{\mathbf{u}})\|_{\mathcal{L}^2}$ (a tensor) are calculated.

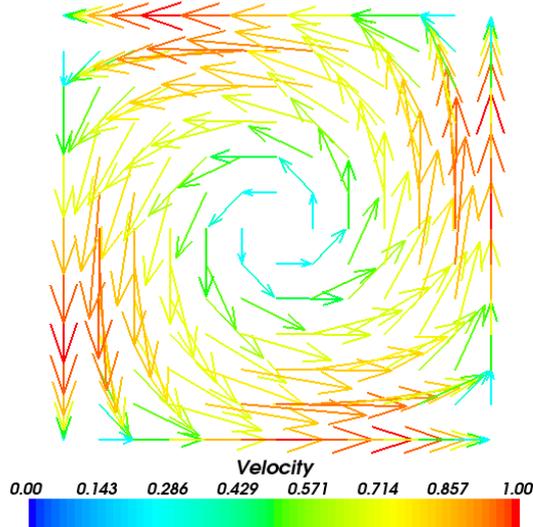
Figure 5.8b shows that the scalar and tensor inverse Helmholtz filters cause identical commutation error magnitudes and that they are roughly independent of mesh type on coarse meshes (A and B). On finer meshes the difference between meshes becomes much more apparent, possibly because of the increasing influence of boundary errors. Lumped-mass commutation error results are not shown because of coding limitations preventing the implementation of a lumped-mass filter subroutine for tensor quantities such as $\nabla \mathbf{u}$ which are needed to compute the commutation error.

Figure 5.8c shows that convergence of the filter error norm $\|\mathbf{u} - \bar{\mathbf{u}}\|_{\mathcal{L}^2}$ tends to second order for the Helmholtz filters as the meshes are refined, and close to 1.5 for the lumped-mass filter at all resolutions, on structured and unstructured meshes.

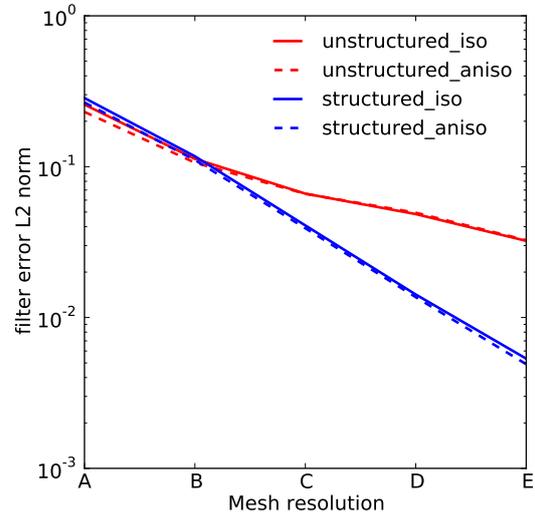
The inverse Helmholtz commutation error norm convergence rate, Figure 5.8d, displays roughly the same trends as the scalar test, Figure 5.5d: as the mesh is refined the rate dips to less than one on unstructured meshes but begins to increase on the finest meshes. The rate is within the bounds set by (5.78): i.e. less than one and greater than zero. The rates on structured meshes are again around 1.5 and dip at the finest resolutions. This seems to be consistent with the prediction of (5.68) with a reduction caused by the boundary error (visible in Figures 5.10a, 5.10c).

Figure 5.9 shows the filter error over the domain on structured and unstructured ‘B’ meshes. The same patterns that were seen in the scalar test are also seen here: both definitions of filter width used in the inverse Helmholtz filter have approximately the same filtering behaviour, with zero filter errors on the boundary on both structured (Figures 5.9a, 5.9c) and unstructured (Figures 5.9b, 5.9d) meshes due to the imposition of the Dirichlet boundary condition $\bar{\mathbf{u}} = \mathbf{u}$. However, the imposition of strong Dirichlet boundary

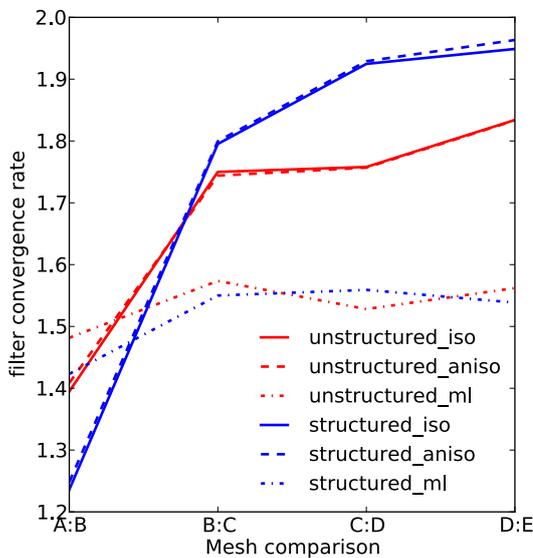
5.3 A Tensorial Differential Filter For LES



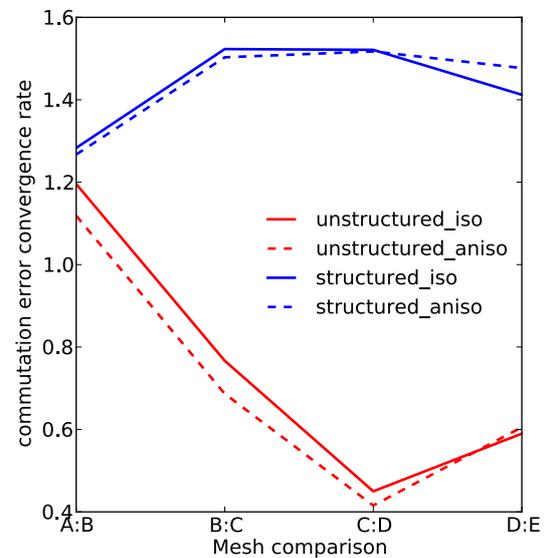
(a) Divergence-free field \mathbf{u} on structured ‘A’ mesh



(b) Commutation error norm



(c) Filter error norm convergence rate



(d) Commutation error convergence rate

Figure 5.8: Vector MMS test results comparing different filters. Source field, commutation error magnitude, filter error convergence rate and commutation error convergence rate are shown. ‘iso’ = scalar filter, ‘aniso’ = tensor filter, ‘ml’ = lumped-mass filter.

5.3 A Tensorial Differential Filter For LES

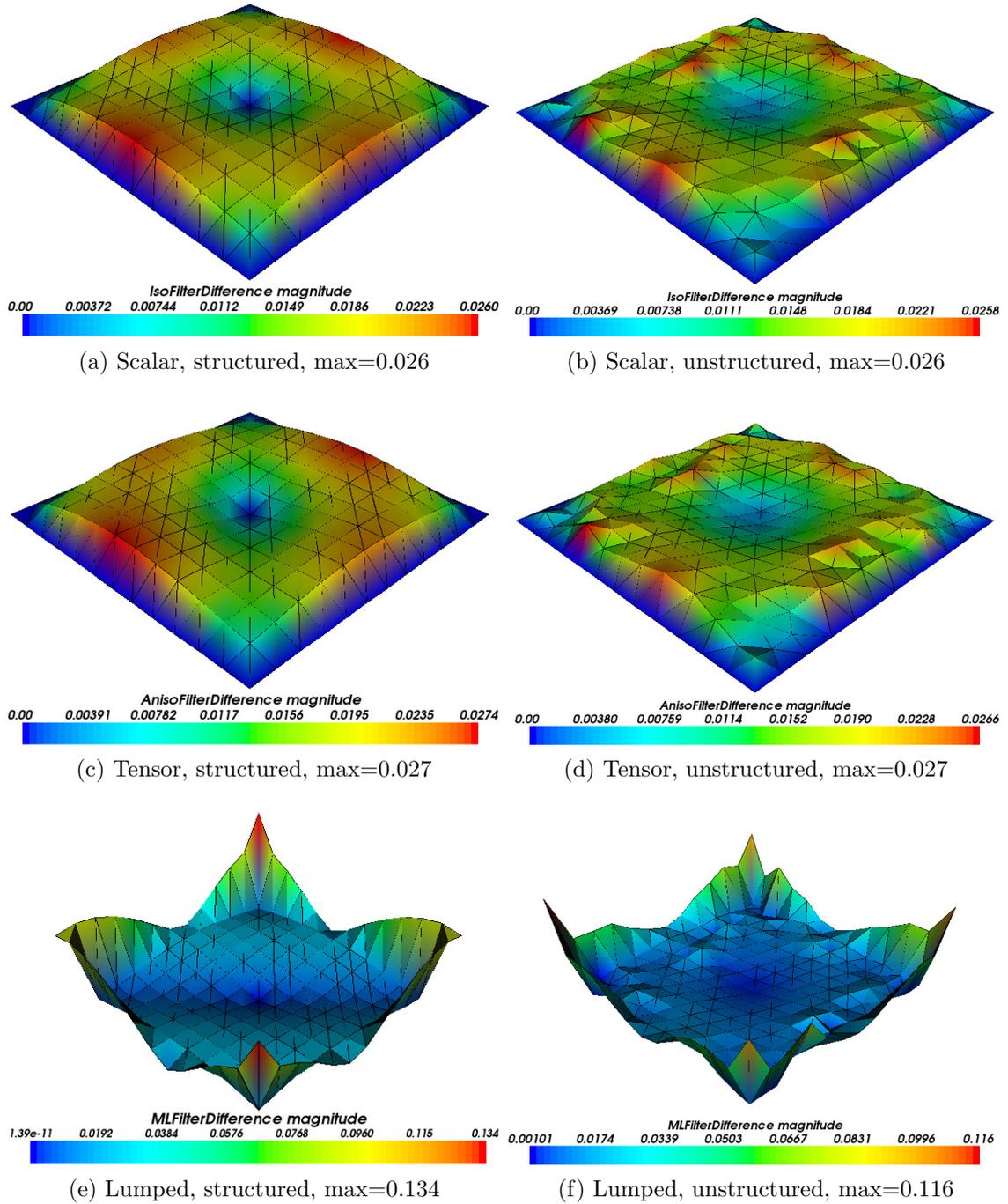


Figure 5.9: Difference between filtered and source vector fields $|\mathbf{u} - \bar{\mathbf{u}}|$ on structured and unstructured 'B' meshes comparing three filters.

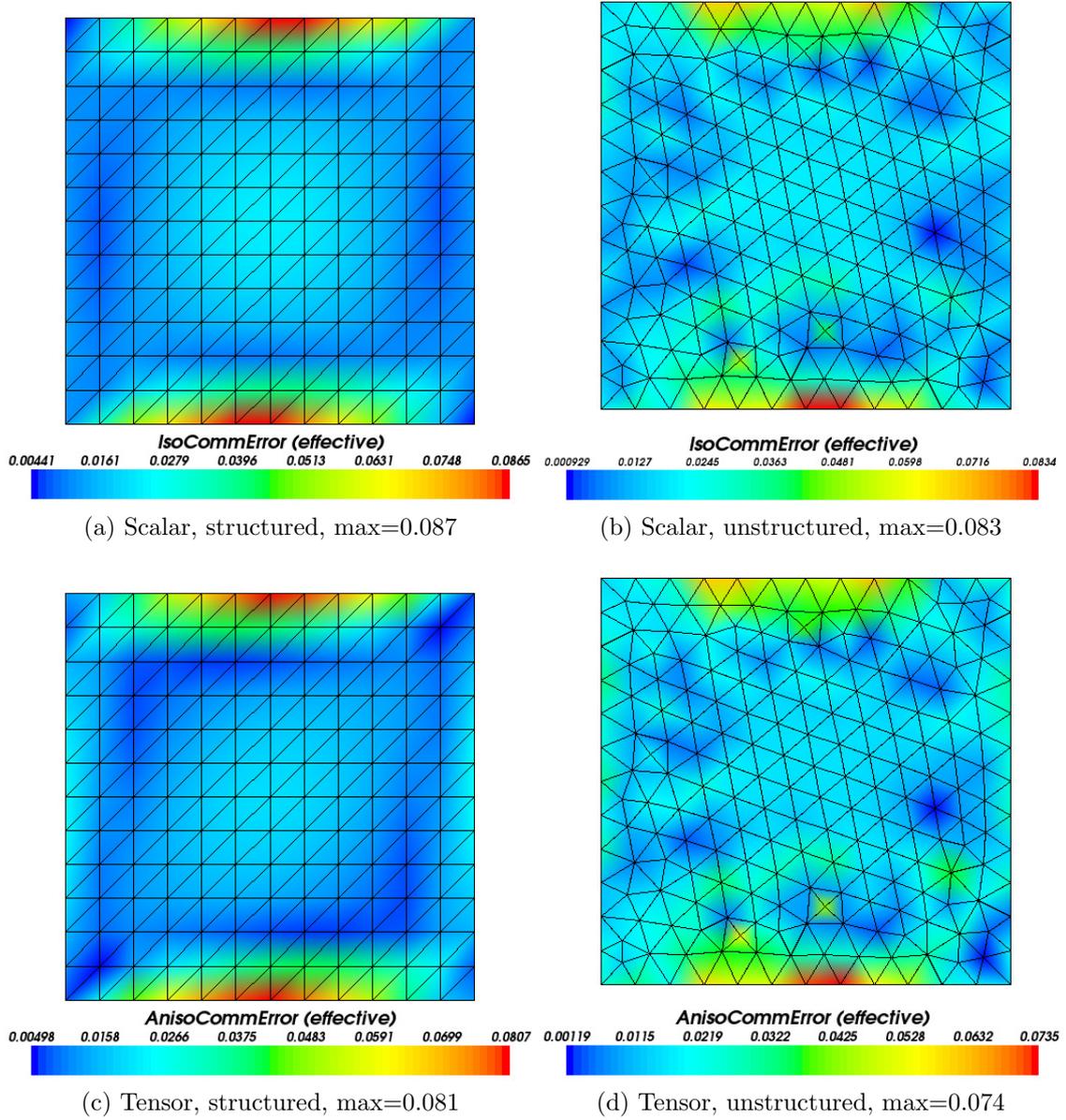


Figure 5.10: Vector commutation error magnitude $|\overline{\nabla(\mathbf{u})} - \nabla(\overline{\mathbf{u}})|$ on structured and unstructured ‘B’ meshes comparing three filters.

conditions has caused the largest errors to appear in the elements just inside the domain. Weak imposition of boundary conditions might improve this situation. Unlike the scalar test, no anisotropy is visible in the tensor filter error on the structured mesh. Lumped-mass filtering is again let down by lack of boundary conditions with comparatively large filter errors on the boundaries and in the corners, Figures 5.9e and 5.9f. Anisotropy is visible in 5.9e suggesting that the lumped-mass filter is sensitive to element shape.

Figure 5.10 shows the commutation error over the domain on structured and unstructured ‘B’ meshes for the Helmholtz filters. The boundary commutation errors are now dominant on the unstructured mesh (5.10b, 5.10d) and are of comparable magnitude to the maximum errors on the structured mesh (5.10a, 5.10c), whereas the difference between the meshes was around an order of magnitude in the scalar test. Interior errors are not significant in this test, perhaps due to differences in mesh generation: the unstructured meshes in this test are more uniform. Now anisotropy is visible in the tensor commutation error on the structured mesh (5.10c, darker blue patches to top left and bottom right of centre).

5.3.7.4 Summary

These tests have verified that the predicted convergence rates of the filter error and commutation error were broadly correct and that commutation errors do converge. On structured meshes the commutation error is zero in the interior and nonzero on boundaries, but these errors converge at $\mathcal{O}(\overline{\Delta}^{-1.5})$. It is not yet known whether different boundary conditions might reduce these errors. On unstructured meshes the commutation error converges at $\mathcal{O}(\overline{\Delta}^{0.4}) - \mathcal{O}(\overline{\Delta}^{1.2})$ depending on mesh uniformity, and significant errors arise both in the interior and on the boundary. The scalar and tensor inverse Helmholtz filters behave similarly though tensor filter errors can display anisotropic effects aligned with the direction of mesh anisotropy. The lumped-mass filter results in larger errors and lower convergence rates but is less sensitive to mesh topology. Further work is needed to compare explicit filtering to higher-order discretisations and to improve boundary treatments with the inverse Helmholtz filter.

5.4 Adaptive LES Modelling

5.4.1 Errors in LES Solutions

Error analysis is a vital aspect of CFD as discussed in Chapter 2. In LES, the modelling, discretisation and convergence errors are combined as shown in Figure 5.11. It has been found by analysing a database of LES simulations that discretisation and modelling errors may partially cancel each other out, leading to the paradox that coarsening the mesh may improve accuracy (Meyers et al., 2003). Some researchers, e.g. Geurts and Fröhlich (2001), have sought to disentangle the errors in order to quantify their effect on the solution. The modelling error is examined by holding the filter width constant while reducing the mesh size to make discretisation errors comparatively small as described in §5.1.3. This necessitates a large filter-width to mesh-size ratio α and in high Reynolds number flow it quickly becomes impractical to maintain if the filter width is to be kept small enough for the LES model to be valid. Consequently, though interesting, this method is only applicable to simulations for academic research.

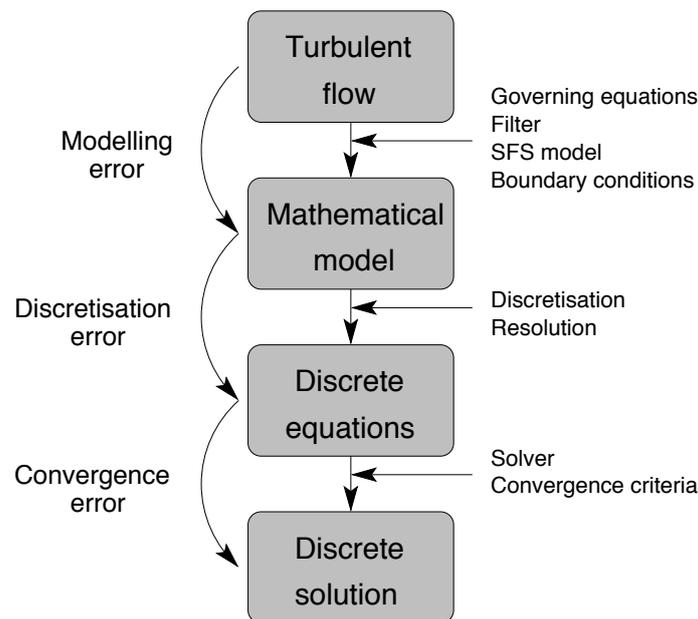


Figure 5.11: Contributions to modelling, discretisation and convergence errors in large eddy simulations (Bentham, 2003). Discretisation and modelling errors may partially cancel out, resulting in unpredictable accuracy improvements as the mesh is refined.

Conversely, fixing the filter-width to mesh-size ratio and refining the mesh shows how the combined errors behave. It is then difficult to determine what is causing the solution to differ from reference data. However, in practice this latter approach is the most commonly used as it is the most efficient use of resources, and the filter width ratio is generally small in order to preserve very fine scales in the filtered solution $\bar{\mathbf{u}}$ as described in §5.1.3. The optimal value of the ratio to minimise the combined error and maximise computational efficiency depends on the discretisation and model, and finding it is usually a matter of trial and error (see §5.2.6).

Computational resources are a strict limitation in commercial CFD activities, so the latter approach is the only practical one. In this case the modelling and discretisation errors are inseparable and the best way of controlling them may be to use one of the complete modelling strategies described here (and to follow best-practice in CFD, e.g. in the selection of discretisation schemes). The starting point for these strategies is to select a useful measure of the accuracy of the solution, discussed in the next section.

5.4.2 Convergence of LES Solutions

Unlike RANS solutions, LES solutions do not generally converge to a grid-independent instantaneous solution as the mesh is refined owing to the inclusion of ever finer scales of motion (Sagaut, 2006). For example, $\|\mathbf{u} - \mathbf{u}^h\|$ is not necessarily convergent, but $\|\langle \mathbf{u} - \mathbf{u}^h \rangle\|$ might be, where $\langle \dots \rangle$ denotes a time or space average. The reason is that one cannot know the initial and boundary conditions with perfect accuracy and that turbulence responds in a highly unpredictable manner to a small change in these conditions. It has been shown that a small initial uncertainty in the small scales of an LES will grow to gradually contaminate the results in a process known as the ‘inverse error cascade’ (Metais and Lesieur, 1986). Hence, the instantaneous solution will probably not match reference data, and convergence should be measured in some solution-sensitive statistical functional.

If such a functional depends on both the energy-containing (inertial) and the energy-draining (dissipative) ranges of turbulent motion, it is expected to reach an asymptote in the inertial range as shown in Figure 5.12 (Pope, 2004). Any contributory physical processes that fall below the filter width are not resolved and account for a modelling error in the measurement. If the functional depends only on the inertial ranges then in an LES it is expected to converge to the ‘true’ value that would be observed in an experiment or DNS (Pope, 2004). A typical example is the reattachment point in a separated flow, which is affected by the mean flow as well as turbulent fluctuations and near-wall effects which cause the point to fluctuate. Coarse near-wall

resolution leads to a discretisation error in the reattachment point as seen in Chapter 6.

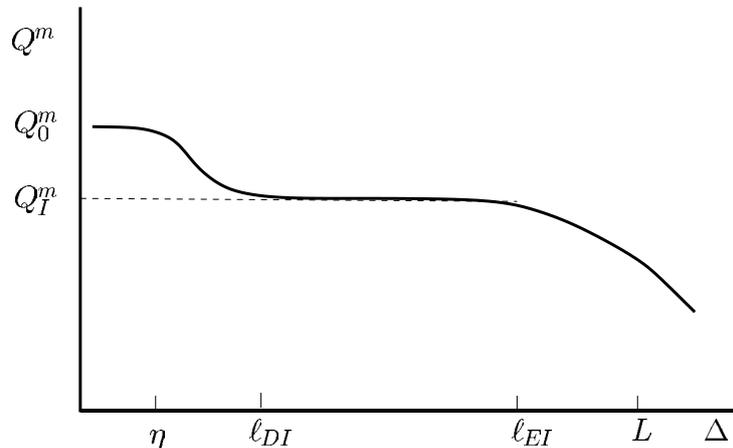


Figure 5.12: Speculative dependence of a functional $Q^m(\Delta)$ on mesh resolution with an asymptote Q_I^m in the inertial range $l_{DI} < \Delta < l_{EI}$, where l_0 and η are the integral and Kolmogorov lengthscales respectively (cf. §3.2.2), l_{EI} and l_{DI} are the upper/lower bounds of the integral range resp. (Pope, 2001). The figure illustrates the possibility that a functional depending on integral- and dissipative-scale processes is convergent in the inertial range of scales where the LES filter width is placed, but contains a modelling error which can only be minimised by resolving down to the dissipative range in a DNS.

In some applications we may wish to know the instantaneous flow accurately such as predictions of precise vortex location in aeroacoustics or meteorology. These applications present a problem for LES at present because there is no clear way to demonstrate that the solution is correct other than comparison to experimental data. A strong pre-requisite for correct prediction of instantaneous quantities is correct prediction of statistical quantities so this discussion remains relevant.

Whether a given functional is computable, i.e. whether it has a sensible answer (such as a spacetime average), can be rigorously quantified. The *adjoint* is a technique allowing us to phrase the question stated in §3.6.2: ‘if this is the answer, what was the question?’ The functional is the answer to, or goal of, our physical problem or question, and the adjoint is the *quantification of computability* of that functional (Becker and Rannacher, 2001). In other words, the adjoint is a quantification of the influence that the solution has on the functional in question. It quantifies how much a particular spacetime

point matters for a given question (Farrell, 2012). Application of this concept to LES of turbulent flow is discussed in §5.4.5.

5.4.3 Complete LES Modelling Strategies

Up to this point in the thesis the LES models that have been discussed have been *incomplete* in the sense that they contain user-defined coefficients or *ad hoc* modifications such as the filter width ratio in the dynamic method. The choice of mesh size Δ is also user-defined and greatly affects the solution. It is generally specified as a spatially varying field which may not bear much resemblance to the scales of motion in the solution. Furthermore, if mesh size is specified *a priori* (as it is for fixed meshes) one has no knowledge of the flow to determine whether the mesh will be fine enough as discussed in §5.2.2.5. Determining the proportions of resolved and modelled turbulence *a posteriori* in response to a specified mesh and then rerunning the simulation with a new mesh in the hopes of improved accuracy is an unreliable and inefficient approach (Davidson, 2011).

One such *a posteriori* technique is to calculate frequency spectra and check whether the $-5/3$ gradient is recreated, though it may not be reliable (Davidson, 2011). A more reliable technique may be to measure the correlation of a turbulent quantity (e.g. Reynolds stress component) between two points which can be used to estimate the size of the largest scales. If the correlation falls to zero over a distance of eight to ten mesh cells then the LES can be deemed well-resolved because the large scales are well-resolved (Davidson, 2011); however this is not a universal rule of thumb because the ratio of largest to smallest scales increases with Reynolds number and can be extremely large in, for example, geophysical flows (Candy, 2008).

LES can be made *complete* by allowing the mesh size to adapt dynamically to the LES solution or a statistical quantity derived from it (Pope, 2004). The principle is to locally determine the mesh resolution in response to a measurement of the *turbulence resolution*. One definition of turbulence resolution in the context of LES is the proportion of energy contained in the SFS to energy contained in the resolved scales, i.e. how much of the Kolmogorov spectrum is resolved. With a complete method it is possible to measure dependence of the solution on a physically meaningful quantity rather than on some arbitrary measure such as total number of nodes in the mesh.

Interpolation error-based mesh adaptivity (cf. §3.6.1) is a complete LES strategy in principle because the mesh resolution $\Delta(\mathbf{x}, t)$ adapts to minimise the interpolation error of the solution, albeit within practical constraints. In practice certain user-specified parameters are necessary to fine-tune the

method, rendering it incomplete, though it is still a powerful technique. This option is discussed in the next section and is used in simulations of the backward-facing step in Chapter 6.

A more targeted strategy is goal-based adaptivity. If the dependent variable is known *a priori*, e.g. from reference data, it can be used to define a goal. Otherwise the criterion might be to maximise lift, heat transfer or some other relevant quantity. This second strategy is outlined conceptually in §5.4.5.

5.4.4 Interpolation Error-Based Adaptivity with LES

Interpolation error-based adaptivity is a useful technique for simulations of turbulent flows if the interpolation error measure is strongly related to turbulence resolution or minimising the error in a flow-dependent functional as described above. The reader is referred back to §3.6.1 for an explanation of the mechanics of mesh adaptivity. Here particular strategies for adapting the mesh are discussed in more detail. In Chapter 6 some of these ideas are tested in the 3D turbulent flow over a backward-facing step using the dynamic LES model.

The choice of target field, or fields, from whose errors to form a metric for interpolation error-based adaptivity is an important consideration. Mitran (2001) used the enstrophy $|\nabla \times \mathbf{u}|$ and helicity $|(\nabla \times \mathbf{u}) \cdot \mathbf{u}|$ on account of their correspondence to ‘interesting’ SFS dynamics such as elongated tubular structures in real turbulence. Leonard et al. (2006) detected the slope of the energy spectrum in the highest resolved wavenumbers using a wavelet representation of turbulence and modified resolution to aim for the ideal $-5/3$ slope of the Kolmogorov energy spectrum.

For LES some decomposition of the velocity may be suitable since the LES model is also a function of decomposed velocity. Either the mesh may be adapted to the interpolation error of time-averaged velocity at infrequent intervals, in which case the representation of fluctuations may be compromised, or of an instantaneous field such as the SFS velocity. A seemingly obvious candidate if using the dynamic method is the difference between mesh- and test-filtered fields, $\bar{\mathbf{u}} - \tilde{\mathbf{u}}$ (Sagaut, 2006). Adaptivity seeks to minimise the error in the SFS which drives resolution towards the DNS limit. It is not clear how to constrict this process in a physically and numerically meaningful way, i.e. to say when the mesh is ‘fine enough’ (Sagaut, 2006).

If an instantaneous field is chosen then the mesh should be adapted frequently – every few timesteps, assuming a moderate Courant number – to follow the flow in detail. While this offers the best accuracy it also engenders a larger computational overhead because of the effort involved in the large

number of adaptivity iterations (Hiester et al., 2011). It also leads to a build-up of interpolation errors because each time the mesh is adapted an error is introduced which is compounded at the next adapt (Farrell, 2012). In highly mesh-sensitive simulations such as interface tracking the technique of metric advection may be useful, advancing the metric forward in time to provide resolution where it will be needed at the next timestep (Hiester et al., 2011).

In combination with a suitable LES model the time-averaged velocity might be suitable in equilibrium flows because the LES model makes up for insufficient resolution of the fluctuations by ‘filling in’ the missing small-scale interactions. Erratic short-lived fluctuations may escape the highly-resolved region, becoming smeared and dissipated in coarser regions. These erratics may be an important part of the flow such as high-temperature excursions in a nuclear reactor. The gradation of edge lengths can be controlled in order to coarsen the mesh gradually away from regions of high solution curvature; this gives some leeway if the region of interest fluctuates between mesh adapts, because the nearby mesh is almost as fine. Certain inherently unsteady flows such as buoyancy-driven flows may result in a relatively uniform mesh being generated because there is no obvious ‘region of interest’ in the mean flow: turbulence is evenly distributed over the domain. In this case a suitable strategy could be to adapt to the velocity fluctuations and increase the frequency of mesh adaptations, or else to simply accept a more uniform mesh as the optimal representation of the solution.

In non-equilibrium flows such as those ‘spinning up’ from rest or from non-turbulent initial and boundary conditions there is the additional issue of the average field being initially unsteady and evolving. Questions of numerical stability and flow history effects are raised if insufficient or inappropriately distributed resolution is used during initial flow evolution. Particularly in periodic simulations there is a clear need for correct resolution from the very start because the flow is recycled, enhancing initial interpolation errors. However, once at statistical steady state the mesh no longer has to be regenerated.

A related point emerges when one considers flow history in terms of advection of errors. A numerical error generated upstream may be magnified as it is advected. Consider, for example, a planar mixing layer in which eddies grow outward from the layer interface or the wake behind a cylinder in crossflow. If the error is observed at some distance away from the source, the mesh will be refined there but the error is not reduced significantly. It is more efficient to refine at the source which may be a small region such as the flow separation point on a cylinder and then use a coarser mesh in the areas into which the flow is advected (Sagaut, 2006). This can be achieved by weighting the desired interpolation error preferentially towards the source.

5.4.5 Goal-Based Adaptivity with LES

Whilst the above strategy is powerful and broadly applicable, in many applications the interpolation error or energy norm does not necessarily provide useful bounds for errors in quantities of real physical interest (Becker and Rannacher, 2001). Then a goal-based approach in combination with mesh adaptivity is more suitable as shown for example by Belme et al. (2012) and Hartmann et al. (2011). Furthermore, by the given definition of completeness, in practice interpolation error-based adaptivity (e.g. as implemented in Fluidity) is not complete while goal-based adaptivity is. An overview of goal-based adaptivity was given in §3.6.2. The technique involves forming the adjoint of the model equations and requires running the model backwards and forwards in time to generate an optimal mesh. Neither of these operations are easy; indeed, one of the main barriers to the use of adjoints is the difficulty of coding a method.

In the complete approach proposed here, a local or global functional of the turbulence resolution, $F(\Delta)$, is formulated in terms of scalar mesh size Δ . One possible measure, derived from the proportions of resolved and SFS turbulent kinetic energy, is developed in §5.4.6. Others are analysed by Davidson (2011). This method is closely related to goal-based adaptivity but with an important difference: conventionally the mesh resolution is chosen to minimise the error in the *computation of a functional* while here it is to minimise the *functional itself*.

The ‘ideal’ turbulence resolution Δ_{goal} is that at which the LES can be considered well-resolved, i.e. all the dynamically important scales (those that determine the accuracy of the dependent variable or variables) are resolved and any assumptions made in deriving the LES model are satisfied. Estimating Δ_{goal} generally involves applying a ‘rule of thumb’.

A functional F is formed which attains an extremum at the desired resolution level Δ_{goal} . The Newton-Raphson method (Raphson, 1690) or a quasi-Newton optimisation method such as BFGS (Papadimitriou and Giannakoglou, 2008) is applied to the functional to find this extremum and the corresponding optimal mesh size Δ_{goal} . Therefore, F should be a smoothly varying, bounded, differentiable function of Δ .

The most general goal-based method would be to optimise F in order to find a tensor field $\Delta_{ij,\text{goal}}(\mathbf{x})$ which represents the mesh as in Belme et al. (2012), but optimising the mesh in terms of both scaling and anisotropy based on a goal functional is not trivial because it involves derivatives with respect to a tensor Δ_{ij} . A simpler starting point is to combine interpolation error-based and goal-based adaptivity schemes. This idea was first published by Venditti and Darmofal (2003) and a similar approach was taken by Power

et al. (2006). The anisotropy of the mesh can be derived from the Hessian of a solution field, or fields, via interpolation error-based adaptivity, while the scalings – a scalar field Δ defining the magnitude of $\Delta_{ij,\text{goal}}(\mathbf{x})$ – can be found by the functional optimisation procedure. Even more simply the inhomogeneity of the mesh, or variation of mesh size, can also be determined by the interpolation error and the functional optimisation simply returns one number: an absolute scaling determining the absolute size of each element and therefore the number of nodes in the mesh.

A putative implementation of this simple combination of adaptive methods is outlined here. An initial metric \mathcal{M}_1 is generated by interpolation error-based adaptivity from the curvature of a field such as velocity. The metric is then rescaled to give a new metric:

$$\mathcal{M}_2 = \left(\frac{\Delta_{\text{goal}}}{\Delta} \right) \mathcal{M}_1 = S\mathcal{M}_1. \quad (5.82)$$

Element anisotropy is determined by \mathcal{M}_1 but element sizes are modified (coarsened or refined) by $S = \Delta_{\text{goal}}/\Delta$ to satisfy the global measure of turbulence resolution, where S may be a single scalar or a scalar field.

A complication arises from the fact that F is generally time-dependent: for example, the global amount of turbulent kinetic energy fluctuates in time. Therefore, the following iterative approach may be required to ensure adequate resolution is provided for all times between mesh adapts:

1. Begin time interval n .
2. Make an initial guess of the required resolution Δ (homogeneous or inhomogeneous) and generate the initial mesh.
3. Run forwards in time over interval n calculating the moving average $\langle F \rangle$ over the interval.
4. Compute extremum of $\langle F \rangle$ by Newton's method or a quasi-Newton variant involving multiple forward and adjoint solutions.
 - (a) If $\langle F \rangle$ is sufficiently close to extremum, return to 1 and begin interval $n + 1$.
 - (b) If $\langle F \rangle$ is not sufficiently close to extremum, generate a 'checkpoint' (write solution fields to file) and go to 5.
5. Regenerate mesh from Δ_{goal} computed at extremum of F using (5.82).
6. Return to 3.

5.4.6 Sub-Filter Kinetic Energy Estimation

A potentially suitable measure of turbulence resolution is the local proportion of resolved to total turbulent kinetic energy (TKE):

$$F(\bar{\Delta}, \mathbf{x}, t) = \int_{\Omega^e} \left[\frac{k_{RES}(\bar{\Delta}, \mathbf{x}, t)}{k_{RES}(\bar{\Delta}, \mathbf{x}, t) + k_{SFS}(\bar{\Delta}, \mathbf{x}, t)} - 0.8 \right]^2 d\Omega, \quad (5.83)$$

where k_{RES} and k_{SFS} are the amounts of resolved and sub-filter TKE respectively, $\bar{\Delta} = \alpha\Delta$ is the filter width and Ω^e is an element in the mesh. Knopp et al. (2010) used a similar resolution sensor based on SFS kinetic energy in statistically steady separated turbulent flow (the backward-facing step) on fixed meshes and found that it was a useful measure. A global metric is defined conceptually in terms of the Kolmogorov energy spectrum (3.5):

$$F = \int_{\Omega} \left[\frac{\int_{l_0}^{\bar{\Delta}} E_k(l)}{\int_{l_0}^{\bar{\Delta}} E_k(l) + \int_{\bar{\Delta}}^{\eta} E_k(l)} - 0.8 \right]^2 d\Omega, \quad (5.84)$$

where E_k is the total turbulent kinetic energy in the domain as a function of turbulence lengthscale (inverse wavenumber), l is the turbulent lengthscale, l_0 is the integral (largest) scale of motion, η is the Kolmogorov (smallest) scale of motion. In practice this would not be calculated in spectral space but by summing (5.83) over the domain Ω .

It is commonly agreed that at least 80% of the TKE (over the domain, though a stricter local condition might be considered) should be in the resolved scales for an LES to be considered well resolved – i.e. modelling assumptions are satisfied and all dynamically important scales of motion are captured – based on arguments derived theoretically from the Kolmogorov spectrum in homogeneous isotropic turbulence with isotropic filter width in the inertial range of the spectrum (Pope, 2000). Although real flows are generally inhomogeneous and anisotropic, the assumed shape of the Kolmogorov spectrum is approximately correct in these flows at high Reynolds numbers (Mydlarski and Warhaft, 1998). However, Davidson (2009) found that it was not a reliable measure of turbulence resolution because more than 80% of energy was resolved in channel flow simulations on five different mesh resolutions: the measure was not sensitive enough to tell them apart.

The assumption of 80% of the TKE being resolved is the reason for subtracting 0.8 and squaring in (5.83) and (5.84): the functionals then attain minima when 80% of the TKE is resolved. Note that without subtracting 0.8 the functionals have minima where 100% of the TKE is resolved, i.e. the DNS limit: this is not a practical option for industrial simulations.

It is possible for *a posteriori* measurements of resolved energy in an LES to be greater than in a DNS, which is “theoretically absurd” (Klein, 2005). Different LES models result in widely varying estimates of SFS TKE according to *a priori* tests by Klein (2005) so it is likely that the optimal proportion of resolved to total TKE differs depending on the choice of model.

The selection of an appropriate measure of SFS kinetic energy is not straightforward. Because the sub-filter scales are modelled it is impossible to know exactly how much energy should be contained in them; all we can do is estimate the amount from the LES model. Different estimates with varying complexity are given in the literature. For this method a measure that varies smoothly in time and has a linear dependence on filter size Δ is advantageous because we must take derivatives of the with respect to Δ .

Knopp et al. (2010) proposed the following simple estimate of SFS TKE in terms of the resolved scales for statistically steady-state flow (total TKE approximately constant in time):

$$k_{SFS} = \frac{1}{2}(\bar{\mathbf{u}} - \tilde{\mathbf{u}})^2, \quad (5.85)$$

where $\bar{\mathbf{u}}$ is the filtered velocity and $\tilde{\mathbf{u}}$ is the twice-filtered velocity (cf. §5.2.2). Resolved TKE in their resolution sensor was given by

$$k_{RES} = \frac{1}{2}(\bar{\mathbf{u}} - \langle \bar{\mathbf{u}} \rangle)^2, \quad (5.86)$$

where $\langle \dots \rangle$ denotes a spatial average. Vreman et al. (1994) suggested another simple model for SFS TKE, given by (5.21) (cf. §5.1.6.5), suitable when using an eddy viscosity model and a positive filter kernel.

Such simple measures may not guarantee conservation of energy. For statistically unsteady or evolving flow it is important to keep track of energy transfers to and from the SFS; in these cases solving an equation is more appropriate. The standard equation for SFS TKE (different to (4.11a) in RANS) is

$$\frac{Dk_{SFS}}{Dt} = \nu_t \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \frac{\partial \bar{u}_i}{\partial x_j} - \frac{C_\varepsilon}{\Delta} k_{SFS}^{3/2} + \frac{\partial}{\partial x_i} \left[\left(\nu + \frac{C_S}{\sigma_k} k_{SFS}^{1/2} \right) \frac{\partial k_{SFS}}{\partial x_i} \right], \quad (5.87)$$

where $C_\varepsilon \approx 0.7$ for homogeneous isotropic turbulence (Schmidt and Schumann, 1989), C_S is the computed/prescribed Smagorinsky coefficient from the LES model, $\sigma_k \approx 1$ is the turbulent Prandtl number and Δ is the LES filter size. Solving this equation as part of an LES model has been recommended by e.g. Gallerano et al. (2005), Ghosal et al. (1995) to correctly account for energy fluxes to and from the SFS.

5.4.7 Discussion

Several points are clear from this speculative discussion on adjoint- or goal-based adaptive LES modelling. Firstly, we need an appropriate measurement of the resolution of turbulence. There is some disagreement in the LES community over what is a suitable measure; it is the subject of a small but crucial area of research. The proportion of sub-filter-scale turbulent kinetic energy has been proposed as a suitable functional and it would be enlightening to compare different functionals with different LES models and in different flow problems.

Secondly, we have to choose a target ‘acceptable resolution’ value. Throughout this thesis the limits of industrial computing resources are kept in mind as a constraint on CFD. It is generally difficult to determine whether an LES simulation is well-resolved, either *a priori* such as comparing to filtered DNS, or *a posteriori* such as measuring the proportion of resolved energy. Adjoint techniques combined with mesh adaptivity could be a sophisticated method of determining the minimum acceptable resolution to achieve a desired accuracy in a quantity of direct interest to engineers. As such, they hold great promise for industrial CFD.

Thirdly, we require a method of adapting the mesh to achieve the target. Interpolation error-based mesh adaptivity is a useful tool in its own right but in combination with the adjoint method it becomes more powerful. The exact combination of these methods can vary and it remains to be seen what the best combination for a given problem may be.

None of these questions is straightforward to answer but a possible method has been presented. An important open problem is to determine the effect of the choice of LES model and discretisation on the definition of acceptable resolution, as discussed by Davidson (2011). If these questions can be answered, it is conceivable that an adjoint-based adaptive LES model could be trusted without *a priori* knowledge of the flow or comparative data from experiments.

6 Validation of the Dynamic Tensorial LES Model in the 3D Backward-Facing Step

In this chapter, the tensorial dynamic LES model developed in the previous chapter is validated in the 3D flow over a backward-facing step using fixed and adaptive meshes. One-point statistics and the reattachment length are reproduced accurately on relatively coarse adaptive meshes thanks to their ability to represent the important flow features while the turbulence model adjusts dynamically to the local resolution. Significant improvements are made by changing the adaptivity parameters. The VLES model is also validated on a fixed mesh.

This chapter expands upon the results presented in [Bull et al. \(2012\)](#).

Contents

6.1	Introduction	193
6.1.1	Modelling Strategies	194
6.1.2	Benchmark Data	195
6.2	Simulation Setup	196
6.2.1	Geometry	196
6.2.2	Dynamic LES Parameters	196
6.2.3	Meshing and Discretisation	197
6.2.4	Boundary conditions	200
6.2.5	Diagnostics	201
6.2.6	Computing Resources	203
6.2.7	CFX Simulations	203
6.3	Results	204
6.3.1	Mesh Type and Resolution of Turbulent Structures	204
6.3.2	LES Model Diagnostics	208
6.3.3	Reattachment Length	212
6.3.4	Mean Velocity Profiles	214
6.3.5	Reynolds Stresses	221
6.3.6	Near-Wall Profiles	221
6.3.7	CFX Results	224
6.3.8	VLES Results	224
6.4	Discussion	226
6.4.1	LES and Adaptivity	229
6.4.2	Tensor Dynamic LES	229
6.4.3	VLES	230
6.4.4	Boundary Conditions	231
6.4.5	Further Work	231

6.1 Introduction

The 3D flow past a backward facing step has been intensively studied using CFD and experiments. It is a simple problem that can be used to study complex turbulent phenomena such as boundary layer development, vortex shedding and flow separation that are relevant to many industrial problems e.g. flow past an aerofoil, vehicles or gas turbine blades (Sagaut and Méneveau, 2006). The problem is twofold: the developing boundary layer in the inflow and downstream of the step and the free shear layer that separates from the step lip. Turbulence is generated by the instability at the step lip and the eddies grow and dissipate as they are advected downstream. The flow regime is illustrated schematically in Figure 6.1.

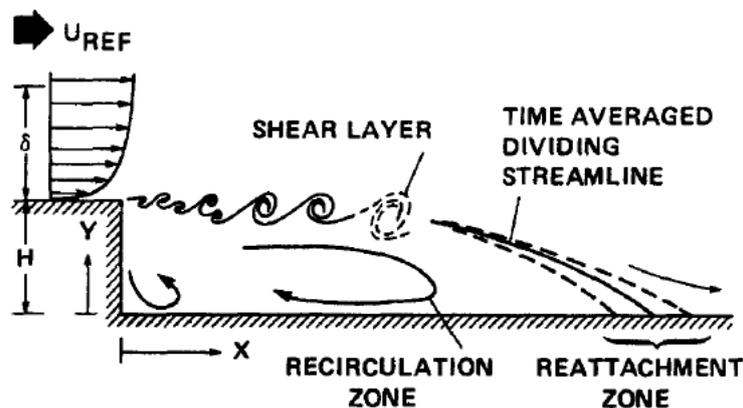


Figure 6.1: Sketch of flow over a backward-facing step with step height H showing the clockwise primary recirculation zone bounded by dividing streamline, anticlockwise secondary recirculation eddy in the corner, turbulence generation in the shear layer, reattachment zone and turbulent boundary layer inlet profile with max. velocity U_{REF} . Image from Simpson (1996).

The backward facing step is a valuable proving-ground for turbulence models for several reasons. Most importantly, there is high-quality benchmark data from DNS studies including Le et al. (1997), from which all manner of statistics – particularly the Reynolds stresses – can be computed. Another reason is that the principal dependent variable, the length of the primary separated eddy behind the step, is a sensitive model performance indicator (Sagaut and Méneveau, 2006).

The turbulence generated by the step is inhomogeneous and anisotropic: the size of eddies varies according to location in the flow and to spatial direction. Therefore, to accurately represent it, a turbulence model is required

that can emulate these properties. However, whilst qualitative agreement with reference data is relatively easy to achieve in terms of the coherent flow structures, quantitative agreement is notoriously difficult. The reattachment length can change by as much as 100% according to the choice of turbulence model and inflow boundary condition (Sagaut and Méneveau, 2006).

6.1.1 Modelling Strategies

The near-wall region poses problems for accurate simulation of the backward-facing step. Kim et al. (2005) stated that the backward facing step flow, regardless of Reynolds number, is strongly affected by several near-wall effects such as the no-slip condition, viscous damping of tangential velocity, and large mean velocity gradients. As a result, the choice of wall model is important. The log law of the wall (cf. §3.4.1.1) proposed by Spalding (1961) does not hold in the presence of adverse pressure gradients, for example in the separation region behind the backward facing step (Speziale and Thangam, 1992). Several attempts have been made to correct it, for example Johnstone and Spalart (2010) modified the log law to depend on wall-normalised pressure p^+ . An alternative near-wall modelling approach is to solve a separate set of reduced equations near the wall e.g. the two-layer model which solves an algebraic (zero-equation) RANS turbulence model (cf. §4.2) on a separate mesh very close to the wall (Diurno et al., 2001). Satisfactory results were reported by Diurno et al. (2001) in the backward facing step case.

Knopp et al. (2010) simulated the backward-facing step at various resolutions using the standard Smagorinsky LES model with wall functions and found that mesh resolution strongly influenced the accuracy of velocity profiles and Reynolds stresses. They argued that under-resolution was the major hurdle to be overcome in industrial LES of complex geometry, and that best-practice guidelines (e.g. AIAA (1998)) on mesh design should be followed, rather than simply increasing the complexity of the LES model. Using wall functions Wang and Moin (2002) demonstrated that reductions of over 90% in simulation time could be achieved without loss of accuracy compared to wall-resolved LES of an aerofoil.

It has been shown (Germano et al., 1991) that the dynamic method negates the need for wall functions because it gives the correct scaling of turbulent stresses near the wall, whereas the standard Smagorinsky model does not. Both the standard and dynamic Smagorinsky models are used without wall functions in this chapter.

The state of the flow entering the backward-facing step domain has a major part to play in determining the qualities of the turbulence downstream, and therefore predictions of reattachment length and Reynolds stresses. The

influence of inflow boundary conditions (cf. §3.4.2) on downstream turbulence in the backward facing step was studied by [Aider and Danet \(2006\)](#). They compared a mean inflow profile perturbed by white-noise with a transient inflow generated from a precursor simulation, and found that the latter caused the destabilisation of the shear layer behind the step and resulted in a reduction in the reattachment length, as well as an increase in the frequency of vortex shedding.

[Jarrin et al. \(2006\)](#) proposed a statistical method of generating synthetic turbulence for inflow conditions, the synthetic eddy method (SEM) (cf. §3.4.2). Tests in turbulent channel flow were able to reproduce first- and second-order one-point statistics (friction coefficient and Reynolds stress profiles) found from a precursor simulation. A fully-developed turbulent boundary layer was able to be generated in a relatively short distance, reducing domain size and thus computational cost. The method is applicable to complex geometries and removes the need for precursors to generate realistic inflow conditions. The original method has been coded in Fluidity and validated by [Pavlidis et al. \(2009\)](#).

Here the use of the tensorial dynamic LES model developed in §5.2 with SEM as an inflow condition is investigated to see if the combination is able to reproduce various quantities in reference DNS data (see below): the reattachment length, mean velocity profiles, Reynolds stress profiles and near-wall profiles. The impact of mesh adaptivity on the accuracy of results is investigated. For comparison, simulations were also performed in the commercial code CFX ([ANSYS, 2011](#)) with the dynamic Smagorinsky model on a fixed mesh.

6.1.2 Benchmark Data

Direct numerical simulation (DNS) at a Reynolds number of 5100 of a 3D backward facing step with the dimensions specified in Figure 6.2 was performed by [Le et al. \(1997\)](#). Even though the Reynolds number is lower than is typical in an industrial CFD problem, the DNS data is of such high quality that it has been used by many authors to validate turbulence models. Experiments in the same geometry were performed by [Jovic and Driver \(1994\)](#), obtaining a reattachment length of 6.0-6.1. The mean reattachment length of 6.28 from the DNS was within 3% of the experimental value. [Panjwani et al. \(2009\)](#) simulated the backward-facing step using the dynamic LES model in a finite-volume code on relatively coarse structured meshes. Their data has been selected for comparison to the current results as the mesh resolution is similar.

6.2 Simulation Setup

6.2.1 Geometry

A schematic of the domain is shown in Figure 6.2. A logarithmic velocity profile is imposed at the left hand boundary (inflow). The region directly downstream of the step is of interest in this problem.

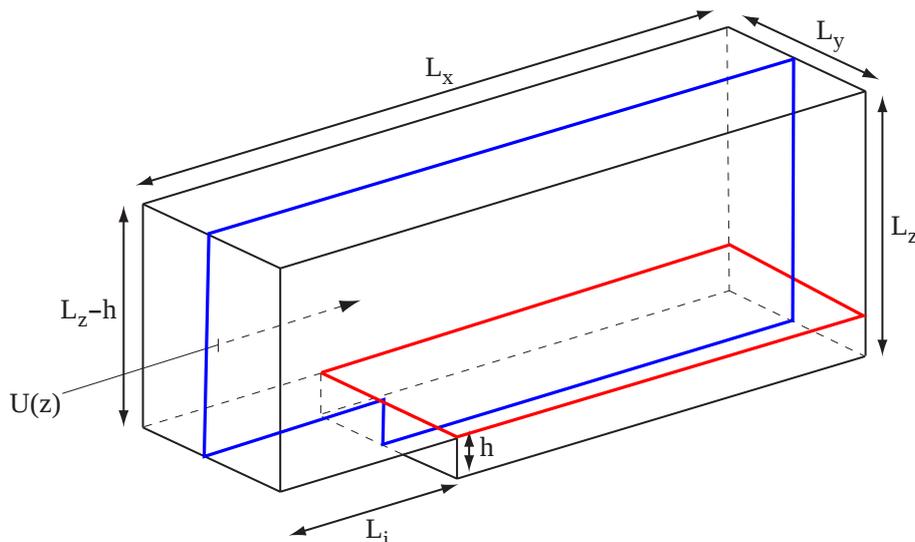


Figure 6.2: Domain for the backward-facing step simulations showing cut plane A (red) and B (blue). Dimensions are $L_x = 30$, $L_i = 10$, $L_y = 4$, $h = 1$, $L_z = 6$.

Following [Le et al. \(1997\)](#) the dimensions are: $L_x = 30$, $L_i = 10$, $L_y = 4$, $h = 1$, $L_z = 6$, so that $L_z - h = 5$ and the expansion ratio is $L_z / (L_z - h) = 1.2$. The base of the domain is located at $z = 0$, the inflow plane is given by $x = -10$ with the step at $x = 0$, and the back of the domain in the spanwise direction is given by $y = 0$.

6.2.2 Dynamic LES Parameters

The tensorial dynamic LES model developed in §5.2 is applied to the backward-facing step. The effects of five independent LES model parameters, listed in Table 6.1, are investigated. With two values for each parameter, a maximum of $2^5 = 32$ separate experiments would be required to fully investigate the entire LES model parameter space. However, only the combinations marked by an ‘x’ in Table 6.1 were tested, a total of 14 experiments, since certain combinations ($\alpha = 2$ with $\beta = 1$, explicit first filter with no upper

limit on c_S) were deemed unlikely to yield improved results based on the results of other combinations.

upper limit on c_S			0.04	0.04	∞
first filter type			implicit	explicit	implicit
filter shape	ratio α	ratio β			
isotropic	1	1	x		
isotropic	1	2	x	x	x
isotropic	2	2	x	x	x
anisotropic	1	1	x		
anisotropic	1	2	x	x	x
anisotropic	2	2	x	x	x

Table 6.1: Dynamic LES parameter space and experiments performed marked with an ‘x’.

6.2.3 Meshing and Discretisation

The effects of periodic vs. non-periodic spanwise boundary conditions and varying mesh resolution were tested. Several unstructured fixed non-periodic (F-series), semi-structured fixed periodic (P-series) and unstructured adaptive (A-series) meshes were used; details are in Table 6.2 and 6.3. Only results using fine fixed (F3, P3) and adaptive meshes are presented here because it was found that meshes F1, F2, P1 and P2 were too coarse to obtain meaningful results: either no stable recirculation eddy was generated or model terms were unstable. The adaptive meshes (A1, A4 and A5) varied from one simulation to another. A complete list of adaptivity settings for meshes A1, A4 and A5 is given in Table 6.3.

The adaptivity algorithm in Fluidity adjusts the mesh resolution locally to meet an interpolation error criterion ε_u on a target field or fields (cf. §3.6.1). A discussion of appropriate choices of target field or fields from whose errors to form a metric for interpolation error-based adaptivity has been presented in §5.4.4 and is expanded upon here. As a simple and sensible first guess at the best target, the filtered velocity $\bar{\mathbf{u}}$ was chosen as the target field for adaptive mesh A1.

The other adaptivity parameters chosen for mesh A1 are now explained. The small value of desired error (in each velocity component) $\varepsilon_u = 0.001$, which should be compared to the maximum inlet velocity of 1.5 to gauge the relative size of the error, was chosen in an attempt to get as high an

6.2 Simulation Setup

Mesh	Structure	Movement	Periodicity	Resolution	Nodes
F1	unstructured	fixed	non-periodic	0.5	7197
F2	unstructured	fixed	non-periodic	0.25	45752
F3	unstructured	fixed	non-periodic	0.125	244378
P1	semi-structured*	fixed	periodic	0.5	10521
P2	semi-structured*	fixed	periodic	0.25	77044
P3	semi-structured*	fixed	periodic	0.125	596706
A1	unstructured	adaptive	non-periodic	see Table 6.3	≤ 250000
A4	unstructured	adaptive	non-periodic	see Table 6.3	≤ 250000
A5	unstructured	adaptive	non-periodic	see Table 6.3	≤ 500000

Table 6.2: Fixed mesh parameters in 3D backward-facing step simulations. *Periodic meshes are structured in the spanwise direction so that surface nodes match up.

Parameter	A1	A4	A5
target field(s)	$\bar{\mathbf{u}}$	$\bar{\mathbf{u}} - \langle \bar{\mathbf{u}} \rangle, \langle \bar{\mathbf{u}} \rangle$	$\bar{\mathbf{u}} - \langle \bar{\mathbf{u}} \rangle, \langle \bar{\mathbf{u}} \rangle$
desired error ε_u in components	0.001	0.1, 0.02	0.05, 0.01
max. no. of nodes	250000	250000	500000
approx. no. of nodes in final meshes	233000	239000	360000
frequency of adapts (timesteps)	100	200	200
max. node increase between meshes	1.3	1.3	1.3
gradation parameter	3.0	1.25	1.25
absolute interpolation error bound(s)	0.001	0.02, 0.1	0.01, 0.05
min. edge lengths	0.05(x) 0.05(y) 0.05(z)	0.03 (x) 0.01 (y) 0.01 (z)	0.01 (x) 0.005 (y) 0.005 (z)
max. edge lengths	1.0 (x) 1.0 (y) 1.0 (z)	1.0 (x) 0.2 (y) 0.2 (z)	0.5 (x) 0.2 (y) 0.2 (z)

Table 6.3: Adaptivity parameters for meshes A1, A4 and A5 in 3D backward-facing step simulations.

accuracy as possible. Minimum and maximum edge lengths were isotropic; the minimum length was five times smaller than the elements of fixed mesh P3 and the maximum length was one to allow for extreme coarsening in the top of the domain. A gradation parameter, controlling the maximum ratio of edge lengths in adjacent elements, was set to three to allow rapid variation of element size e.g. from the wall to the free stream, although this would generate a significant commutation error. Maximum node increase from one mesh to the next was set to 1.3 in order to reduce fluctuations in gradient values caused by a sudden change of resolution (see discussion in §4.6). Mesh adapts were set to happen every 100 timesteps as a compromise between fidelity to the flow and computational effort-saving (one adapt costs approximately the same as one timestep), as well as to prevent the accrual of excessive interpolation errors (cf. §5.4.4).

It became clear that mesh A1 was not ideally set up for the backward-facing step problem. Firstly, simulating this problem accurately involves faithfully representing both mean and transient phenomena, for example the reattachment length is a function of the mean flow while the Reynolds stresses are a function of the fluctuating velocity¹. Therefore, in an attempt to follow the important flow details more faithfully and economically, a weighted combination of the fluctuating velocity, $\bar{\mathbf{u}}' = \langle \bar{\mathbf{u}} \rangle - \bar{\mathbf{u}}$, and time-averaged velocity, $\langle \bar{\mathbf{u}} \rangle$, was selected as the target fields for meshes A4 and A5. By adjusting the desired error ε_u on each field the error in the average velocity was weighted more heavily (see Table 6.3). Larger values of ε_u were set for A4 and A5 compared to A1 as it was realised that achieving an error of 0.001 might be impossible within the constraints of maximum node numbers and edge lengths.

Since the average velocity varies slowly in statistically steady-state flow, the mesh does not have to be changed very often; hence the adapt period was increased to 200 timesteps. Adapting to the fluctuation as well was intended to supplement the resolution in regions where the greatest errors in fluctuating and average velocities do not overlap. Clearly, there is a compromise in the frequency of adapts because the fluctuation changes far more rapidly than the average velocity (on the order of 10 timesteps).

Special consideration had to be paid to the start of the simulation because initial flow evolution affects conditions later i.e. errors propagate in time. It is important to provide adequate resolution during this phase and the combination of targets is designed to provide it. Time averaging was set to start from 200 seconds into the simulation because that was found to be the

¹Adapting to the mean flow may not be optimal in other flows, for example the wake behind a cylinder, in which the dominant dynamics are bimodal

upper bound on the length of time for the flow to reach statistical steady state (equivalent to 7.5 advective periods), judged by the L^2 norm of the velocity. Less than 100 seconds were needed on meshes A4 and A5, so time averaging began at 100 seconds on these meshes. Prior to that point, the average velocity was equal to the velocity $\bar{\mathbf{u}}$ and the fluctuation was zero and the mesh initially followed $\bar{\mathbf{u}}$. Once averaging started, the fluctuation was nonzero and the mesh adapted to the combination of averaged and fluctuating velocities. This had an effect on the overall number of nodes: prior to 200 seconds, the total number in the A4 mesh gradually grew from $\sim 100,000$ to $\sim 150,000$ as the flow evolved to a statistical steady state and the turbulent downstream region grew. At 200 seconds there was an immediate jump to $\sim 240,000$ nodes, which was maintained constant for the rest of the simulation. This observation can be explained by the additional resolution added to resolve the fluctuating velocity.

Other adaptivity parameters were also changed in meshes A4 and A5. The gradation parameter was reduced to 1.25 to improve mesh smoothness and reduce commutation errors (cf. §5.3.4). Minimum and maximum edge lengths were made anisotropic with larger lengths in the streamwise direction to reflect the direction of the smallest curvature of the velocity field (spanwise and vertical gradients are larger than streamwise).

Discretisation was by the continuous Galerkin method with piecewise linear function spaces for velocity and pressure (P_1 CG – P_1 CG) and fourth-order pressure stabilisation cf. §3.5.6. No advective stabilisation was added.

6.2.4 Boundary conditions

Turbulent inflow conditions were provided by the synthetic eddy method (SEM) which linearly interpolated the DNS data of [Le and Moin \(1992\)](#) (available from [ERCOFTAC \(2012\)](#)) onto the computational mesh. The mean velocity profile is shown in [Figure 6.3](#). One simulation was performed with no SEM applied for comparison (cf. §6.3.7). On all meshes, strong no-normal flow boundary conditions were applied at the top boundary, strong no-slip boundary conditions at the bottom of the domain and at the step down and a zero-pressure boundary condition on the outflow. A region of absorption was created near the outflow to reduce the chance of turbulent flow re-entering the domain and causing the solver to diverge. It was defined by a linear increase of a velocity absorption term with vector components a ranging from zero to 0.1 towards the outflow:

$$a(x) = \begin{cases} 0.0 & \text{if } x \leq 25 \\ 0.1(x - 25)/5 & \text{if } x > 25. \end{cases} \quad (6.1)$$

6.2 Simulation Setup

On F- and A-series meshes no-normal-flow (free-slip) conditions were specified on the sides while the P-series meshes had periodic boundary conditions. By plotting pressure contours it was found that there is some effect of non-periodicity on the flow despite the no-normal flow boundary conditions. Two remedies are possible: either run in serial (Fluidity does not yet support 3D parallel periodic adaptivity), or increase the domain width in the hopes of reducing spanwise effects. Neither of these options were attempted because of the associated increase in computational time.

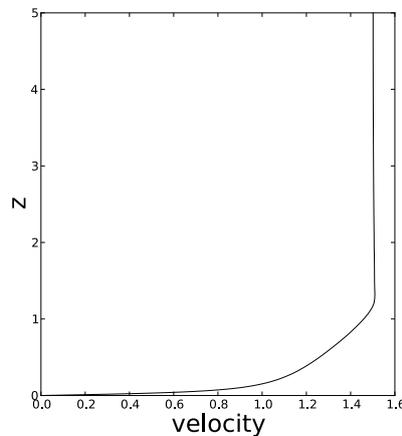


Figure 6.3: Mean velocity profile from data of [Le and Moin \(1992\)](#) used in 3D backward-facing step simulations. Max velocity = 1.5ms^{-1} .

6.2.5 Diagnostics

6.2.5.1 Reattachment Length Calculation

The reattachment length refers to the distance from the step to the boundary between flow going towards and away from the step at the bottom boundary i.e. the length of the primary recirculating eddy. It is referred to as the primary eddy because it is common to observe a smaller secondary eddy near the step where flow reverses direction again.

Several approaches to the calculation of reattachment length (R_L) are possible ([Le et al., 1997](#)). Here R_L was calculated by finding the point at which the instantaneous velocity changed direction along a line running in the x -direction at a height of $0.01h$ above the bottom of the domain. The value of R_L was fairly insensitive to small changes in the sampling height: since piecewise linear shape functions were used, the sampled values within an

element varied linearly with height. Choosing 0.01 ensured that the samples were taken from the first layer of elements in the finest adaptive meshes. In Fluidity, simulated prognostic and diagnostic fields are output in `.vtk` format which allows user-defined Python routines to extract data via the VTK library (VTK, 2012), in this case using the `ProbeData` function on the velocity field $\bar{\mathbf{u}}$, which finds the velocity at each point on a line by linear interpolation from the closest mesh nodes using the linear shape functions. Each line was represented as an array which the Python routine searched along, looking for the first point $x[i]$ where $u[i]/u[i+1] < 0$ and $u[i+1] > 0$. R_L was found by linearly interpolating between $x[i]$ and $x[i+1]$.

In 3D significant spanwise variations in R_L were detected (around $\pm 50\%$), so 41 evenly spaced parallel lines of closely spaced points were created and $R_{L,av}$ was taken as the average of the reattachment points R_L on each line. It was found that 41 sampling lines were adequate to calculate an accurate average value; using 81 led to a change of no more than ≈ 0.1 in $R_{L,av}$. After spanwise averaging, the instantaneous reattachment point fluctuated in time because of periodic vortex shedding. Therefore, an additional step was taken of time-averaging the instantaneous value of $R_{L,av}$ to provide the results in Table 6.4.

6.2.5.2 Reynolds Stress Calculation

Several diagnostic fields were calculated ‘online’ i.e. during a simulation. Time-averaged velocity $\langle \bar{\mathbf{u}} \rangle$ and pressure $\langle \bar{p} \rangle$ were calculated, beginning after the initial 200 seconds of ‘spin-up’ time as with the reattachment length. Fluctuating velocity was then found from $\bar{\mathbf{u}}' = \bar{\mathbf{u}} - \langle \bar{\mathbf{u}} \rangle$ and equivalently for fluctuating pressure. The three principal Reynolds stresses are simply $|\bar{u}'_i|$, $|\bar{u}'_j|$ and $-(\bar{u}'_i \bar{u}'_j)$. A further step was taken of time-averaging the Reynolds stresses to obtain meaningful profiles.

6.2.5.3 Near-Wall Profile Calculation

Near-wall velocity profiles in wall units (u^+, y^+) were calculated post-simulation by extracting the time-averaged velocity at the final time along vertical lines through the domain mid-plane (plane B in Figure 6.2). It was found that the average velocity at the final time was sufficiently spanwise-invariant that multiple sampling lines were unnecessary. The wall shear stress was calculated locally by extracting the value of the velocity gradient tensor field (a diagnostic field in Fluidity) at the wall.

6.2.5.4 LES Model Diagnostics

Several diagnostic fields were output for the purposes of debugging the dynamic LES model:

- dynamic Smagorinsky coefficient c_S ;
- eddy viscosity ν_T ;
- filter widths $\overline{\Delta}$ and $\widetilde{\Delta}$ in scalar (5.7) and tensor form (5.45);
- filtered velocity fields $\overline{\mathbf{u}}$ and $\widetilde{\mathbf{u}}$ and their difference (called the resolved fine scales).

6.2.6 Computing Resources

Simulations were run on the CX1 cluster at Imperial College London, each using 16 processors and up to 150 hours of run-time ($\approx 400 - 600$ seconds of simulated time).

6.2.7 CFX Simulations

For reference purposes, dynamic LES simulations were also run in CFX. These used a fixed unstructured tetrahedral mesh of approximate edge length 0.125, similar to the P3 mesh in Fluidity. Discretisation was by a control volume-based finite volume method similar to the CV-FE method in Fluidity (cf. §3.5.5). Periodic boundary conditions were applied in the spanwise direction. A laminar inlet profile was prescribed. The effect of adding turbulent fluctuations by imposing an initial condition with an added RMS velocity fluctuation was found to be negligible, in contrast to Fluidity, where the synthetic eddy method (SEM) for inlet turbulence generation significantly enhanced the results. This is thought to be because the RMS fluctuation does not represent realistic turbulent fluctuations, whereas SEM does. More advanced synthetic turbulence methods are available in FLUENT but these have not been tested here.

The version of dynamic LES available in CFX is the Germano-Lilly method with clipping of the dynamic Smagorinsky coefficient to $0 \leq c_S \leq 0.04$ and relaxation in time to maintain stability by the following formula:

$$c_S^n = \theta c_S^n + (1 - \theta) c_S^{n-1}, \quad (6.2)$$

where $\theta = 0.01$ and n is the current timestep. The first filter width ratio α (mesh to first filter) is one and the second ratio β (first to second filter) is

fixed at two. First filtering is implicit, while second filtering is by volume-weighted averaging of the velocity in elements surrounding a node, and filter width is given by Deardorff's method (5.7) (ANSYS, 2011).

Reattachment length R_L was found by a different method to that used for Fluidity results, namely, the average of location at which the zero streamwise velocity contour touched the bottom. Mean velocity and Reynolds stress profiles were not plotted at positions normalised by R_L because it was too big and the position $\frac{5}{3}R_L$ would have been outside the domain. Instead, the positions used by Le et al. (1997) were used: $x/h = [4, 6, 10, 19]$.

6.3 Results

6.3.1 Mesh Type and Resolution of Turbulent Structures

6.3.1.1 Adaptivity Parameters

Figure 6.4 shows the velocity field magnitude on plane B (see Figure 6.2) through the A1 adaptive mesh behind the step. The mesh is refined in the shear layer and extremely coarse near the top of the domain. Elements are fairly uniform in size and shape in the shear layer. Whilst this mesh may be better suited to resolving the flow than a fixed mesh with the same number of nodes, it is not particularly well calibrated. Resolution could do with being finer at the wall and step lip and grade more smoothly to large elements towards the top (the gradation parameter is three in this mesh).

Figure 6.5 shows the fluctuating and average velocity fields on plane B through the A4 adaptive mesh, using the improved parameters in Table 6.3, demonstrating more effectively the ability of mesh adaptivity to place resolution in the regions of interest, in this case the shear layer behind the step and the bottom boundary layer. The mesh aligns reasonably well with the regions where the two fields are varying most rapidly. Resolution is not as coarse at the top and is more gradually varied throughout the domain.

Figure 6.6 shows the same set of results, but now the time-averaged Reynolds stresses at a later time on plane B and a closer view of the reattachment region. The mesh follows the Reynolds stresses quite well, which is not surprising given that they are functions of the fluctuating velocity. In the close-up it is possible to see fine stretched elements at the wall which capture the large gradient of the stresses there. Regions where the stresses are varying less rapidly (blue) have enlarged elements.

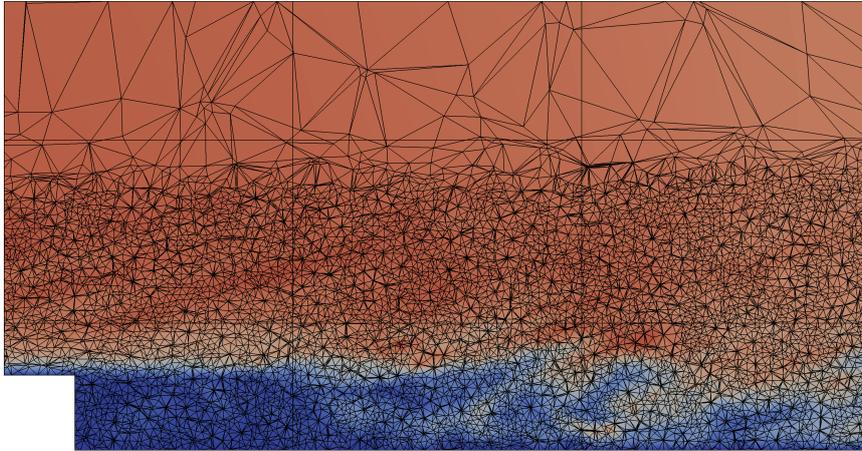
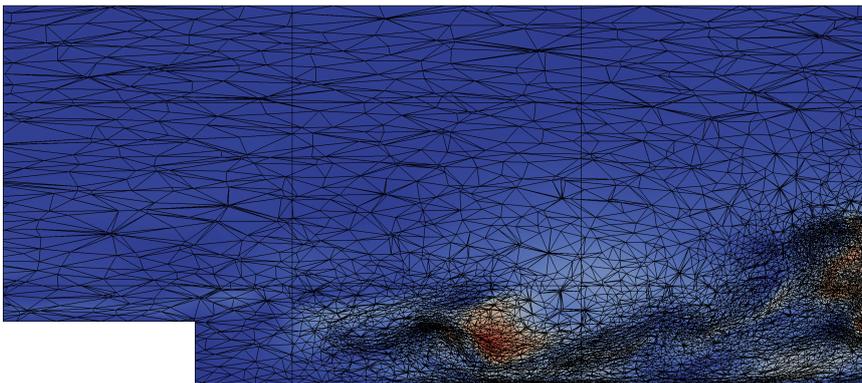
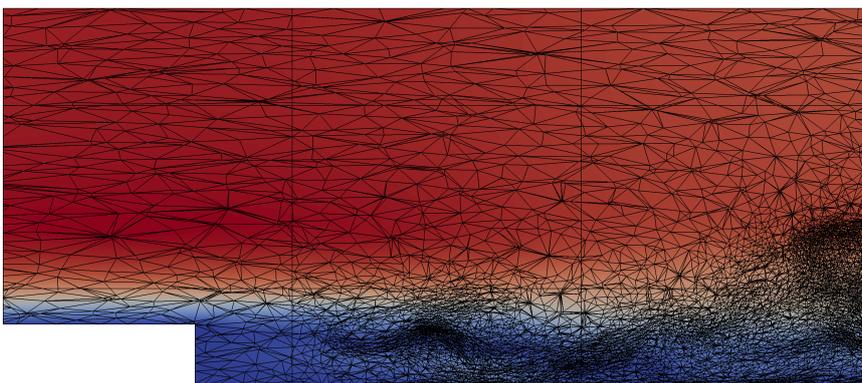


Figure 6.4: Mesh A1 adapting to velocity field, showing velocity magnitude.

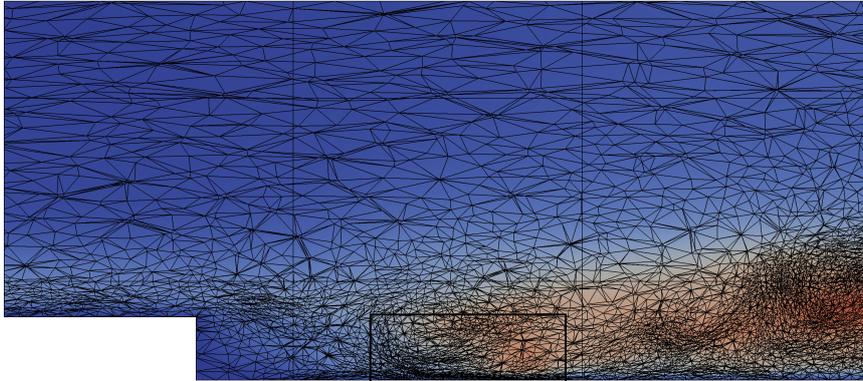


(a) Fluctuating velocity magnitude $|\bar{\mathbf{u}}'|$

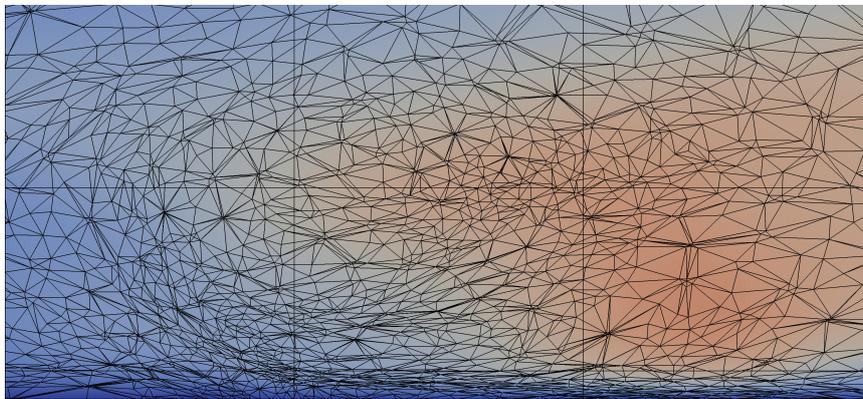


(b) Average velocity magnitude $|\langle \bar{\mathbf{u}} \rangle|$

Figure 6.5: Mesh A4 adapting to combination of average and fluctuating velocity fields, coloured by field magnitude.



(a) Time-averaged Reynolds stress magnitude in reattachment region



(b) Close-up of inset in (a) showing near-wall resolution

Figure 6.6: Mesh A4 coloured by time-averaged Reynolds stress magnitude.

6.3.1.2 Variation of Filter Width and Commutation Error

Figure 6.7 shows the dynamic LES first (mesh) filter width and its gradient on a horizontal plane (plane A in Figure 6.2) in the A4 adaptive mesh, demonstrating the variations of mesh size which arise when adapting to a turbulent velocity field. The gradient of filter width is significant because the commutation error in the inverse Helmholtz filter is proportional to the gradient (as shown in §5.3.4, and verified in §5.3.7). A maximum value of 0.4 near the step means that the most significant pollution of the result by the commutation error is in this region.

Since turbulence is generated here, it would be advisable to reduce the minimum element size in order to reduce the error. The gradation parameter is already close to one and cannot be reduced much further. The commutation error manifests itself in the test-filtered field $\tilde{\mathbf{u}}$, hence the Leonard

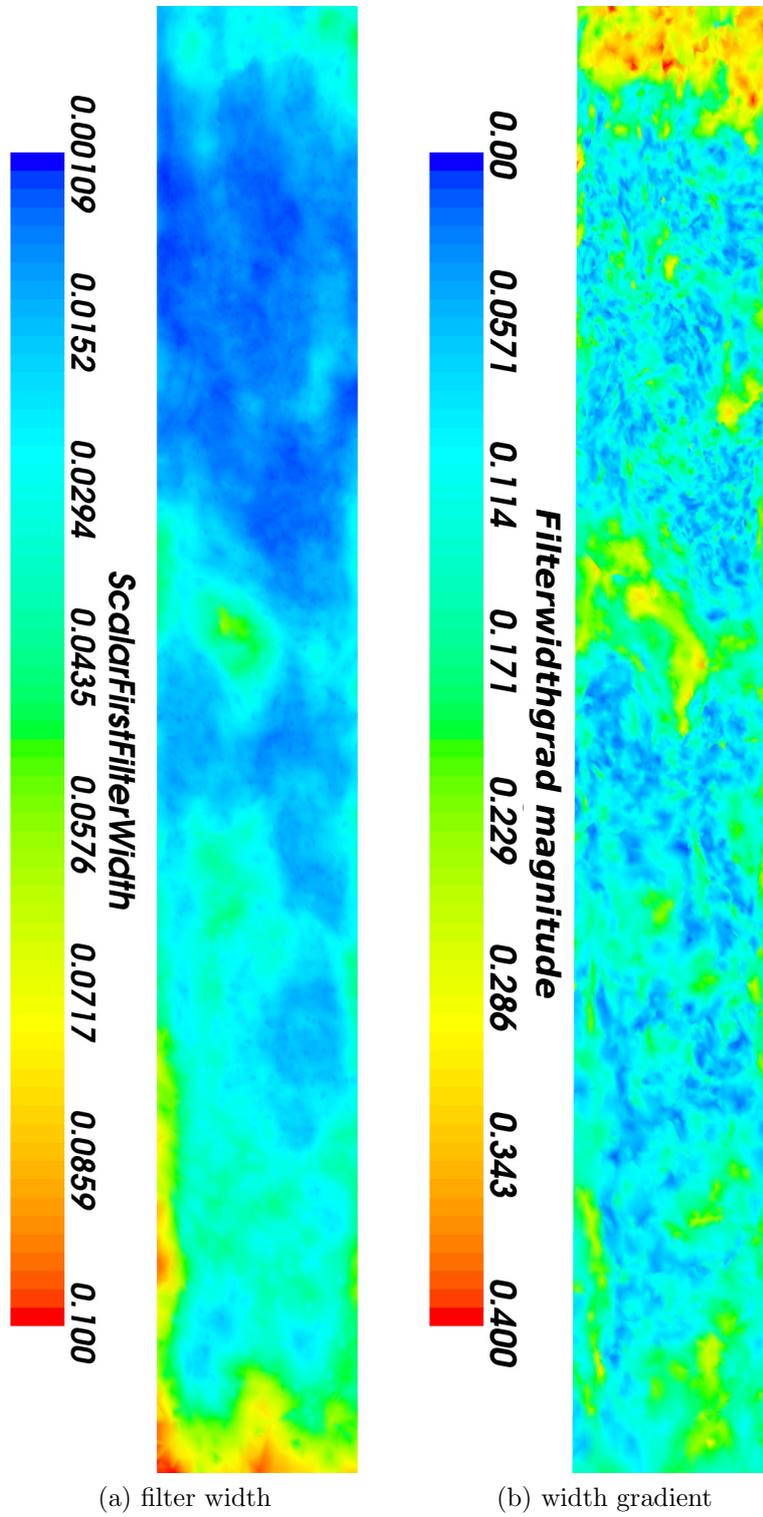


Figure 6.7: First filter width $\bar{\Delta}^2$ and its gradient on plane A for dynamic LES on adaptive mesh A4. Step lip is at the top.

tensor and dynamic Smagorinsky coefficient. If an explicit first filter is used then calculating $\overline{\mathbf{u}}$ incurs a commutation error as well, having a direct effect on the computed eddy viscosity.

6.3.1.3 LES Model Comparison

Figure 6.8 shows velocity magnitude $|\overline{\mathbf{u}}|$ on plane A through fine periodic mesh P3 and adaptive mesh A1. Results with no LES model (6.8a, 6.8b) and with the tensor Smagorinsky LES model with $C_S = 0.1$ (6.8d, 6.8c) are shown.

The Smagorinsky model appears to be too dissipative as the structures in the velocity field are larger and their evolution is slower: no fluctuations are visible close to the step. The no-model results are much more complex with small structures evolving close to the step and growing downstream in the shear layer. There is little difference between the periodic and adaptive mesh results. This is encouraging for the use of mesh adaptivity, where periodic boundaries cannot yet be used in parallel simulations in Fluidity.

Figure 6.9 shows $|\overline{\mathbf{u}}|$ on plane A through periodic P3 (6.9a), fixed F3 (6.9b) and adaptive A1, A4 and A5 (6.9c-6.9e) meshes with the tensor dynamic LES model, isotropic filter, limited c_S , and $\alpha = \beta = 2$. Spatial evolution of turbulent structures is slowest on the fixed mesh F3 while periodic boundaries (P3 mesh) appear to induce turbulence sooner. The size and distribution of structures on P3 and A1 meshes are comparable. Mesh A4 with its modified adaptivity parameters causes a slight reduction in the length at which turbulence is induced, while mesh A5, with its larger number of nodes and smaller minimum edge lengths (other parameters the same as A4), resolves smaller structures closer to the step than any other mesh.

The choice of adaptivity parameters, and not just the number of nodes or minimum edge lengths, clearly has a significant role to play in resolving turbulent structures. Far superior resolution of the important structures can be achieved with comparable or fewer nodes than a fixed mesh, despite lacking periodic boundary conditions. For its ability to resolve turbulence, mesh A5 is the best in this test.

6.3.2 LES Model Diagnostics

Figure 6.10 shows the computed dynamic Smagorinsky coefficient c_S , eddy viscosity ν_T and resolved fine-scale velocity $|\overline{\mathbf{u}} - \tilde{\mathbf{u}}|$ on plane A through A4 mesh with the dynamic LES model with $c_S \leq 0.04$. Very fine structures visible in Figure 6.10c demonstrate the effectiveness of the inverse Helmholtz filter on an inhomogeneous unstructured mesh. ν_T (Figure 6.10b) corre-

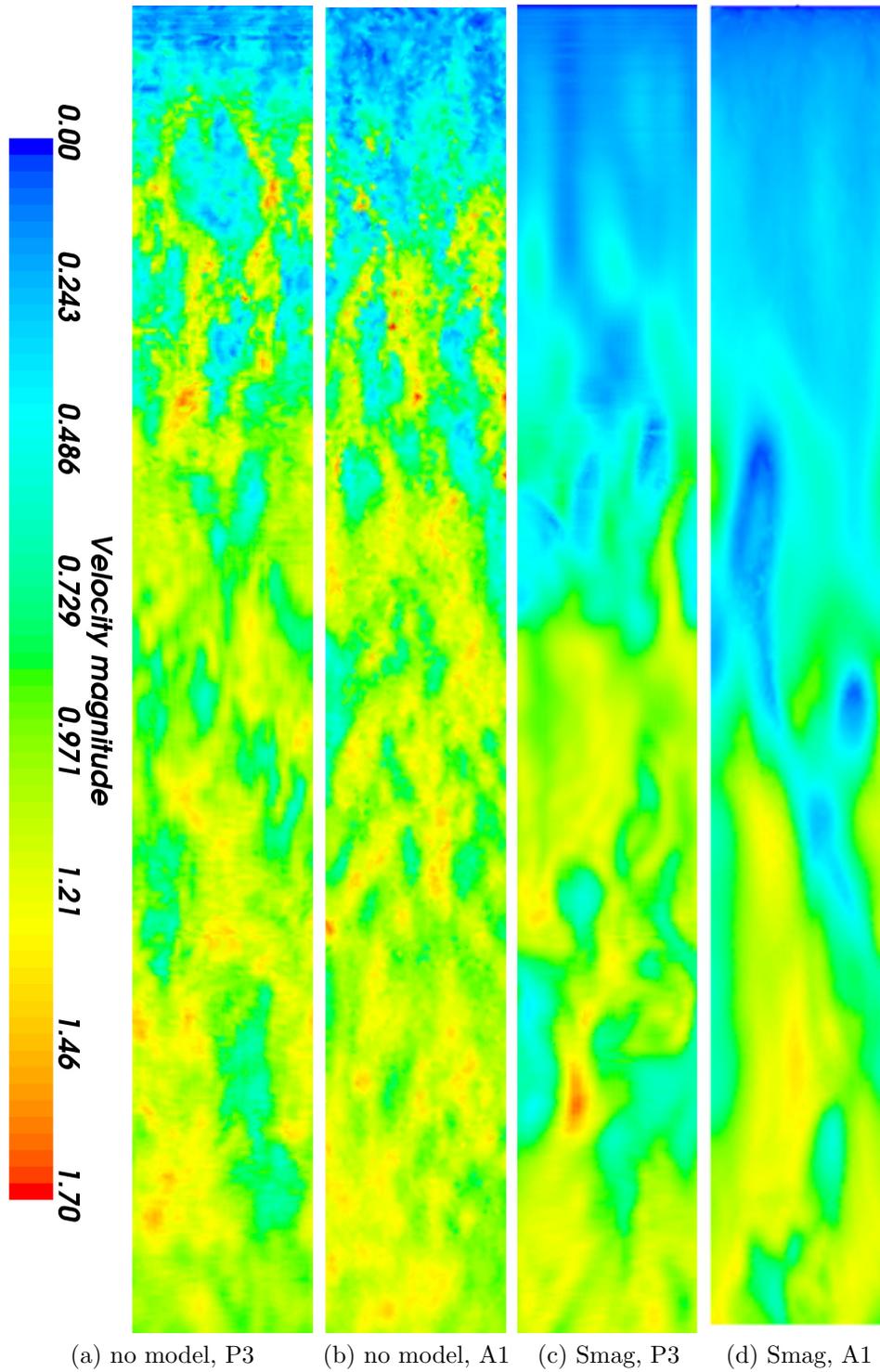


Figure 6.8: Velocity magnitude $|\bar{\mathbf{u}}|$ on plane A for no-model and tensor Smagorinsky LES results on P3 and A1 meshes.

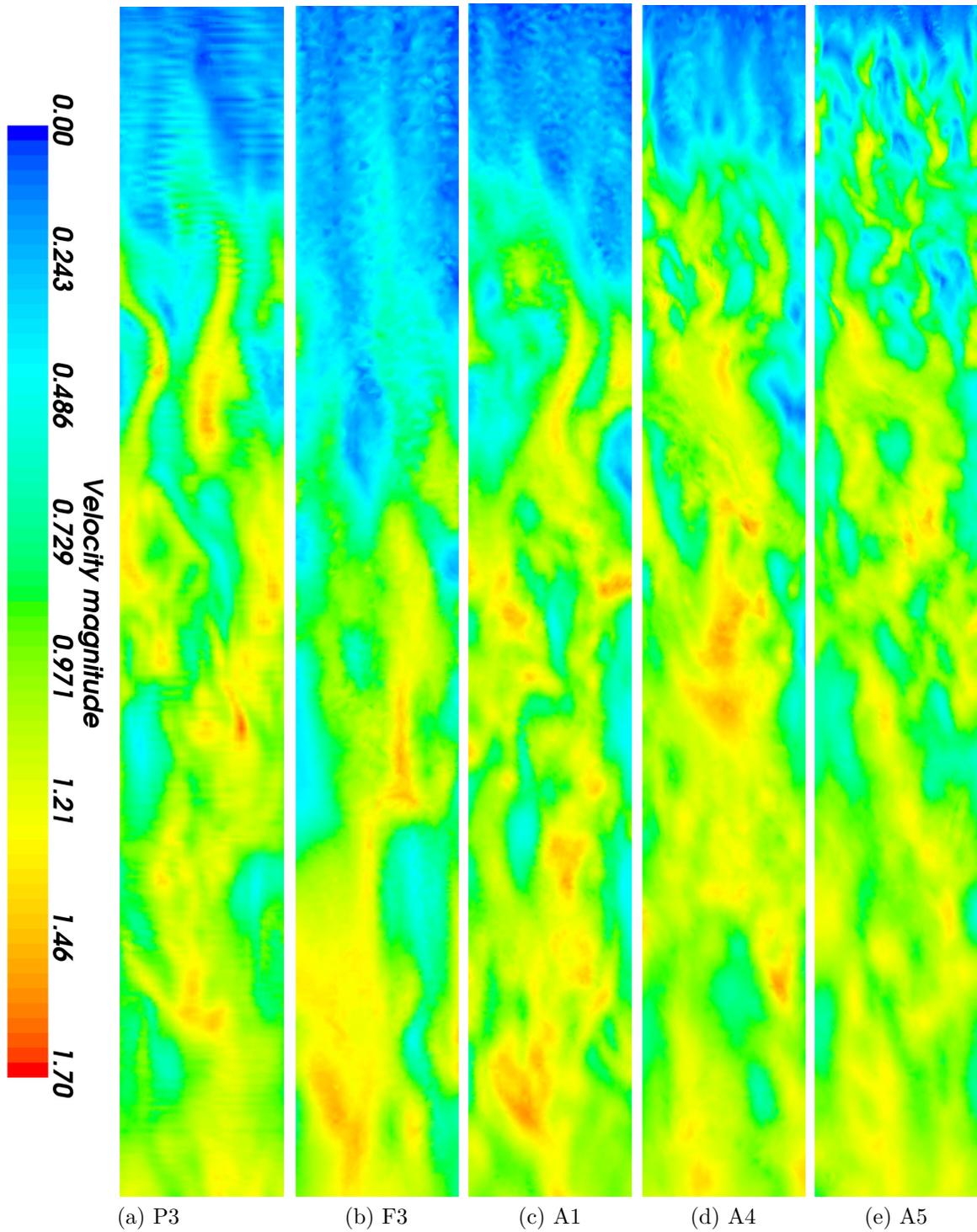


Figure 6.9: $|\bar{\mathbf{u}}|$ on plane A for tensor dynamic LES with isotropic filter on fixed (P3, F3) and adaptive (A1, A4, A5) meshes.

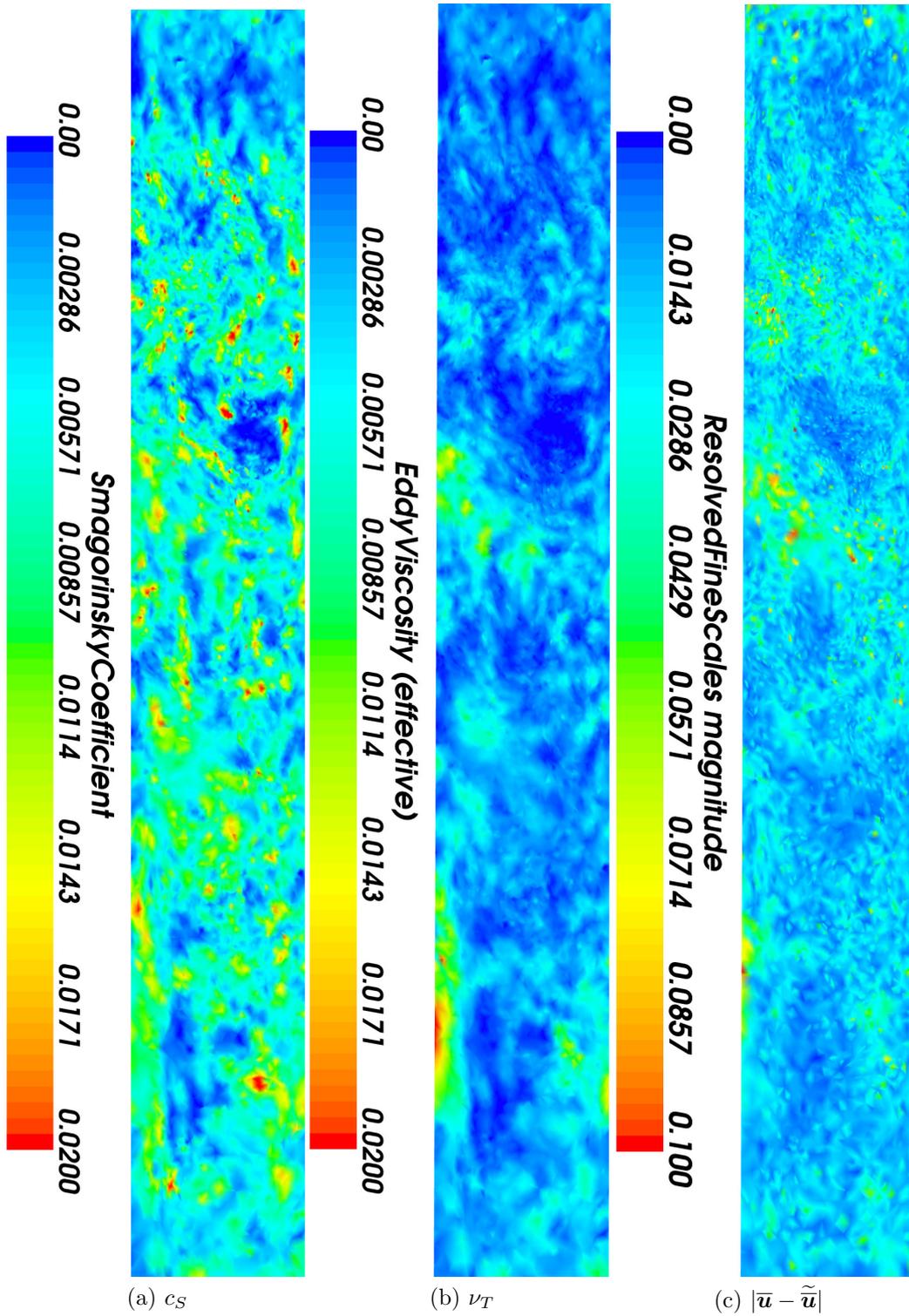


Figure 6.10: Dynamic Smagorinsky coefficient c_S , eddy viscosity ν_T and resolved fine scales $|\bar{\mathbf{u}} - \tilde{\mathbf{u}}|$ on plane A through mesh A4.

lates quite closely to $|\overline{\mathbf{u}} - \widetilde{\mathbf{u}}|$, confirming that the dynamic model bases the SFS on the highest resolved wavenumbers, as they should be according to Kolmogorov's hypothesis (cf. §3.2.2). The dynamic Smagorinsky coefficient (Figure 6.10a) displays a mottled pattern with rapidly alternating high and low values. This suggests that the standard Smagorinsky model, with its single value of C_S , is an inadequate description of the locally varying rate of energy flux from the resolved scales to SFS. Furthermore, larger values are very localised which is indicative of stable model behaviour.

6.3.3 Reattachment Length

Table 6.4 lists the time-averaged reattachment length ($R_{L,av}$) predictions from several meshes, LES models and LES settings. The best results were obtained with the dynamic LES model, adaptive meshing and $\alpha = 1$. The dynamic LES model on mesh P3 with no SEM applied at the inflow predicted slightly longer $R_{L,av}$ than with SEM. Using no LES model led to under-prediction of $R_{L,av}$ on P3 and A1 meshes, suggesting that the CG discretisation with fourth-order pressure stabilisation is acting like an LES model in its own right. Further research is needed to understand this behaviour. The VLES model with SEM on mesh P3 led to slight under-prediction of $R_{L,av}$. The tensor Smagorinsky model (5.22) of Bentham (2003) performed poorly on mesh A1 but matched dynamic LES on mesh P3, perhaps because it cannot compensate for the varying resolution like the dynamic method.

Explicit first filtering made no difference to reattachment length despite its improvements to mean velocity profiles. Removing the upper limit on c_S was variable in its impact: it made no difference to the isotropic results, while the anisotropic results were considerably worsened, suggesting that the coefficient was generally larger in the latter case.

Changing the adaptivity parameters in mesh A4 made little difference, while increasing the maximum number of nodes in mesh A5 improved the isotropic result but did not dramatically improve the anisotropic result. This may be because these simulations did not run for as long as the A1 runs (110 seconds compared to > 300 seconds for runs on mesh A1, owing to the increased number of nodes) and therefore the time-averaged quantities had not reached steady state.

The CFX dynamic LES R_L result is extremely large compared to the closest comparable Fluidity result (dynamic LES model on mesh P3 with no SEM), suggesting that not enough dissipation was generated. Indeed, running an LES without inlet fluctuations to seed turbulence is akin to a laminar flow, in which a very long R_L is expected. It highlights the need for SEM or a similar method of inflow turbulence generation in CFX. Nevertheless, the

6.3 Results

Model		Mesh	$R_{L,av}$
Experiment (Jovic et al., 1994)		-	6.0-6.1
DNS (Le et al., 1997)		structured, 9.4M nodes	6.28
Dynamic LES (Panjwani et al., 2009)		structured, 213k nodes	7.3
CFX dynamic LES		similar to P3	19.65
isotropic no SEM		P3	8.46
VLES		P3	5.47
no model		P3	5.81
Smagorinsky		P3	7.94
isotropic		P3	7.94
anisotropic		P3	7.77
no model		A1	6.02
Smagorinsky		A1	13.11
isotropic		A1	7.74
anisotropic		A1	7.90
isotropic		A4	7.39
anisotropic		A4	7.87
isotropic		A5	6.04
anisotropic		A5	7.53
isotropic	$\alpha = 1, \beta = 2$	A1	6.65
anisotropic	$\alpha = 1, \beta = 2$	A1	6.68
isotropic	$\alpha = 1, \beta = 1$	A1	6.51
anisotropic	$\alpha = 1, \beta = 1$	A1	7.04
isotropic	explicit f.f.	A1	7.52
anisotropic	explicit f.f.	A1	8.20
isotropic	$\alpha = 1$, explicit f.f.	A1	6.58
anisotropic	$\alpha = 1$, explicit f.f.	A1	6.75
isotropic	unlimited c_S	A1	7.27
anisotropic	unlimited c_S	A1	12.93
isotropic	$\alpha = 1$, unlimited c_S	A1	6.55
anisotropic	$\alpha = 1$, unlimited c_S	A1	9.05

Table 6.4: Time-averaged reattachment length $R_{L,av}/h$ in the 3D backward-facing step. ‘No model’ = simulations without an LES model. ‘Smagorinsky’ = tensor eddy viscosity Smagorinsky model (5.22). All other entries are dynamic LES. ‘explicit f.f.’ = explicit first filter. $\alpha = 2, \beta = 2, c_S \leq 0.04$, implicit first filter and SEM inflow unless otherwise stated.

Fluidity R_L result with no SEM is surprisingly good; deficiencies are only obvious when the mean velocity profiles are observed (Figure 6.19, page 225).

6.3.4 Mean Velocity Profiles

6.3.4.1 Method

In accordance with [Le et al. \(1997\)](#) and [Panjwani et al. \(2009\)](#), the mean velocity profiles are plotted at locations downstream of the step which are normalised with respect to the average reattachment length for each simulation. The locations are $R_{L,av} \times [\frac{2}{3}, 1, \frac{5}{3}, \frac{5}{2}]$. This procedure is intended to provide a fairer comparison of the shapes of the velocity profiles through the separation and recovery regions. R_L was calculated by the spanwise averaging procedure described above and profiles were extracted from the simulation results using the same ‘point probe and interpolation’ method as the reattachment length calculation described above.

The results of dynamic LES simulations are compared against no-LES-model results and the dynamic LES results of [Panjwani et al. \(2009\)](#) and DNS of [Le et al. \(1997\)](#). The effect of changing the filter type (isotropic/anisotropic), filter width ratios α and β , and limiting c_S are investigated. CFX results and Fluidity results with no SEM applied are presented in §6.3.7. All results presented used the values $c_S \leq 0.04$ and $\beta = 2$ with an implicit first filter unless otherwise stated.

6.3.4.2 Filter Type and Width Ratios

Figure 6.11 shows the mean velocity profiles obtained with adaptive meshing and the isotropic filter, and Figure 6.12 with the anisotropic filter. All results used an implicit first filter. The anisotropic model with $\alpha = 2$ is more dissipative than the isotropic model with $\alpha = 2$, indicated by the more rounded profile and significant lag behind the DNS in Figure 6.12. The stepped line in Figure 6.12 is due to coarse elements in the mesh; that these elements were generated close to the wall indicates that mesh A1 is not optimal for this flow.

Setting $\alpha = 1$ greatly improves the predictions of both the isotropic and anisotropic models, giving approximately identical results to each other, suggesting that the filter width is more important than the filter shape in determining model dissipation. Changing β made comparatively little difference. One reason for the observed insensitivity to the filter shape could be that although the mesh is adaptive, the elements in it are still fairly isotropic.

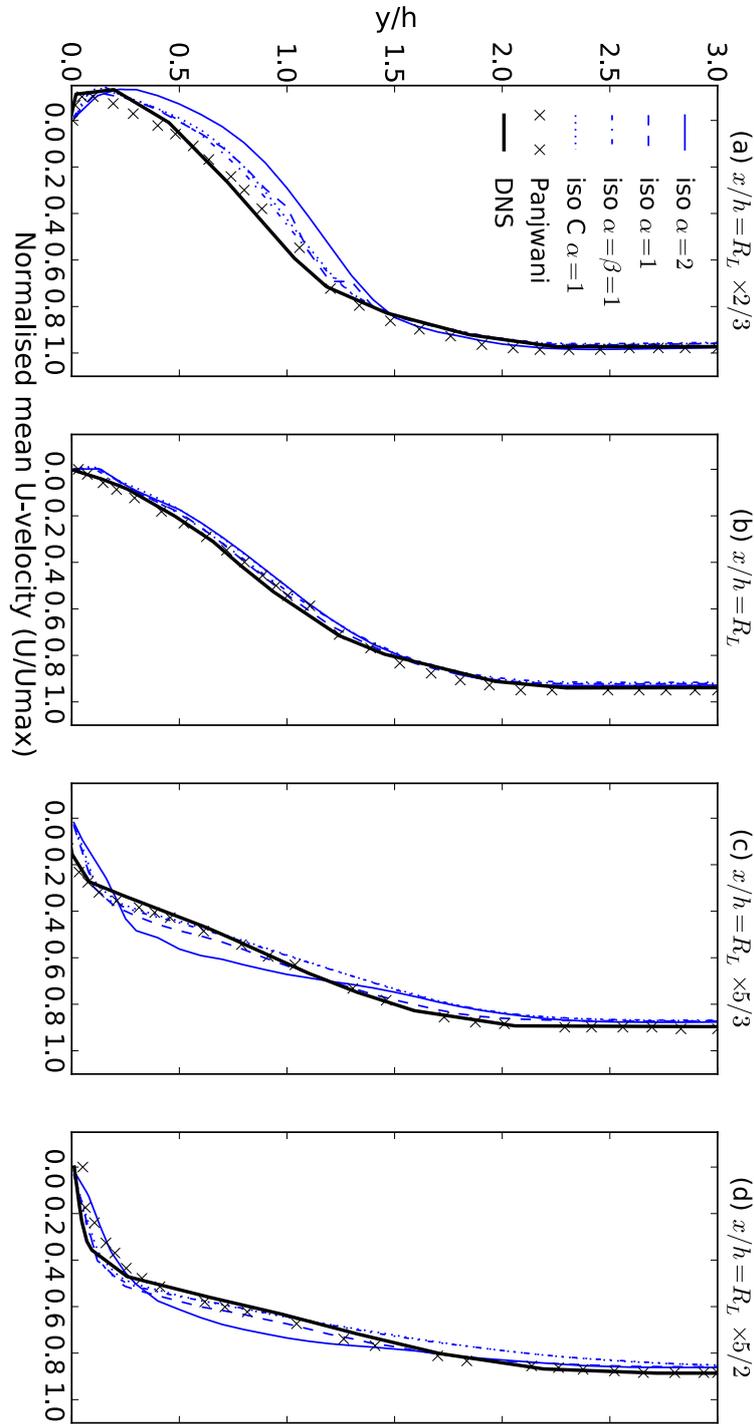


Figure 6.11: Mean velocity profiles in 3D backward-facing step using isotropic filter on mesh A1. ‘DNS’ = DNS of Le et al. (1997); ‘Panjwani’ = dynamic LES of Panjwani et al. (2009); ‘iso’ = isotropic dynamic model; ‘C’ = no upper limit on c_S . $c_S \leq 0.04$, $\beta = 2$ unless otherwise stated.

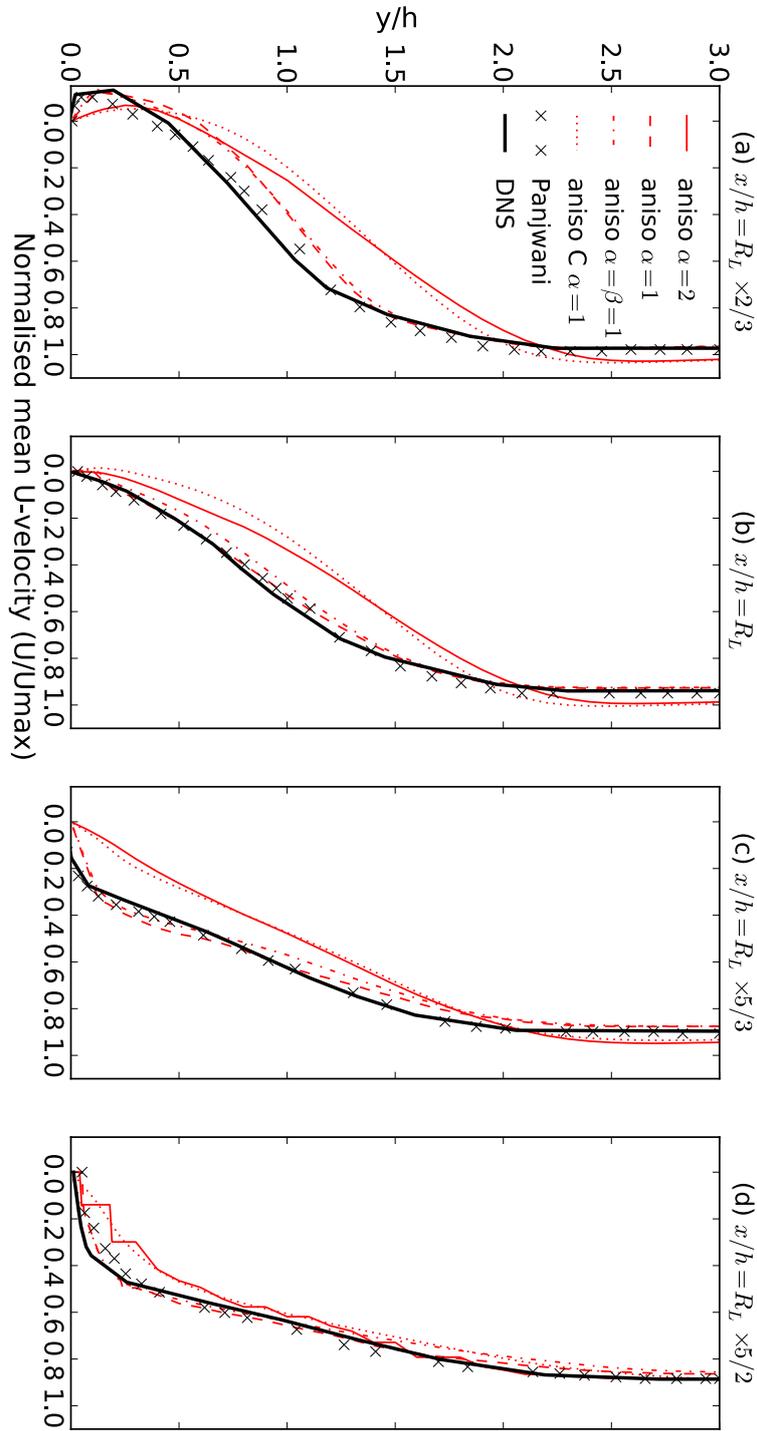


Figure 6.12: Mean velocity profiles in 3D backward-facing step using anisotropic filter on mesh A1. ‘aniso’ = anisotropic dynamic model. Stepped line in (d) is due to coarse elements in adaptive mesh. $c_S \leq 0.04$, $\beta = 2$, implicit first filter unless otherwise stated.

6.3.4.3 Limiting c_S

Figures 6.11 and 6.12 also show results in which the upper limit on c_S has been removed. In the isotropic case this makes no difference compared to the equivalent limited result, but in the anisotropic case the profiles are worsened: the unlimited run with $\alpha = 1$ is comparable to the limited run with $\alpha = 2$, offsetting the gain achieved by changing α . These results imply that the isotropic model is more robust as it does not require limiting of the coefficient to make it stable.

Plots of c_S show that the anisotropic model is inclined to make the coefficient, and therefore the eddy viscosity, take on large values over more of the domain than the isotropic model, and limiting is required over a greater portion of the domain. The likely reason is that the isotropic filter width is smaller than the anisotropic one for a particular element. Although the anisotropic width has already been reduced by the normalisation procedure (5.45), this may not have been the correct way to do so, and it may need to be further revised (cf. §8.3.1).

6.3.4.4 Fixed vs. Adaptive Mesh

Figure 6.13 shows results from periodic mesh P3 and adaptive meshes A1, A4 and A5 with $\alpha = \beta = 2$. The P3 profiles are comparable to the A1 profiles, indicating that the poorer R_L predictions (cf. Table 6.4) on the fixed mesh are probably due to insufficient wall resolution, and that adaptive meshing with far fewer elements can capture the flow profile. Mesh A4 captures the profiles most accurately, while mesh A5 is the poorest, but the simulation in this case was shorter and the averaging period consequently too short to obtain meaningful profiles. This problem does not affect the reattachment length because it was computed from instantaneous velocity and it was verified that statistical steady state had been reached by the end of the run.

6.3.4.5 Implicit vs. Explicit First Filter

Figures 6.14 and 6.15 compare results from simulations using an implicit first filter i.e. the ‘filtered’ velocity $\bar{\mathbf{u}}$ is simply the discrete velocity on the computational mesh, and an explicit first filter i.e. $\bar{\mathbf{u}}$ is obtained from the discrete velocity on the computational mesh by an explicit application of the inverse Helmholtz filter. The first filter width is the same in both cases. Details of these procedures are in §5.2.2.5.

Adaptive mesh A1 was used and both isotropic and anisotropic filters were tested with both $\alpha = 1$ and $\alpha = 2$. Explicit first filtering greatly improves the mean profiles; the effect is most pronounced with the anisotropic filter.

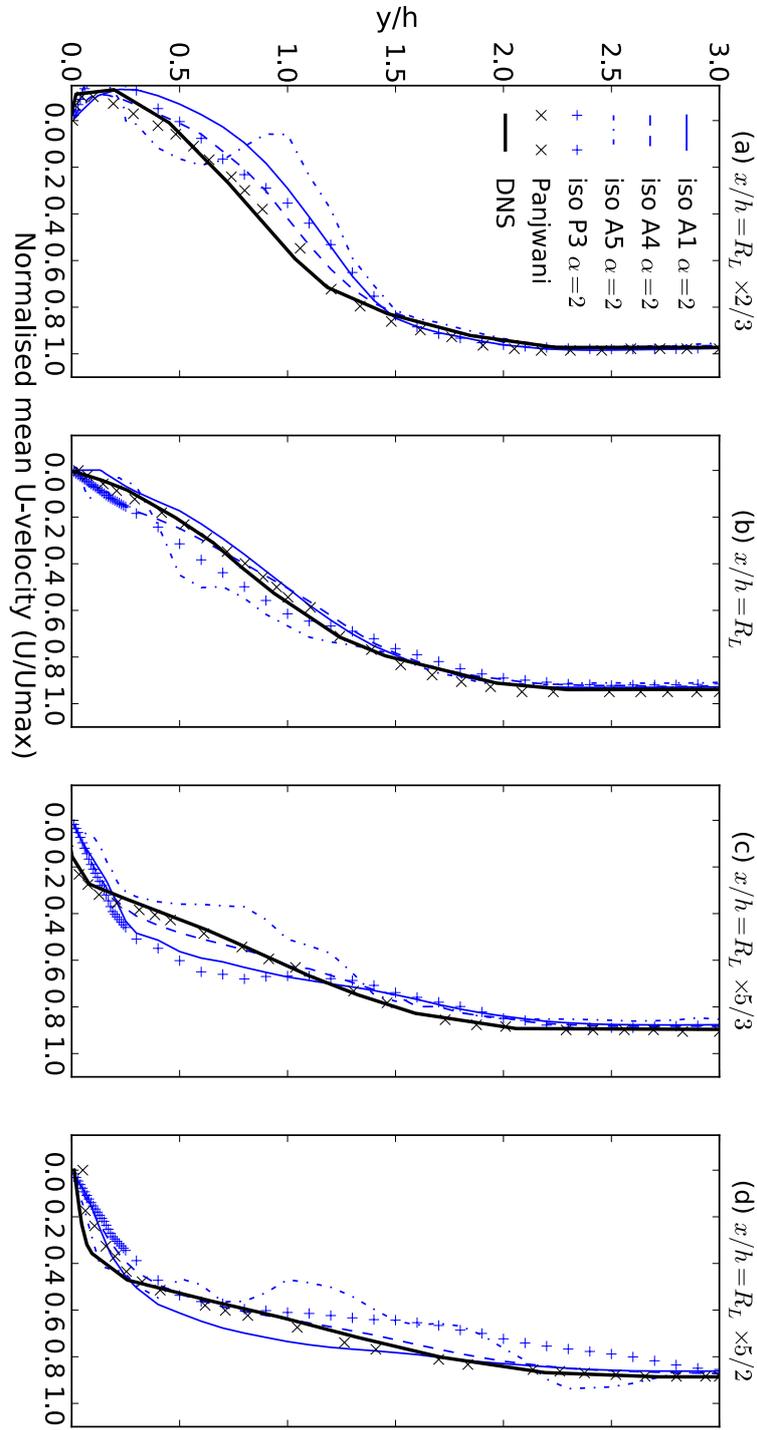


Figure 6.13: Mean velocity profiles in 3D backward-facing step using isotropic filter on adaptive meshes A1, A4 and A5 and periodic mesh P3. $c_S \leq 0.04$, $\beta = 2$, implicit first filter.

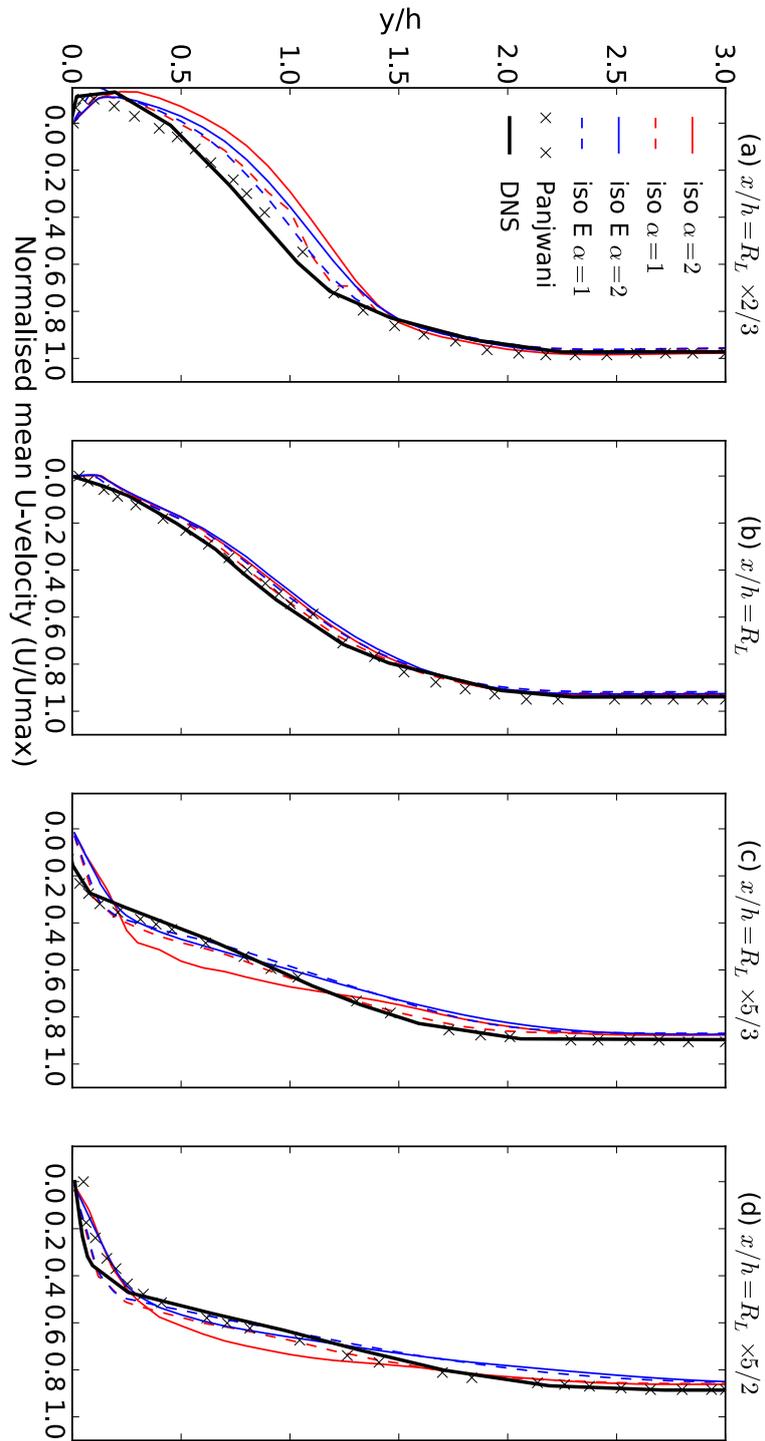


Figure 6.14: Mean velocity profiles in 3D backward-facing step using isotropic implicit/explicit first filter on mesh A1. ‘E’ = explicit first filter. $c_S \leq 0.04$, $\beta = 2$, implicit first filter unless otherwise stated.

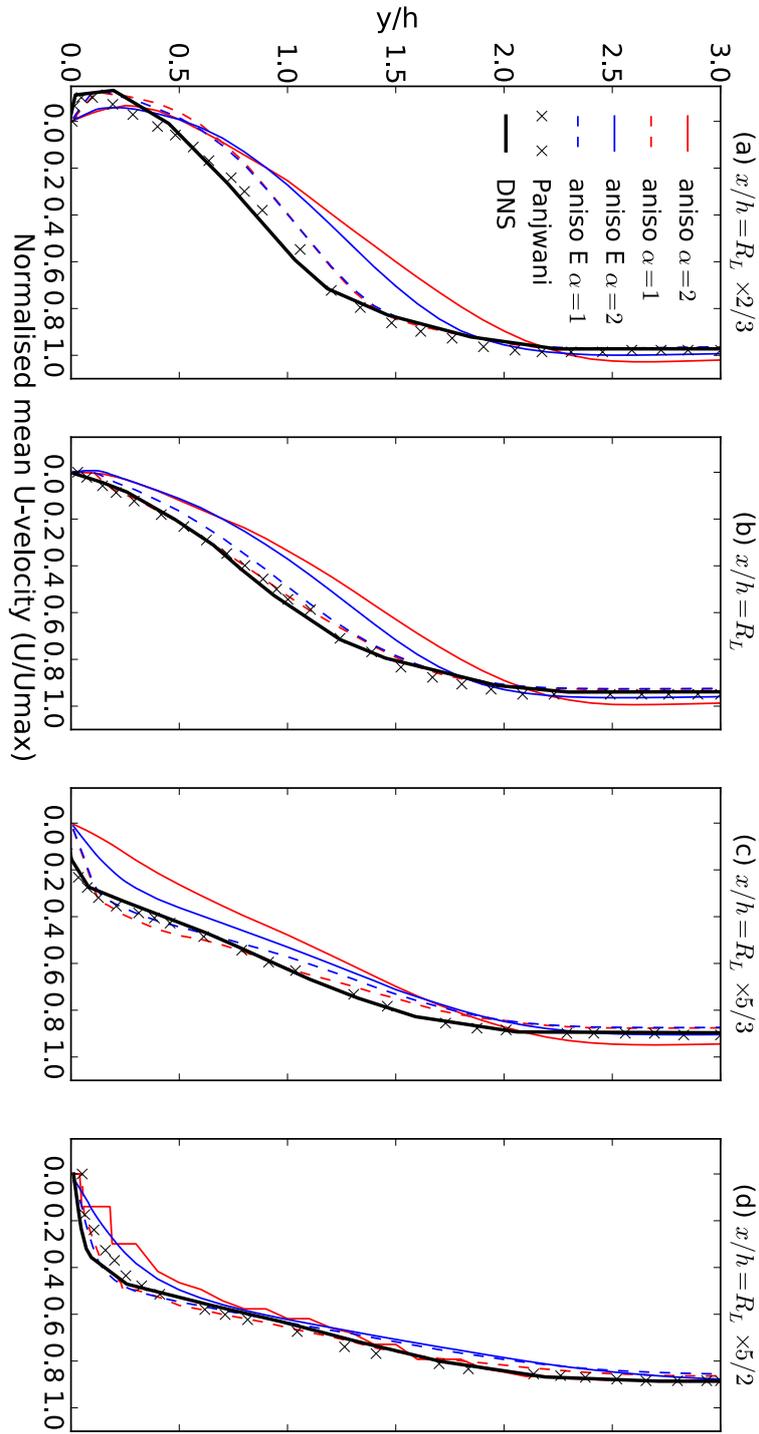


Figure 6.15: Mean velocity profiles in 3D backward-facing step using anisotropic implicit/explicit first filter on mesh A1. ‘E’ = explicit first filter. $c_S \leq 0.04$, $\beta = 2$, implicit first filter unless otherwise stated.

Changing α has less effect on the explicit filter results than the implicit. These results suggest that maintaining similarity between the first and second (test) filters (here by using the inverse Helmholtz filter to define both $\bar{\mathbf{u}}$ and $\tilde{\mathbf{u}}$) is important, perhaps for the reasons given in §5.2.2.5.

6.3.5 Reynolds Stresses

Time-averaged Reynolds stresses were plotted on a vertical line on plane B at a distance of $x = \frac{2}{3}R_{L,av}$ downstream of the step. Figures 6.16a and 6.16b show the Reynolds stresses from the adaptive mesh A1 for the same cases shown in Figures 6.11 and 6.12. The plots show some lack of smoothness which is due to the relatively short time averaging period of around 7 – 10 flow-through periods. The gradient of $|\bar{\mathbf{u}}'|/u_0$ is poorly captured by LES near the wall and would benefit from reduced element size at the wall.

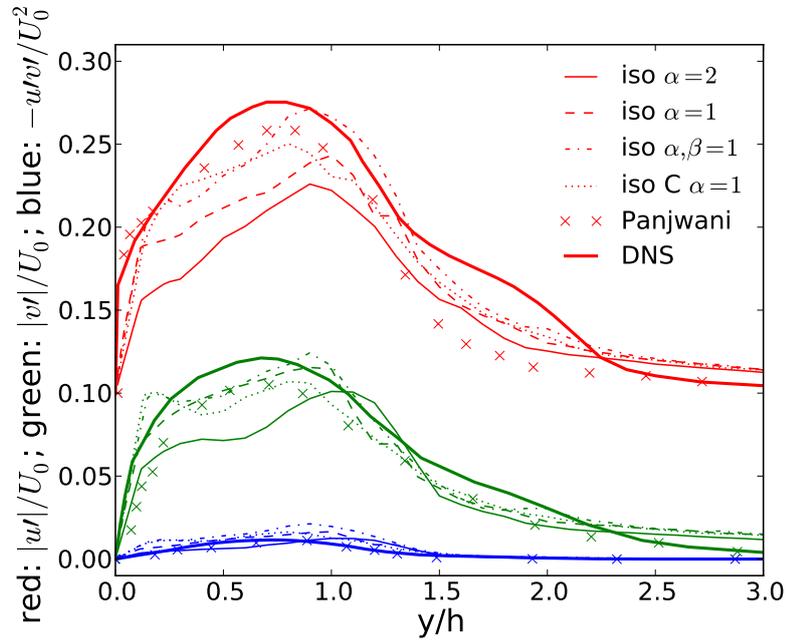
As with the mean velocity, setting $\alpha = 1$ improves Reynolds stress profiles. The best fit to DNS data is obtained with $\alpha = \beta = 1$ with either the isotropic or anisotropic filter; both these results are comparable to the dynamic LES of Panjwani et al. (2009). The worst result is obtained with $\alpha = \beta = 2$ and the anisotropic filter, consistent with the mean velocity results. Reduction of α and β corresponds to reducing the modelled dissipation, perhaps compensating for numerical dissipation.

Figure 6.17 compares the Reynolds stresses captured on three meshes: A1, A4 and P3, with the isotropic filter. The plots suffer from the same short averaging period problem as the previous figures, resulting in a noticeable deficit region between $z = 0.1$ and $z = 1$. The best results are those from mesh A4, demonstrating the superiority of the adaptivity settings for that mesh. A4 allows for smaller element size and as a result representation of the near-wall (up to $z = 0.1$) variation of Reynolds stresses is superior.

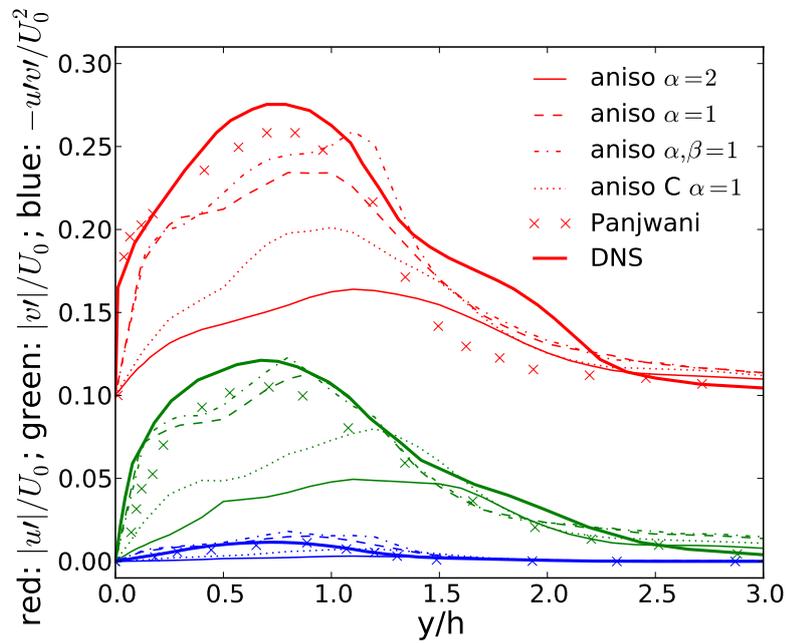
CFX results and Fluidity results with no SEM applied are presented in §6.3.7.

6.3.6 Near-Wall Profiles

Le et al. (1997) found that the velocity profile downstream of the step did not recover completely to the universal log law within 20 step heights downstream. The near-wall velocity profiles in wall units (y^+, u^+) are plotted against DNS and experimental data and the log law of the wall in Figure 6.18 at a distance of $x/h = 19$ downstream of the step, where the boundary layer profile is almost fully redeveloped after the disturbance of separation and reattachment of the shear layer. Dynamic LES on periodic mesh P3 and adaptive mesh A1 come reasonably close to the reference data, while on



(a) Isotropic filter



(b) Anisotropic filter

Figure 6.16: Reynolds stress profiles in 3D backward-facing step using isotropic and anisotropic filters on mesh A1. Red lines shifted by 0.1. $c_S \leq 0.04$, $\beta = 2$ unless otherwise stated.

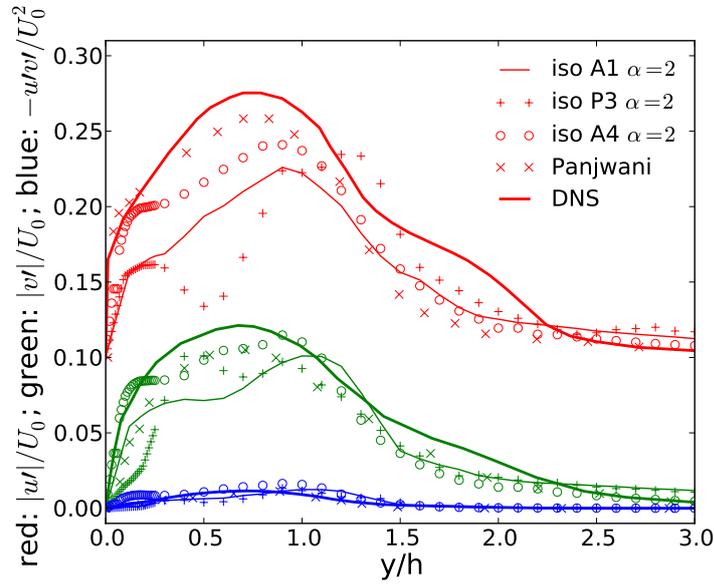


Figure 6.17: Reynolds stress profiles in 3D backward-facing step using isotropic filter on meshes P3, A1 and A4. Red lines shifted by 0.1. $c_S \leq 0.04$, $\beta = 2$.

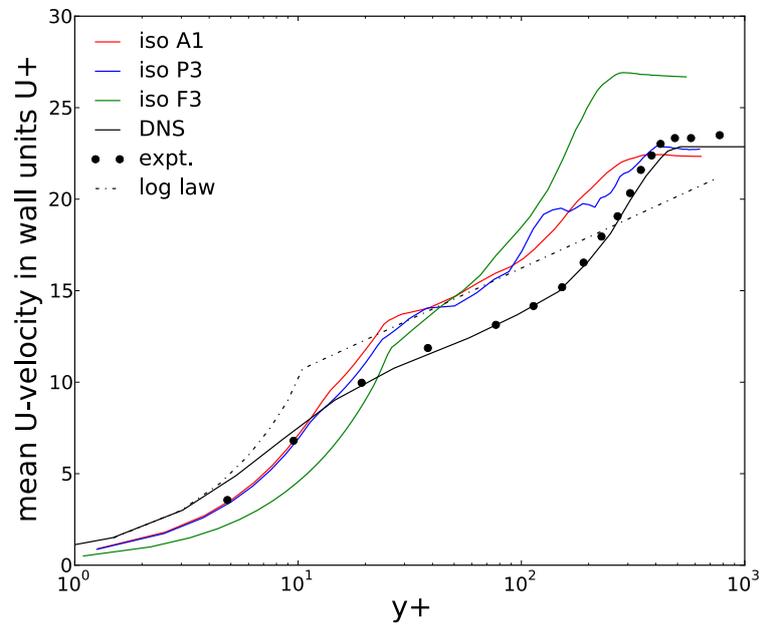


Figure 6.18: Velocity profiles in wall units at $x/h = 19$ in 3D backward-facing step. ‘expt.’ = experimental data of [Jovic and Driver \(1994\)](#); ‘log law’ = linear-log law of the wall (cf. [Table 3.1](#)). $c_S \leq 0.04$, $\alpha = \beta = 2$.

the fixed mesh F3 the profile is not as close. A region in which the log law is approximately satisfied (straight line) is visible in the A1 results between $20 \leq y^+ \leq 100$ which is roughly the range of the log-law region in the DNS and experimental data. The fluctuations in the P3 profile are due to a shorter averaging period for that run.

6.3.7 CFX Results

The closest setup in Fluidity to CFX was the isotropic dynamic LES model on mesh P3 with implicit first filtering and no SEM at the inlet. A spanwise-averaged reattachment length of 19.65 was found for the CFX dynamic LES simulation. Figure 6.19 shows the mean velocity profiles from CFX and the reference Fluidity results at four locations downstream of the step: $x/h = [4, 6, 10, 19]$. The CFX profiles are very poor compared to DNS indicating that the LES model has generated insufficient dissipation; the solution is more like laminar flow than turbulent as a result of the lack of inlet turbulence. The Fluidity profiles are comparable; the lack of inlet turbulence is clear compared to Figure 6.13. CFX does not contain a sophisticated turbulent inflow generation method like SEM, and so it is not possible to improve upon these results.

Figure 6.20 shows the Reynolds stresses at the same positions in CFX and the reference Fluidity results. In the CFX profiles, in the range $4 \leq x/h \leq 10$ there is a large deficit in $\overline{u'}$ (blue) at around $y/h = 0.7$ and the other components are not well predicted at any height. However, Reynolds stresses are in very good agreement with DNS at the final position (close to R_L), suggesting that the flow has finally developed a significant amount of turbulence. A similar trend is observed in the reference Fluidity profiles, strongly suggesting again that SEM contributes largely to the accuracy of the backward-facing step results by inducing earlier development of turbulence in the step wake.

6.3.8 VLES Results

The scale-adaptive VLES model described in Chapter 4 has also been validated in the 3D backward-facing step. Any turbulence model that resolves transient multiscale dynamics can only truly be validated in 3D, since turbulence is an inherently three-dimensional phenomenon. The simulation presented was on the P3 mesh with SEM at the inflow. High-Re boundary conditions were specified on the bottom for k and ε and the weak log law for velocity.

6.3 Results

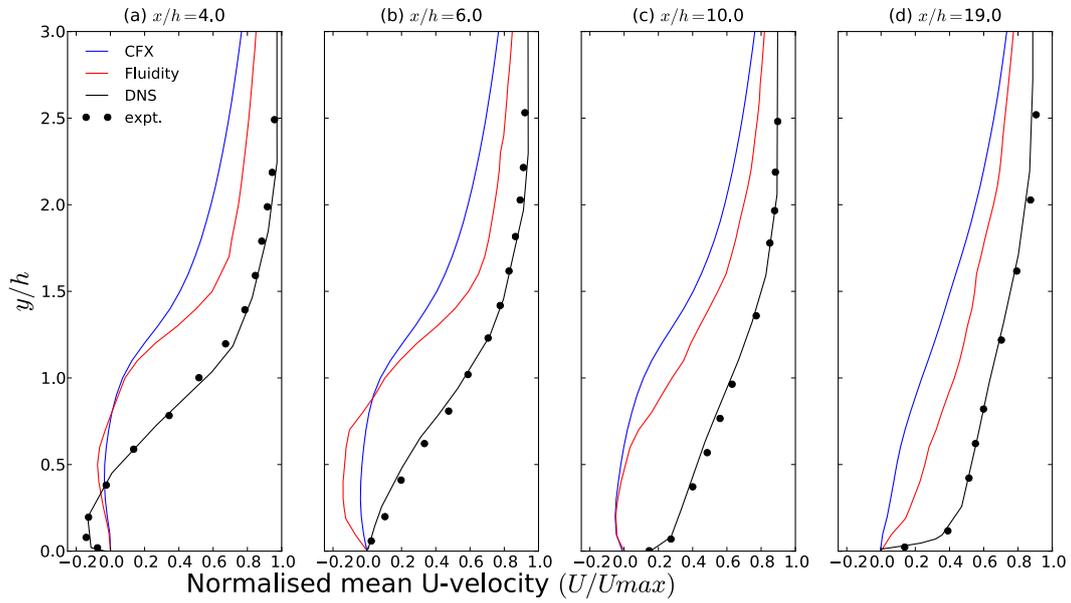


Figure 6.19: Mean velocity profiles in CFX simulation of 3D backward-facing step vs. Fluidity dynamic LES with no SEM on P3 mesh, DNS and experimental data.

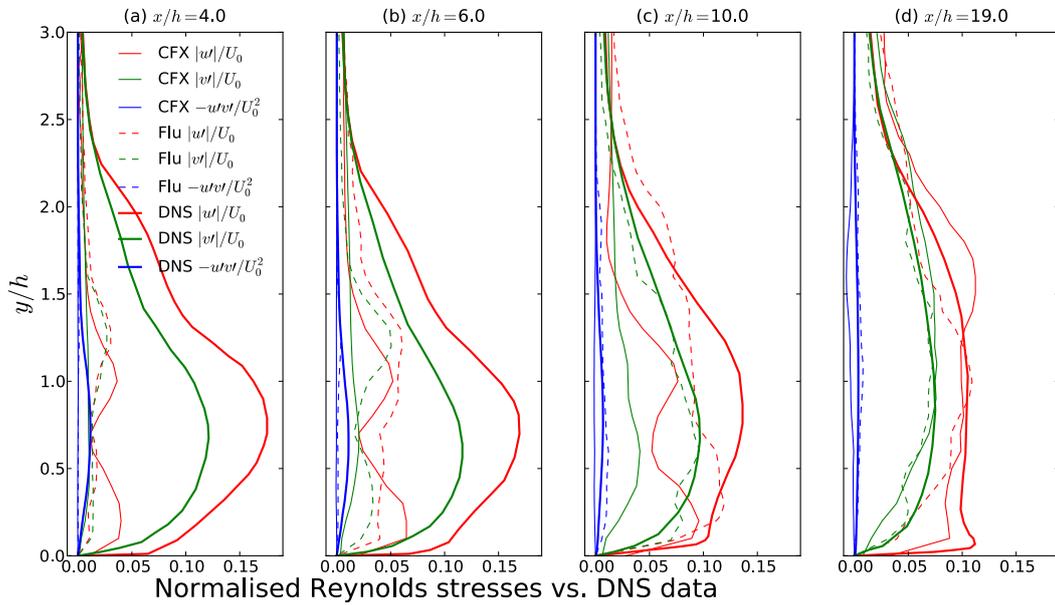


Figure 6.20: Normalised Reynolds stress profiles in CFX simulation of 3D backward-facing step vs. Fluidity dynamic LES with no SEM on P3 mesh and DNS data. ‘Flu’ = Fluidity.

Figure 6.21 shows the velocity, eddy viscosity and VLES damping function on plane B. A fully turbulent velocity field is visible in Figure 6.21a, enhanced by the turbulent inflow conditions. A turbulent separation eddy and reattachment zone can be seen. The maximum eddy viscosity (Figure 6.21b) is far smaller than typical values observed in dynamic LES results. Its outline demarcates the spreading turbulent shear layer formed downstream of the separation point. This is even more starkly visible in the filter distribution (Figure 6.21c): a sharp division separates the turbulent region with very small values of the function from an outer layer where the value is one. The outer layer represents a RANS solution and the turbulent region an LES-like solution. A similar division can be seen very close to the bottom although the RANS region is intermittent. The reasons for this intermittency are as yet unknown.

Figure 6.22 shows the velocity on plane A. The velocity distribution is comparable to the dynamic LES and no-model velocity distributions in Figures 6.9a and 6.8a respectively. The VLES model appears to be functioning like an LES model, resolving the formation of fine scales near the step which grow downstream.

Comparisons of the mean velocity and Reynolds stress profiles to the DNS and experimental data are shown in Figures 6.23 and 6.24. Mean velocity is very well captured everywhere except the near-wall region, possibly because there is a thin, intermittent RANS layer in which the dynamics are not resolved. Reynolds stresses follow a similar pattern, being very close to the benchmark data above $y/h = 0.3$ but unrealistic nearer the bottom. A reattachment length of 5.47 was calculated, lower than the benchmark data and dynamic LES results. The low reattachment length is probably a result of the inaccurate representation of velocity profiles seen in Figure 6.23. Simulations where flow must be accurately represented near walls may present a problem for VLES modelling approaches.

6.4 Discussion

A novel tensor dynamic LES model was implemented and tested in the 3D backward facing step. Predicted reattachment length came within 4% of benchmark DNS data using the isotropic filter with $\alpha = \beta = 1$, an implicit first filter, adaptive meshing and the synthetic eddy method (SEM). Using the anisotropic filter with $\alpha = 1, \beta = 2$, an implicit first filter and adaptive meshing, the reattachment length came within 7%. A comparable dynamic LES simulation was carried out by Panjwani et al. (2009), finding a reattachment length within 14% of DNS using a 213000-node structured mesh. Mean

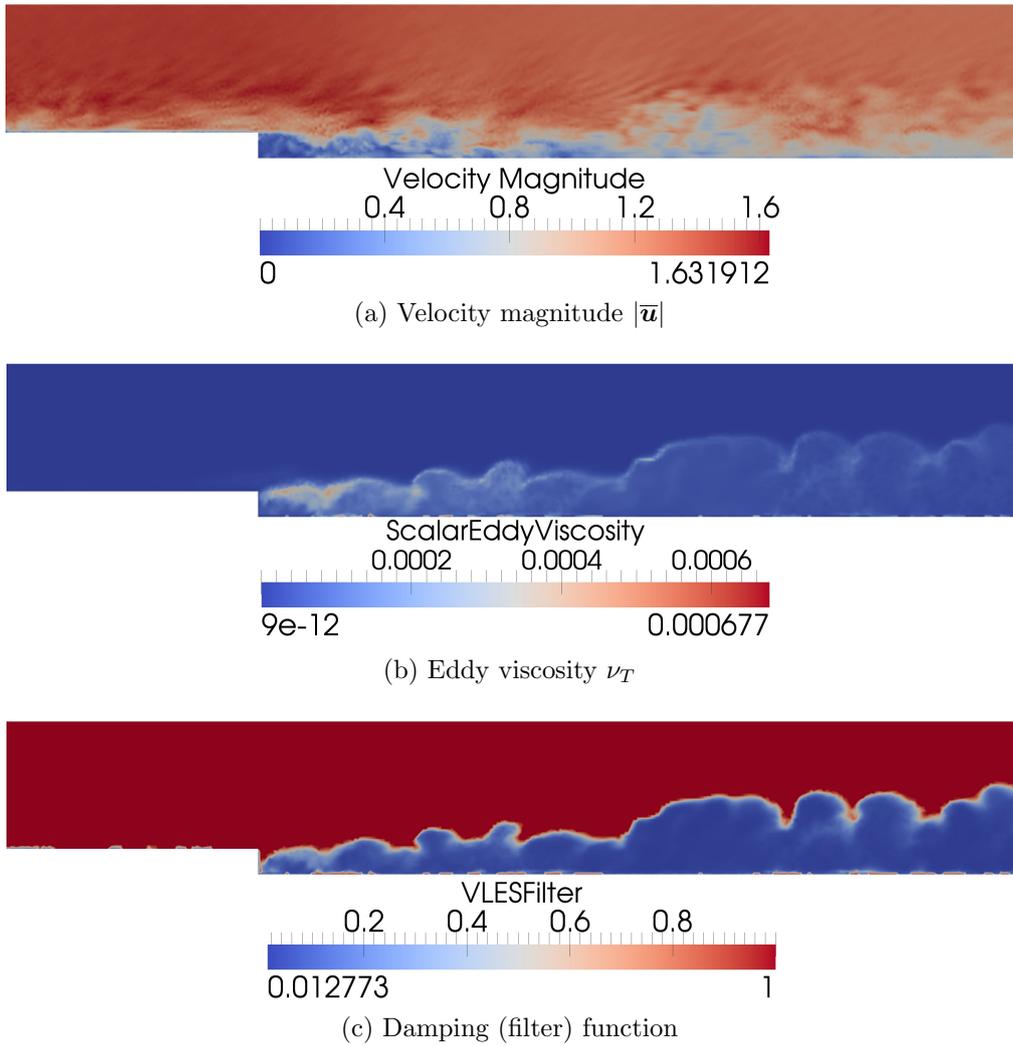


Figure 6.21: VLES model velocity, eddy viscosity and damping (filter) function on plane B of 3D backward-facing step.

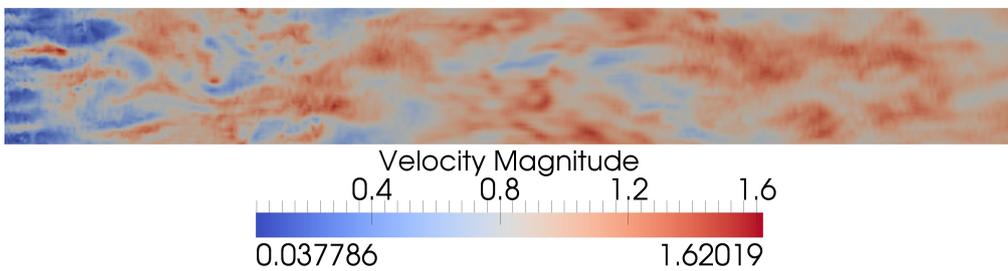


Figure 6.22: VLES model velocity magnitude $|\bar{u}|$ on plane A. Step lip is on the left, outlet on the right.

6.4 Discussion

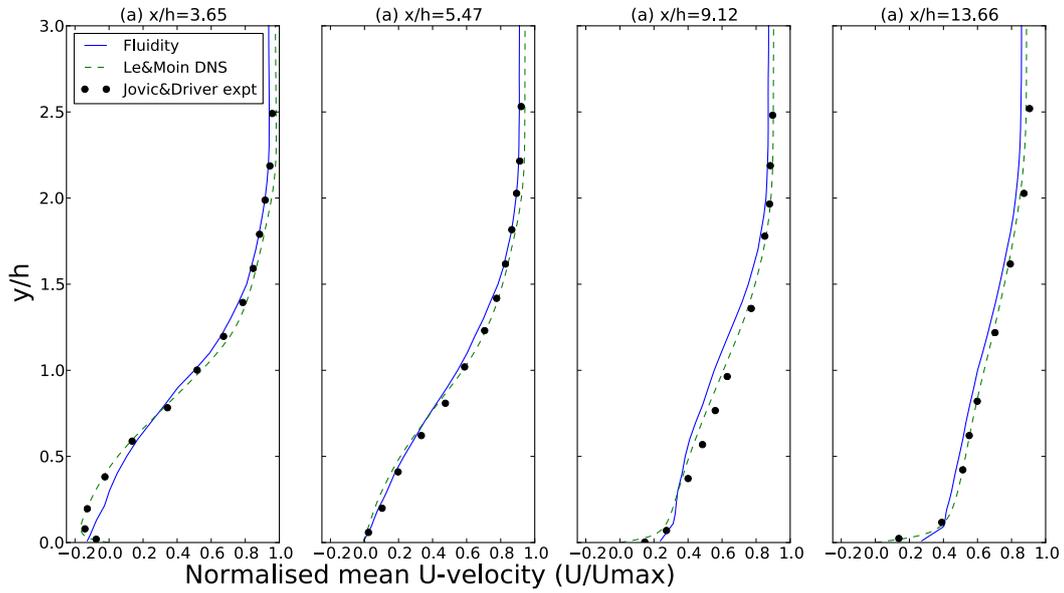


Figure 6.23: Mean velocity profiles in VLES simulation of 3D backward-facing step with SEM on P3 mesh vs. DNS and experimental data. Solid: VLES, dashed: DNS, dots: experiment.

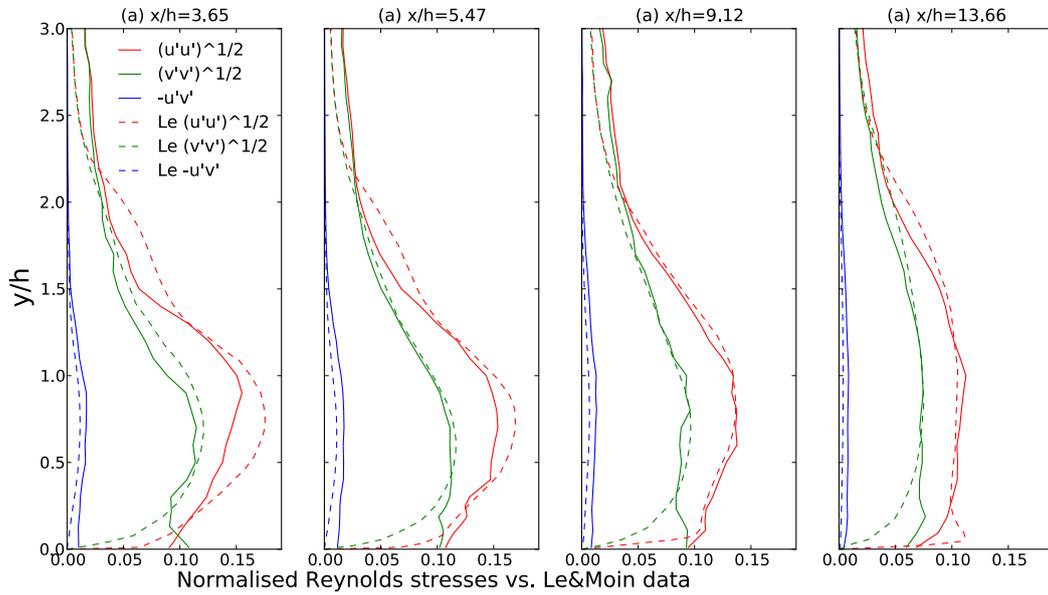


Figure 6.24: Normalised Reynolds stress profiles in VLES simulation of 3D backward-facing step with SEM on P3 mesh vs. DNS data. Solid : VLES, dashed: DNS.

velocity and Reynolds stress profiles were captured accurately in comparison to benchmark data and Panjwani et al. (2009). An almost identical setup in CFX returned an almost laminar solution, and subsequently very large R_L , as a result of the lack of a suitable turbulent inflow generation method. It was shown that the use of SEM for this purpose in Fluidity leads to vastly improved accuracy. The particular choice of LES model then affects how that inlet turbulence evolves and dissipates.

6.4.1 LES and Adaptivity

The adaptive results were better than those from the fixed periodic mesh indicating that a fine near-wall mesh was important to resolve the dynamics there. The combination of dynamic LES with unstructured mesh adaptivity improved upon the fixed-mesh results while using 60% fewer nodes (comparing meshes A1 and P3), although this was partly because of superior near-wall resolution in the adaptive case. Adaptivity, properly calibrated, automatically refined the mesh in the near-wall region to provide the resolution required. The computational overhead of adapting the mesh was negligible in comparison, since it was performed only once every 100-200 timesteps at a cost of about one timestep per adapt.

The effect of different choices of adaptivity target has not been isolated. Velocity $\bar{\mathbf{u}}$ was used as the target for the A1 mesh, while a combination of the velocity fluctuation (or Reynolds stress) $\bar{\mathbf{u}} - \langle \bar{\mathbf{u}} \rangle$ and the time-averaged velocity $\langle \bar{\mathbf{u}} \rangle$ was used for the A4 and A5 meshes. Results on these latter meshes vary depending on which diagnostic we look at, with no improvement in R_L predictions over A1 except for A5 with the isotropic filter.

Mean velocity profiles and Reynolds stresses, however, were improved by using A4 instead of A1. Resolution of turbulent structures by A4 is excellent compared to A1, so the disappointing results may be down to not adapting frequently enough to follow the flow at all times. Results files were only written at the same timestep as the mesh was adapted, so the images (e.g. Figure 6.5) do not show whether the mesh gets ‘left behind’ by the flow at intermediate times. It is plausible that the frequency of mesh adapts needs to be increased significantly from every 200 timesteps in order to faithfully follow the flow details throughout the simulation. Mesh A5 did not provide conclusive results owing to the shorter simulation times.

6.4.2 Tensor Dynamic LES

The main positive effect on model accuracy was from setting the filter width parameter $\alpha = 1$, whereas the shape i.e. scalar vs. tensor, made less differ-

ence. In general, the anisotropic filter led to over-predictions of the reattachment length and under-predictions of mean velocity and Reynolds stresses, indicating that it is overly dissipative. The reason for this may be that the normalisation of anisotropic filter width by (5.45) is inappropriate. Violation of frame invariance by the tensor model, explained in §5.2.1, may also be responsible for the lack of improvement over the scalar model.

Using an explicit first filter to satisfy the first condition of scale-similarity, that is, the first and second filters having similar shapes, improved the mean profile and reattachment length. This may contradict the finding of [Tejada-Martinez and Jansen \(2003\)](#), namely that using similar mesh and test filters is unnecessary.

With $\alpha = 1$ or 2 , removal of the upper limit on the dynamic Smagorinsky coefficient did not significantly change the isotropic model results, indicating that the isotropic dynamic LES model developed in this work is more stable than the method as [Lilly \(1992\)](#) conceived it, which required either local averaging or clipping the lower and upper values of c_S to maintain stability (see §5.2.2): in the current method only the lower limit $c_S \geq 0$ has to be imposed. However, removing the upper limit made the anisotropic model results slightly worse with either $\alpha = 1$ or 2 .

It has not been verified whether Kolmogorov's local isotropy hypothesis was confirmed by these simulations, that is, whether the scales at cut-off were isotropic. The Smagorinsky model assumes that they are, but the use of a tensorial anisotropic eddy viscosity avoids this assumption. Use of mesh adaptivity with the tensor dynamic LES model may have satisfied scale similarity because anisotropy of the smallest resolved scales was prescribed by the anisotropy of the mesh. This anisotropy was transferred to the SFS via the tensor filter width definition.

However, whether the mesh anisotropy was a true reflection of the anisotropy of the real flow is difficult to answer. More work is needed to test different adaptivity settings and target fields to find the optimal setup for this and other LES problems. Furthermore, to truly represent the anisotropy of the small scales in the mesh, the mesh would have to be adapted every timestep, which is not only impractical from a computational effort point of view but is unwise due to the interpolation errors that would accrue.

6.4.3 VLES

The VLES model behaved like an LES model in its ability to resolve turbulence. Unlike LES, however, the domain was split into a turbulence-resolving region and a smoother RANS region like a DES model or other forms of hybrid RANS/LES model. By adapting to the local flow conditions the model

appeared to be able to represent the important scales of motion while suppressing model dissipation in other areas.

The near-wall region was not well captured by the model, probably because there was an intermittent RANS region along the bottom. However, the excellent agreement of mean velocity and Reynolds stresses was encouraging for the use of the model in flows in which the wall region is not the subject of interest. Slight under-prediction of reattachment length in both the 3D validation test suggests that the model is slightly too dissipative. The model requires further validation efforts to see if it can make quantitative predictions in other test cases. Many hybrid models are reported to have unrealistic behaviour in the ‘switch-over’ region between RANS and LES-like behaviour. By resolving the larger dynamics and modelling the smaller scales with RANS, it is not clear whether the full range of scales is accounted for.

6.4.4 Boundary Conditions

No-normal-flow side boundary conditions (used on adaptive meshes) have been found to exert a slight drag compared to periodic conditions. The effect on the results presented has not been fully quantified. Further work is required to implement periodic boundary conditions with parallel mesh adaptivity in Fluidity and should improve modelling accuracy.

6.4.5 Further Work

Future research will attempt to develop a truly robust parameter-free LES model which has great potential for reducing computational effort in solving complex industrial problems. For example, it may be possible to remove the *ad hoc* stability constraint $c_S \geq 0$. The anisotropic filter definition may need to be revised in order to improve its behaviour on highly stretched meshes. Alternatively, theoretical reasons might be found to show why the anisotropic filter width does not improve accuracy over an isotropic filter width.

Another problem is to find the optimum set of parameters for mesh adaptivity for resolving the dynamics affecting quantities of interest: reattachment length, mean velocity and Reynolds stresses. It has been shown for this problem that adapting to the curvature of the average velocity was a good strategy (certainly if one wishes to reduce the frequency of mesh adapts) because the flow evolves to a statistically steady state in which the average does not move; therefore instantaneous velocity is well captured by such a mesh. Careful consideration of the characteristics of a particular flow is necessary in order to choose a suitable target. A simple precursor simulation might identify the flow behaviour, for example showing whether it reaches

equilibrium or not, before embarking on a full adaptive study. Choice of minimum edge lengths is a critical parameter, as is the rate of change of edge lengths across the domain. The former determines how close the resolved scales are to the Kolmogorov lengthscale, while the latter acts to reduce the commutation error which inevitably results from varying filter width.

7 Comparison of Fluidity and CFX in Simulations of a Vortex Diode

A challenging flow problem from Rolls-Royce Submarines, the vortex diode, was chosen to further investigate CFD model behaviour and provide modelling recommendations to the company. CFX simulations with various turbulence models were conducted on placement at Rolls-Royce Submarines and Fluidity simulations with the $k - \varepsilon$, VLES and Smagorinsky LES models were carried out at Imperial College London. The strongly swirling flow makes this a difficult test case for CFD. Accurate prediction of pressure drop in the vortex is found to be sensitive to the choice of turbulence model, discretisation and solution algorithm. Models based on the Boussinesq hypothesis are unable to match experimental data.

Contents

7.1	Introduction	234
7.2	Experimental Data	238
7.3	Simulation Setup	241
7.3.1	Geometry and Meshing	241
7.3.2	CFX Model Setup	242
7.3.3	Fluidity Model Setup	244
7.4	Results	246
7.4.1	Flow Dynamics: Forward	246
7.4.2	Flow Dynamics: Reverse	251
7.4.3	Pressure Drop vs. Mass Flow Rate: Forward	253
7.4.4	Pressure Drop vs. Mass Flow Rate: Reverse	257
7.4.5	Diodicity	260
7.5	Discussion	262
7.5.1	Summary of Findings	262
7.5.2	Conclusions and Recommendations	263

7.1 Introduction

Vortex diodes (VDs) are passive devices used in various industries as ‘leaky no-return valves’, allowing flow preferentially in one direction over the other. The VD studied here, shown in Figure 7.1, is used to control the bypass flow in a reactor pressure vessel (RPV) under normal and accident conditions (see Figure 2.2). At present, Rolls-Royce have experimental results for the VD at a limited range of mass flow rates. It is important to be able to simulate its behaviour at a wider range of flow rates occurring in various operating conditions, which is the motivation for this research.

In this study, simulations of the VD were conducted using the commercial code CFX (ANSYS, 2011) and the open-source code Fluidity (AMCG, 2012). Actual dimensions of the device and experimental conditions are not given here to protect commercially and nationally sensitive information. Pressure drop predictions are made at a range of mass flow rates and compared to experimental data. The aim of this study is to investigate the behaviour of various CFD models in a challenging flow, rather than to make quantitative predictions for use in design and safety calculations.

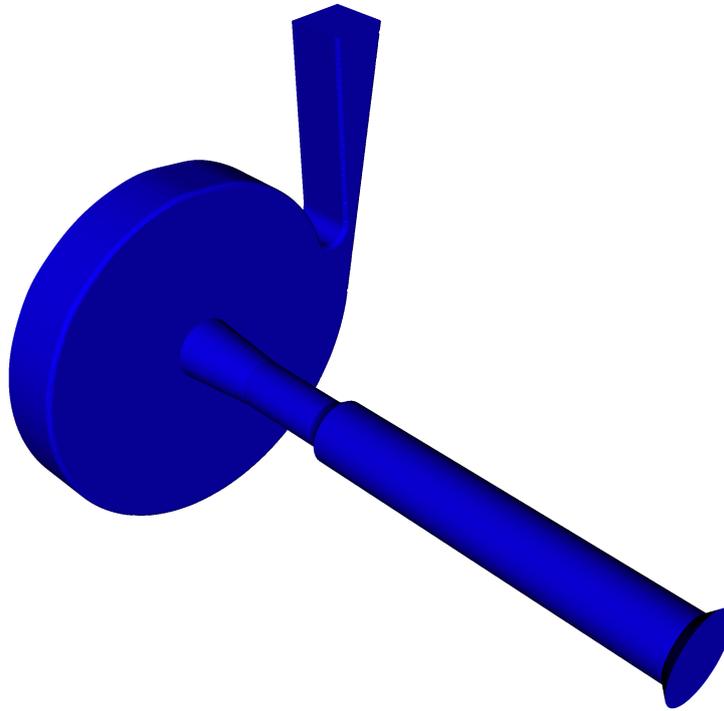


Figure 7.1: Vortex diode geometry.

The diode effect is achieved by designing the geometry so that a vortex is induced in one flow direction (the forward direction). Angular momentum is conserved in the vortex (away from the wall) causing a large pressure drop which resists flow. In the opposite (reverse) direction no vortex is induced and the pressure drop is consequently smaller. A factor of interest in the analysis of VDs is its ‘diodicity’, the ratio of forward and reverse pressure drops.

In forward mode, the flow enters the tapered tangential duct at the top of Figure 7.1, swirls inside the circular chamber, and leaves via the axial duct. The expansion sections in the axial duct reduce turbulence at the outflow. In reverse mode, flow enters the circular duct and spreads out evenly into the chamber, forming a toroidal vortex which is much weaker than the forward vortex. Figure 7.2 shows streamlines in the forward and reverse flow coloured by velocity in simulations using the VLES model in Fluidity at mass flow rates of 0.29 kg s^{-1} and 0.25 kg s^{-1} respectively.

Despite their geometric simplicity, flow through VDs can be complex. For example, in the vortex chamber, which is not perfectly circular due to the presence of the tangential inlet, the vortex may ‘wobble’ and the vortex core precesses periodically around the outlet pipe axis (Kulkarni et al.,

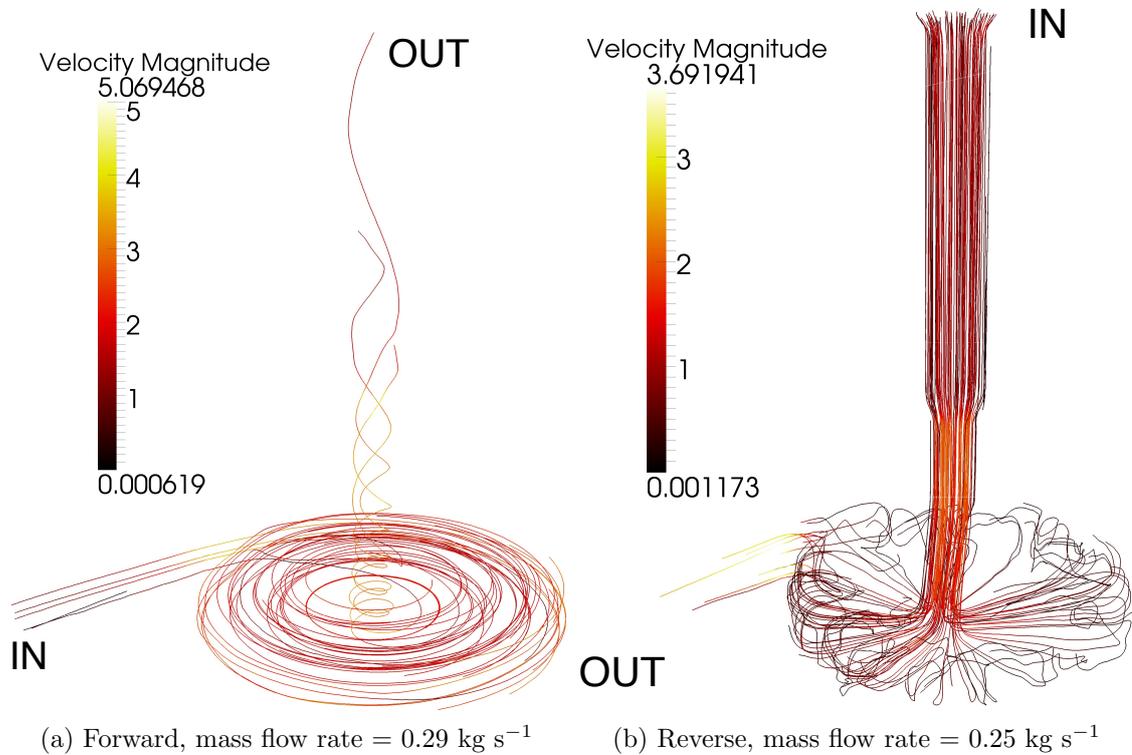


Figure 7.2: Streamlines in forward and reverse flow coloured by velocity (m s^{-1}) in simulations using the VLES model in Fluidity.

2008). This results in pressure fluctuations. A test of any CFD model will be whether it can reproduce this behaviour. Another effect is that at high Reynolds numbers the vortex relaminarises the forward flow by stretching it tangentially, suppressing turbulence (Ragab and Sreedhar, 1995; Kulkarni et al., 2008). The effect may be difficult to simulate with a turbulence model. Flow in the inlet and outlet ducts and in the reverse direction is turbulent, so a potential model has to be able to reproduce both laminar and turbulent regimes.

The critical factor in obtaining correct pressure drop and diodicity predictions with CFD is the formation of a suitably strong vortex in the forward direction. Without this the pressure drop would be solely determined by the sharpness of bends around which the fluid must travel and thus similar in each direction. It is important to understand how vortex formation and strength are affected by the computational mesh, boundary conditions, numerical method and turbulence model. The VD is a challenging test for CFD modelling because of the sensitivity of the pressure drop to all these factors.

There is a limited amount of research into simulating the flow through VDs, mostly using commercial CFD codes. The most thorough study is that of [Kulkarni et al. \(2008\)](#), who investigated the effect of various design and numerical parameters on the pressure drop using the commercial code FLUENT, obtaining good agreement with experiments over a wide range of Reynolds number (5,000 – 40,000). Poor predictions were given by the $k - \varepsilon$ model in the laminar vortex flow. The best results were obtained with no turbulence model, despite the presence of turbulent flow in the tangential inlet. In forward mode the vortex centre did not align with the chamber centre and oscillated around the axis resulting in a fluctuating pressure drop. The average pressure drop over a large number of iterations was found and the Euler number (ratio of the pressure drop from frictional loss to the pressure drop from inertial effects) was calculated by $Eu = 2\Delta p / \rho u_I^2$, where u_I is the inlet velocity, Δp is the pressure drop and ρ is the fluid density. Diodicity was calculated by $Eu_{\text{fwd}}/Eu_{\text{rev}}$. They found that diodicity increased with flow rate and above a critical flow rate, diodicity was constant.

[Kulkarni et al. \(2008\)](#) set the outlet boundary conditions to “outlet vent” in FLUENT which specified a pressure loss coefficient across the outlet and an ambient pressure. Dimensionless pressure loss coefficients parameterise the hydraulic resistance of a component and may be constant or Reynolds number-dependent ([Çengel and Turner, 2001](#)). A large value of the loss coefficient functioned like a velocity absorption term, preventing turbulent flow re-entering the domain and destabilising the simulation ([Kulkarni et al., 2008](#)). Neither pressure drop nor velocity profiles in the chamber were significantly affected by the inlet velocity profile. It is for this reason that constant inlet profiles have been used in the current study.

The discretisation scheme for pressure is significant in strongly swirling flows ([Kulkarni et al., 2008](#)). [Muntean et al. \(2005\)](#) and [Susan-Resiga et al. \(2005\)](#) found that the PRESTO (Pressure Staggered Option) on structured meshes performed well in swirling pipe flow because it could capture large pressure gradients. PRESTO gave a higher pressure drop in forward mode and a similar pressure drop in reverse mode to the standard second-order pressure scheme but took far more iterations to converge ([Kulkarni et al., 2008](#)).

The transient flow during spin-up was studied using the Smagorinsky LES model by [Yin et al. \(2010\)](#). Their results were accurate compared to experimental data, although meshes up to 6,100,000 nodes were required. However, as recognised by the authors themselves, their findings are merely a good starting point in attempting to simulate transient flows in vortex diodes.

[Yang and Priestman \(1991\)](#) used the standard $k - \varepsilon$ model and a Reynolds

stress model (RSM) (cf. §4.2.5) to simulate forward flow in a VD. They found that the RSM was much more accurate, predicting pressure drops close to experimental results. The $k-\varepsilon$ model over-predicted turbulent kinetic energy in the vortex chamber, sapping energy from the mean flow and thus reducing vortex strength. Additionally, their study found that the flow was insensitive to inlet turbulence intensity and sensitive to inlet flow swirling strength.

The realisable $k-\varepsilon$ model, a variant of the standard $k-\varepsilon$ model, was used by Yoder Jnr. et al. (2011) to simulate forward flow. Their results indicated that diodicity depended on mesh resolution, and mesh independence was not yet obtained with a 1.5 million node mesh. Experimental diodicity results were significantly larger than simulated results.

Because of the spread of differing results in the literature, it has been decided to test a range of commonly-used models in the current study: the standard $k-\varepsilon$ and $k-\omega$ SST models in CFX and the standard $k-\varepsilon$, VLES and Smagorinsky LES models in Fluidity. Following the advice of Kulkarni et al. (2008), simulations without a turbulence model were also performed in both codes. Some RSMs are available in CFX but were not tested; their ability to predict the vortex flow are a subject for further research.

7.2 Experimental Data

Experimental data for the current VD is available for a range of flow rates in forward and reverse flow modes (0.22 – 0.67 kg s⁻¹ forward and 0.27 – 1.44 kg s⁻¹ reverse) from Burgess and Tough (1971) and Burgess (1972). Two experiments were performed at high pressure and at ambient temperature using different pressure measurement techniques and one was conducted at 100°C. It was found that the temperature did not affect the results which validates the use of fluid properties at ambient temperature in simulations. An uncertainty of ±15% was calculated by Burgess (1972) for the pressure drop measurements which sets the bar for the accuracy of CFD predictions.

Correlations were fitted to the mass flow-pressure drop curves:

$$Q_{\text{fwd}} = 12.8(dp/\rho)^{1/2.14}, \quad (7.1a)$$

$$Q_{\text{rev}} = 97.2(dp/\rho)^{1/1.92}, \quad (7.1b)$$

where Q is the mass flow rate, dp is the pressure drop, ρ is the fluid density and all quantities are in Imperial units. These correlations are plotted with 15% error bounds in Figures 7.12 (page 255) and 7.14 (page 259) respectively. They have been extrapolated to cover the range of mass flow rates used in simulations.

7.2 Experimental Data

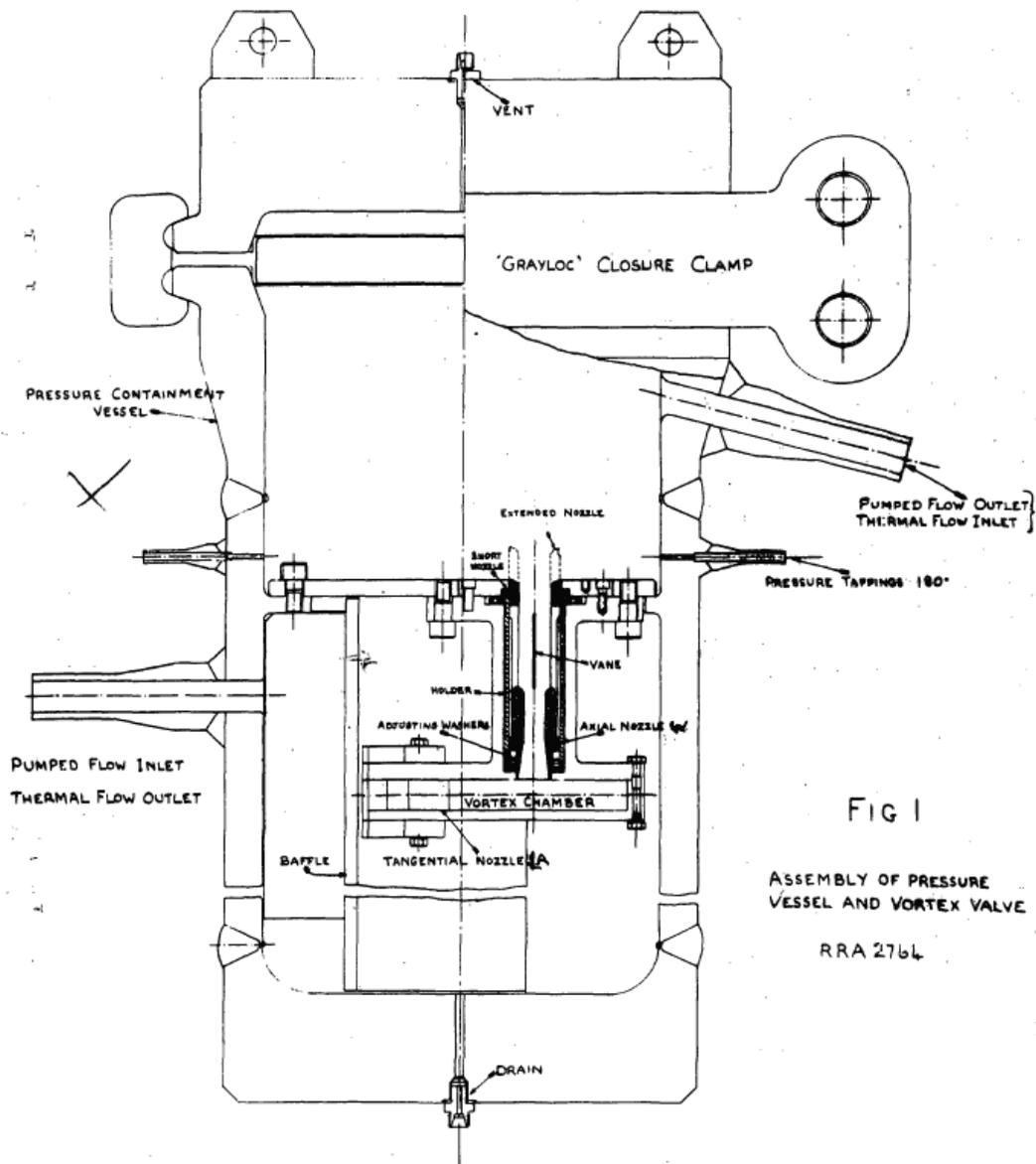


Figure 7.3: Side view of vortex diode experimental setup showing inlet and outlet plena and pressure tapping Burgess and Tough (1971). Pressure tappings shown near top of VD pipe.

7.2 Experimental Data

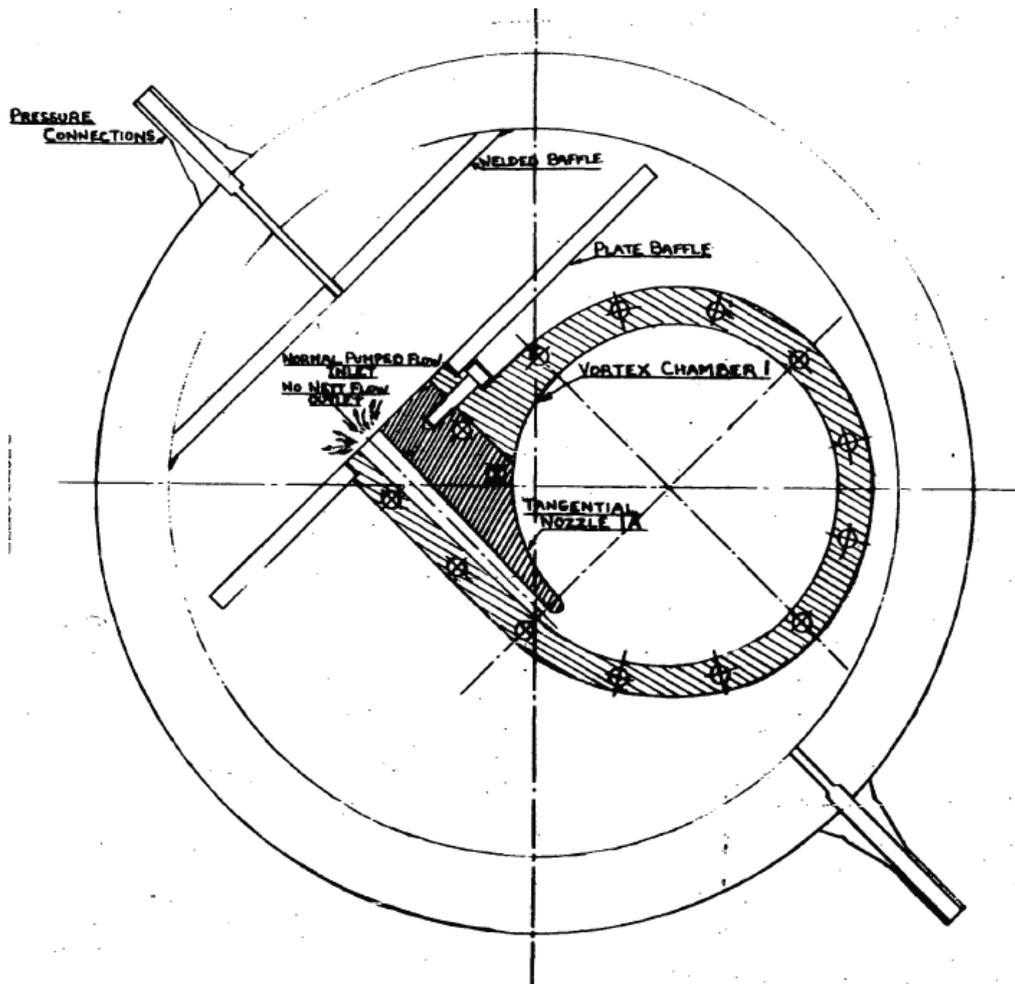


Figure 7.4: Top view of vortex diode experimental setup showing inlet plenum, vortex chamber and pressure tapping Burgess and Tough (1971).

The experimental setup is shown from the side and top in Figures 7.3 and 7.4. In forward mode water was pumped into a chamber and through a baffle plate before entering the vortex diode tangential nozzle. Water exited the VD axial nozzle into a large outlet plenum before exiting in a slanting pipe. Inlet pressure measurements were taken at an unspecified location upstream and outlet pressure measurements in the outlet plenum close to the VD axial nozzle. Because the experimental pressure measurements were not taken at the exact inlet/outlet of the VD, but at points in the inlet/outlet plena, the values of pressure drop do not represent the exact pressure drop across the VD. An additional uncertainty is thus introduced on top of the $\pm 15\%$.

Unfortunately it was impossible to recreate the setup in CFD because of the lack of detailed drawings. Although exact drawings are available for the diode itself, the best available drawings of the experimental apparatus are Figures 7.3 and 7.4. The exact locations of the pressure measurements cannot be replicated. However, comparing diodicity values may remove the influence of the location of pressure measurements. Diodicity is dimensionless so the absolute values of pressure drop are not being compared.

7.3 Simulation Setup

7.3.1 Geometry and Meshing

The VD geometry, Figure 7.1, was provided in Parasolid (.igs) format and imported into ICEMCFD and GiD mesh generation software. The former produces meshes compatible with CFX, the latter with Fluidity. Although best practice with ICEMCFD and CFX is to use prismatic (stretched hexahedral) boundary layers to enhance boundary layer resolution, Fluidity is unable to accept mixed-element meshes such as a prism/tetrahedron mix. In order to be able to compare the results of the two codes, meshes without prismatic layers were used i.e. only containing tetrahedral elements. Tests showed that addition of a prismatic boundary layer did not significantly affect pressure drop predictions. Four times finer elements were specified on the walls than in the volume. The total number of nodes was approximately 105,000. A mesh with the same specifications was generated for Fluidity.

A second mesh was generated for Fluidity by running an adaptive simulation to steady state in an attempt to optimise the mesh. A final ‘checkpoint’ was generated containing the adapted mesh which could be used for other simulations. The adapted mesh had an approximately uniform resolution and simulations using this mesh resulted in better pressure drop predictions in forward mode. The likely reason for this is that the solution interpolation

error had been reduced on the adapted mesh. Uniformity of the mesh is probably due to the laminar nature of the vortex flow. Although the Fluidity and CFX meshes were now different, Fluidity pressure drop predictions were still lower than CFX and it was decided to use the checkpoint mesh thereafter for all simulations and results presented here.

As the purpose of this study is to understand model behaviour rather than conduct model validation, resolution has been kept deliberately low in order to permit a large number of simulations with the limited available computing power at Rolls-Royce Submarines.

7.3.2 CFX Model Setup

Temperature was set to ambient, density to the actual experimental value and molecular viscosity to the corresponding value for water from thermodynamic tables. It was found that true steady-state solutions were impossible to obtain in CFX. The equation residuals oscillated suggesting that the true solution was a transient quasi-periodic one. A transient run confirmed this supposition: the vortex was observed to ‘wobble’ or precess around the chamber as observed by [Kulkarni et al. \(2008\)](#). Pseudo-steady-state solutions were then used as initial conditions for transient runs. Table 7.1 summarises the transient simulation setup.

Spatial discretisation	CV-FE (cf. §3.5.5)
Time discretisation (steady-state)	implicit (first-order)
Time discretisation (transient)	second-order backward Euler
Pressure-velocity coupling	Rhie-Chow
Momentum equation advection scheme	high resolution
(U-)RANS model equation scheme	first-order
Max. nonlinear iterations per timestep	20
Timestep (secs)	4.5×10^{-5}
Steady state convergence criterion	judged by eye
Solver tolerance (relative)	1×10^{-5}
Mass flow rates (kg s^{-1})	0.01, 0.02, 0.04, 0.1, 0.2, 0.43, 0.86, 1.7, 4.0, 8.0
Reynolds number range*	160 – 126,000

Table 7.1: CFX transient simulation solution parameters. High resolution scheme blends first order and second order ([ANSYS, 2011](#)). *Reynolds number at smallest flow cross-section (tangential chamber port).

The implicit second-order backward Euler scheme used for time discreti-

sation in CFX is second-order accurate in time but not bounded (ANSYS, 2011). This scheme treats the diffusion term implicitly and the advection term explicitly and is diffusive (Heinrichs, 1993). The high resolution scheme is a blend of first- and second-order-accurate discretisations resulting in a bounded second-order-accurate upwind-biased scheme (ANSYS, 2011). When using a RANS or U-RANS two-equation turbulence model, the high resolution scheme was applied to the momentum and model advection-diffusion-reaction equations.

7.3.2.1 Turbulence Models

CFX simulations with no model, the $k - \varepsilon$ model and the $k - \omega$ SST model (cf. Chapter 4) were performed. All transient simulations started from an initial condition provided by a steady-state solution calculated with the same model including no-model runs.

7.3.2.2 Boundary Conditions

Inlet boundary conditions for RANS runs in CFX were prescribed as mass flow rate conditions with 5% inlet turbulence superimposed. An inlet condition with no added turbulence was also tested in RANS runs and found to cause the simulations to diverge because the flow in the inlet duct was insufficiently stabilised. In no-model runs laminar inflow conditions were prescribed. CFX prescribed a velocity profile which satisfied the specified mass flow rate.

CFX zero-pressure ‘outlet/opening’ type boundary conditions were prescribed at the outlet (not the same as the ‘outlet vent’ condition used by Kulkarni et al. (2008)). These do not allow flow to come back into the outflow: if a turbulent eddy is advected out of the domain, detection of inward-pointing vectors causes the program to build artificial ‘walls’ in order to prevent numerical instability (ANSYS, 2011).

Solid walls were prescribed with no-slip conditions on velocity with the no-model runs. The standard high-Re $k - \varepsilon$ wall functions were used in scalable form (cf. §4.3.2) with the $k - \varepsilon$ model, and CFX ‘automatic wall treatment’ were used with the $k - \omega$ model (ANSYS, 2011).

7.3.2.3 Convergence Control

A maximum of ten nonlinear iterations per timestep was recommended in order to obtain convergence to the solver tolerance (ANSYS, 2011). In practice it was found that twenty were necessary to obtain convergence during the first few hundred timesteps; during the latter stages as few as three

were necessary. The large number of iterations is required for the implicit time-stepping scheme to converge (Jameson, 1991). CFX runs were continued until either the pressure measured at points at the inflow and outflow planes had converged (judged by eye from plot in CFX-Solve), or when 20,000 timesteps (0.9 s) had elapsed, whichever came sooner. If pressure had not converged by 20,000 timesteps, then the results were still used as long as they displayed ‘spin-up behaviour: i.e. the vortex was gaining strength in the expected manner and the pressure drop was showing signs of converging. Running out of time was only a problem in the forward direction because of the time taken for the vortex to gain strength which increased with mass flow rate. The final value of the pressure drop across the VD was found from the difference of the average pressures at the inflow and outflow. Although the value fluctuated slightly, the value at the final timestep was used (rather than a time-average) since the fluctuations were relatively small and well within experimental uncertainty.

7.3.3 Fluidity Model Setup

All Fluidity simulations were transient and started from zero initial conditions. Table 7.2 summarises the Fluidity simulation solution parameters.

Spatial discretisation	P_1 CG – P_1 CG
Time discretisation	Crank-Nicolson
Advection scheme	central difference (Galerkin)
RANS model equation scheme	high-order flux-limited (CV-FE)
Nonlinear iterations per timestep	2
Timestep (secs)	adaptive: $CFL \leq 4$
Steady-state convergence criterion	judged by eye
Solver tolerance (relative)	1×10^{-5}
Mass flow rates (kg s^{-1})	0.006, 0.013, 0.028, 0.07, 0.14, 0.285, 0.57, 1.15, 2.8, 5.5
Reynolds number range*	100 – 93700

Table 7.2: Fluidity simulation solution parameters. *Reynolds number at smallest flow cross-section (tangential chamber port).

7.3.3.1 Turbulence Models

In Fluidity several turbulence models were tested: no model, the tensor Smagorinsky LES model (cf. §5.2.1), the $k - \varepsilon$ U-RANS model (cf. §4.2.3)

and the VLES model (cf. §4.4.4). In practice, not using a model led to solver divergence because there was insufficient numerical dissipation to stabilise the simulations. No results from these simulations are presented. Suggestions for further research into numerically stabilised simulations of a VD are given in the discussion (§7.5).

7.3.3.2 Boundary Conditions

In line with the findings of Kulkarni et al. (2008), flat profile inlet boundary conditions were used for velocity. The profile was ramped to zero at the edge of the relevant inlet within 10% of the edge (forward or reverse) to aid development of a boundary layer and improve turbulence model stability. In the absence of good quality experimental data on the inlet flow this was judged to be the best option. Fluidity does not have an option for prescribing bulk quantities such as mass flow rate as boundary conditions. The maximum velocity was adjusted so that the integral over the inlet gave roughly the same values of mass flow rate as those used in CFX.

In Fluidity, inlet turbulence can be applied directly to the velocity field by the synthetic eddy method (SEM) (cf. §3.4.2). SEM was calibrated to generate 5% turbulent intensity, similar to the CFX inlet condition, but the addition of SEM made no difference to the Fluidity pressure drop results and unless otherwise stated all Fluidity results presented here used a laminar inlet condition. The reasons for these observations are discussed in §7.4.3.

k and ε were set to values which generated the same magnitude of eddy viscosity as observed in CFX at the inlet plane. However, setting k and ε to zero made no difference to the pressure drop and simulations did not diverge in contrast to CFX. A possible reason is that the realisations of the $k - \varepsilon$ model in CFX and Fluidity differ in their treatment of inlet conditions.

Wall boundary conditions were provided by the standard high-Re wall functions for k and ε and the weak log law on velocity in $k - \varepsilon$ and VLES simulations. LES simulations used strong no-slip boundary conditions.

7.3.3.3 Solution

A linear momentum equation of state was used in Fluidity which assumes density is a variable and thus requires the inclusion of density in the CFD model equations. This allowed the actual value of density from the experimental data to be used but it was found that Fluidity's solvers did not cope well with large values of density. Therefore, the viscosity was normalised by the actual density and the simulation density was set to one. The resultant equations were unchanged since they were simply divided through by the

density. Then the resultant pressure drops and mass flow rates were scaled up by the actual density.

7.3.3.4 Convergence Control

The Fluidity simulations were allowed to run until the \mathcal{L}^2 norms of velocity and pressure had been judged by eye to have reached pseudo-steady-state (allowing for periodic oscillations due to vortex precession around the chamber) in the same manner as CFX: see for example the pressure drop plot in Figure 7.13b. The solver convergence criterion was met at every timestep by carrying out two nonlinear iterations for velocity and pressure and three for k and ε . Fewer iterations per timestep than CFX were needed, perhaps because CFX is a coupled solver whereas Fluidity solves fields separately in a Picard iteration (cf. §3.5.3). Pressure measurements and mass flow rates were taken as integrals over the inlet and outlet planes.

7.4 Results

A note of caution is given here on the interpretation of pressure drop calculations. When using a turbulence model based on the Boussinesq eddy-viscosity hypothesis (cf. §3.30), the pressure that is returned by the CFD solver is the modified pressure, $p' = p + \frac{2}{3}k$. Therefore, any quantitative comparison of pressure values between different turbulence models, or to experimental data, is questionable. Subtracting $\frac{2}{3}k$ from the modified pressure is unreliable as a calculation of p because the inlet and outlet boundary values for k can differ between the RANS models tested. Additionally there is no available information on k in the LES simulations. The only true pressure values in these results are generated by the no-model model in CFX and Fluidity. However, as noted in §7.2, it is acceptable to match the diodicity values, since the modifications in forward and reverse flow will partially cancel out. Nevertheless, an unknown modelling uncertainty has to be included to account for the pressure modification.

7.4.1 Flow Dynamics: Forward

The qualitative results presented in this and the next section are intended to highlight some differences between the turbulence models used in each of the codes. Pressure values quoted in the captions and in the text of this section are not the total drop across the VD inlet and exit but only across the vortex itself and are given as indicators of model performance.

7.4 Results

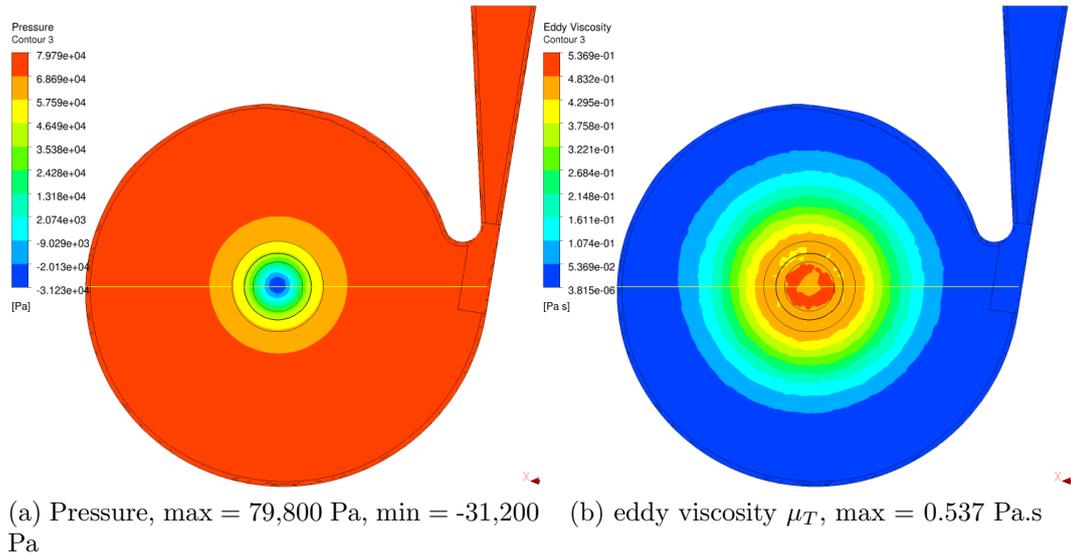


Figure 7.5: Flow dynamics in the forward direction, CFX, $k-\omega$ SST model, mass flow rate 0.43 kg s^{-1} .

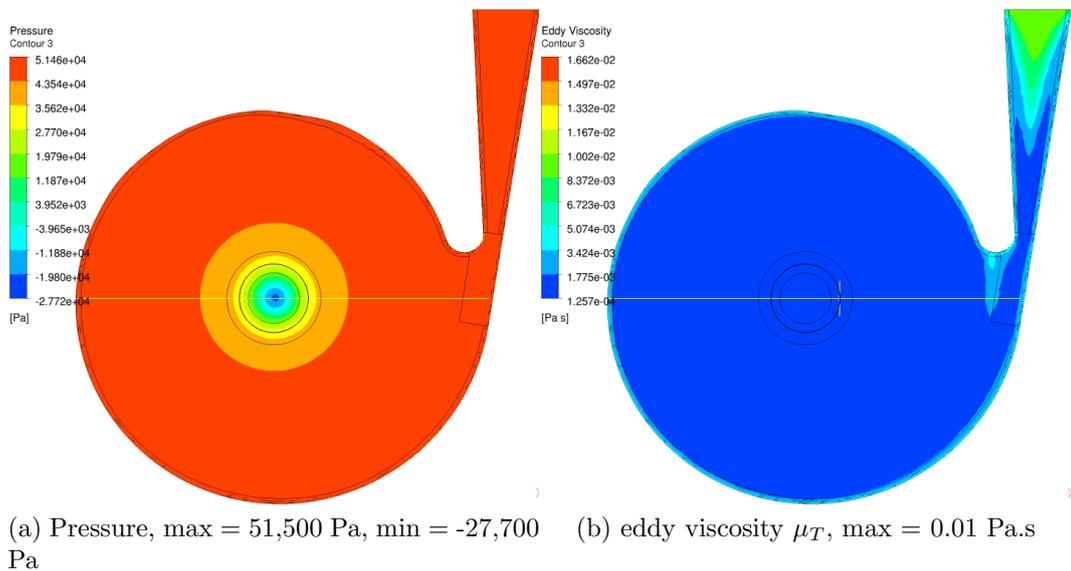


Figure 7.6: Flow dynamics in the forward direction, CFX, $k-\epsilon$ model, mass flow rate 0.43 kg s^{-1} .

7.4.1.1 CFX

Figures 7.5a and 7.5b show pressure and eddy viscosity on the centre plane through the vortex chamber in forward flow simulated by CFX with the $k-\omega$ SST model at a mass flow rate of 0.43 kg s^{-1} . Figures 7.6a and 7.6b show pressure and eddy viscosity on the same plane at the same mass flow rate but with the $k-\varepsilon$ model. Whilst the pressure distribution is similar to the $k-\omega$ SST model (Figure 7.5a), the pressure drop across the vortex (from inlet to centre) is lower – approximately 79,000 Pa for $k-\varepsilon$ versus 111,000 Pa for $k-\omega$ SST – indicating that the vortex is not as strong when using the $k-\varepsilon$ model. Results were found to be independent of the cut plane height in the chamber.

Eddy viscosity is more strikingly different: with the $k-\varepsilon$ model the maximum values are at the inlet (0.01 Pa.s), in the wake or separation region behind the curved wall at the chamber inlet and on the chamber wall (approximately 0.005 Pa.s) with very low values in the vortex. In Figure 7.5b the maximum eddy viscosity is at the vortex centre (0.537 Pa.s). A potential reason for this is that the SST model allows for a stronger vortex, which results in a larger eddy viscosity. Further research is required to investigate this possibility. Small variations in ν_T at the inlet and in the wake region are not observed because they fit within the first contour band.

7.4.1.2 Fluidity

Figures 7.7a and 7.7b show pressure and eddy viscosity on the same plane as the above CFX results for Fluidity with the $k-\varepsilon$ model at a mass flow rate of 0.29 kg s^{-1} in the forward flow direction. The mass flow rate is different to the CFX results above for the reason explained in §7.3.3.2. As explained in the model setup description, pressure and eddy viscosity have been multiplied by density to recover their correct values. Taking into account the different mass flow rates, the pressure drop of 9200 Pa is still much lower than the CFX $k-\varepsilon$ result and the maximum eddy viscosity of 0.366 Pa.s is much greater. The eddy viscosity distribution is more comparable to the CFX $k-\omega$ result (Figure 7.5b) apart from a streak in the wake region behind the chamber entrance which is similar to the CFX $k-\varepsilon$ result. Here the strong eddy viscosity may be *preventing* vortex formation whereas in the CFX $k-\omega$ SST results the large eddy viscosity may be *because of* strong vortex formation. Further examination of the evolution of the vortex is required to determine whether this is the case.

Figures 7.8a and 7.8b show pressure and eddy viscosity with the VLES model, and Figures 7.8c and 7.8d with the Smagorinsky LES model, at a

7.4 Results

mass flow rate of 0.29 kg s^{-1} in the forward flow direction. VLES has made a significant improvement on the $k - \varepsilon$ model: pressure drop across the vortex here is approximately 17,300 Pa. Eddy viscosity shows a threefold decrease compared to $k - \varepsilon$ which may be the reason for the higher pressure drop. As described in §4.4.4, the VLES model is designed to scale down dissipation to an appropriate level for the mesh.

The LES results are the poorest of the three models tested in Fluidity: pressure drop is approximately 8230 Pa and tensor eddy viscosity magnitude is far greater with a maximum of 7.74 Pa.s occurring at the chamber inlet. Large amounts of dissipation may have inhibited the development of a strong vortex. The likely reason for this is that the Smagorinsky model does not scale the dissipation correctly at the wall: wall functions should have been used as explained in §5.1.6. Fairly high values of eddy viscosity are seen in the interior of the chamber; further research is required to determine the cause.

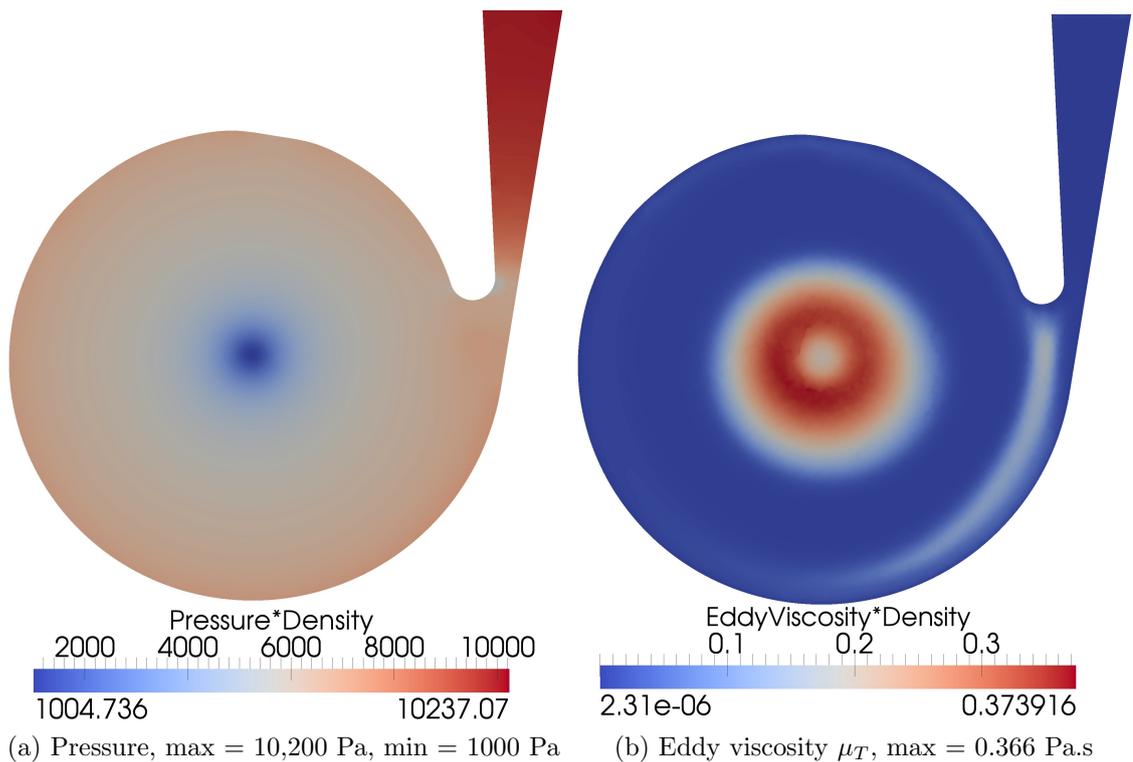


Figure 7.7: Flow dynamics in the forward direction, Fluidity, $k - \varepsilon$ model, mass flow rate 0.29 kg s^{-1} . Pressure and eddy viscosity have been multiplied by density to recover their true values.

7.4 Results

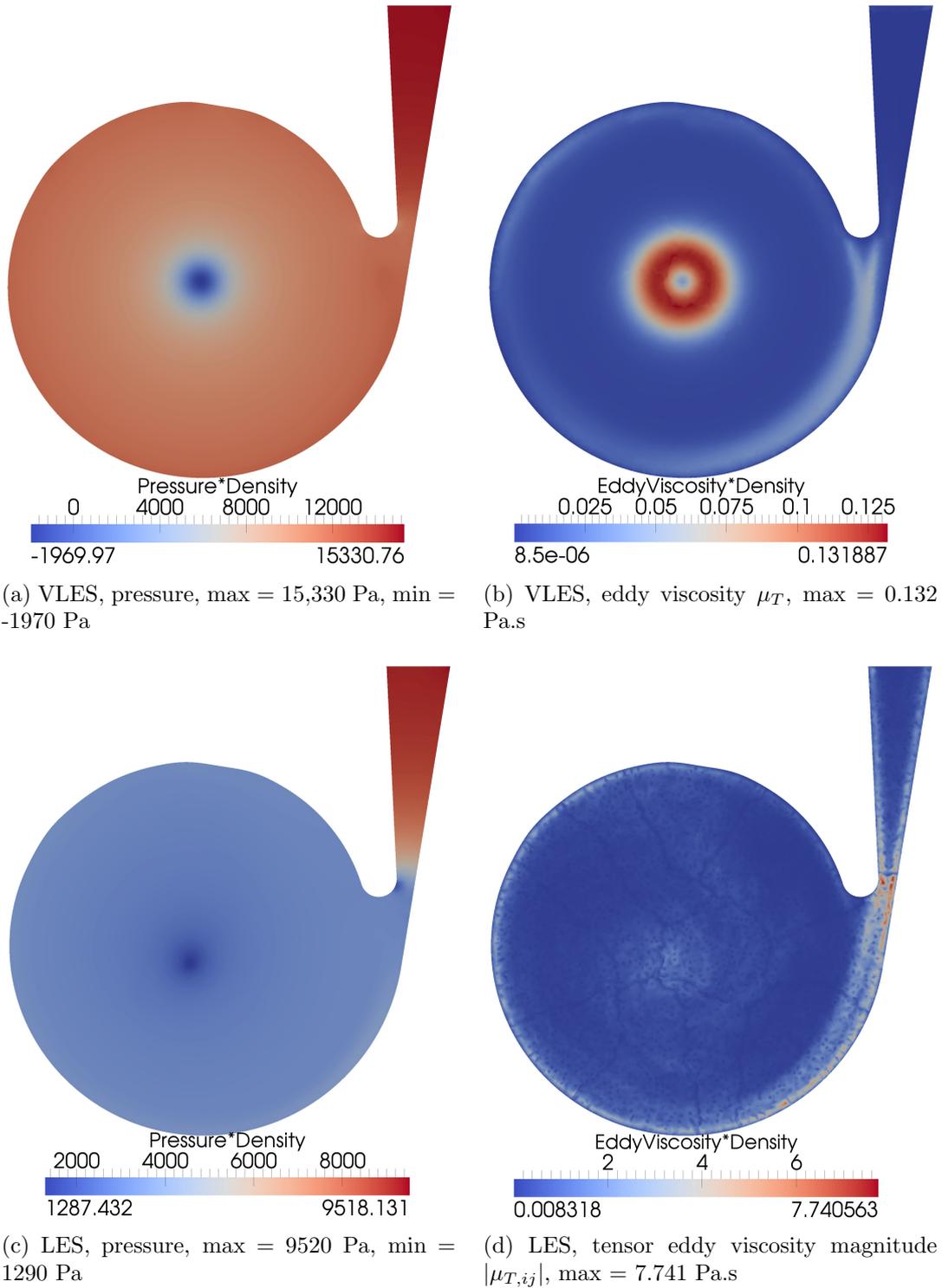


Figure 7.8: Flow dynamics in the forward direction, Fluidity, VLES and LES models, mass flow rate 0.29 kg s^{-1} . Pressure and eddy viscosity have been multiplied by density to recover their true values.

7.4.2 Flow Dynamics: Reverse

Unlike forward flow, reverse flow is not necessarily laminar. As observed in Figure 7.2b, reverse flow is characterised by a weak toroidal vortex in the chamber which becomes complex as the streamlines gather near the tangential chamber exit. The pressure drop in reverse flow is mostly caused by losses as the flow exits the vortex chamber where the flow path is narrowest.

7.4.2.1 CFX

Figures 7.9a, 7.9b and 7.9c show velocity, pressure and eddy viscosity on a horizontal cut plane through the vortex chamber in reverse flow simulated by CFX with the $k - \omega$ SST model at a mass flow rate of 0.43 kg s^{-1} . The lack of strong vorticity compared to the forward flow is evident from the relatively uniform low pressure of approximately 2200 Pa. The lowest pressure (-8500 Pa) and highest velocity occur at the chamber exit as it is the point of smallest flow cross-sectional area. The maximum eddy viscosity of 0.07 Pa.s, located in the toroidal vortex, is much smaller than in forward flow.

Figures 7.10a and 7.10b show velocity and eddy viscosity on the same plane at the same mass flow rate. However, with the $k - \varepsilon$ model pressure is not shown because it was no different to the $k - \omega$ SST pressure. Whilst the velocity pattern is similar to the $k - \omega$ SST model (Figure 7.9a), eddy viscosity is much larger at the chamber exit (0.503 Pa.s) where the peak velocity occurs. This is the opposite case to forward flow in which the $k - \omega$ SST model generated higher viscosity than the $k - \varepsilon$ model. Eddy viscosity is generally larger throughout the chamber than in the $k - \omega$ SST model.

7.4.2.2 Fluidity

Velocity and eddy viscosity at a mass flow rate of 0.25 kg s^{-1} are shown on the horizontal plane for the Fluidity $k - \varepsilon$ results in Figures 7.11a and 7.11b and the VLES results in Figures 7.11c and 7.11d. The distributions and maximum values of velocity and eddy viscosity in the $k - \varepsilon$ results are similar to the CFX $k - \varepsilon$ results if the differing mass flow rates are taken into account. The structure of the toroidal vortex can be made out in the higher values of velocity and eddy viscosity forming a ring around the chamber. The main differences are that the toroidal structure is more uniform and that eddy viscosity remains large in the exit duct whereas in the CFX results it decreases.

VLES flow patterns are markedly different to $k - \varepsilon$ results: a toroid or ring is no longer evident but instead a mass of smaller chaotic turbulent scales.

7.4 Results

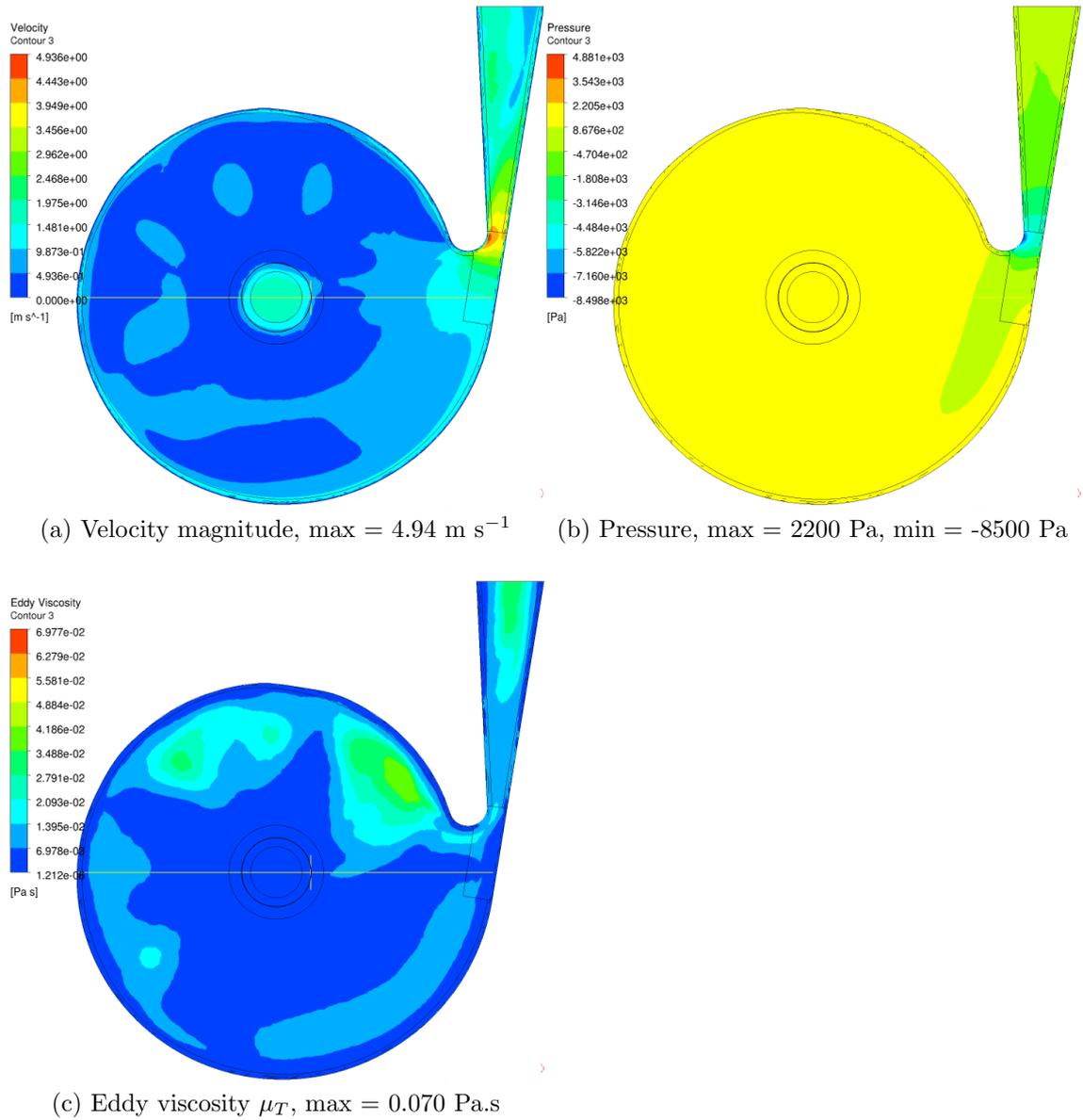


Figure 7.9: Flow dynamics in the reverse direction, CFX, $k-\omega$ SST model, mass flow rate 0.43 kg s⁻¹.

7.4 Results

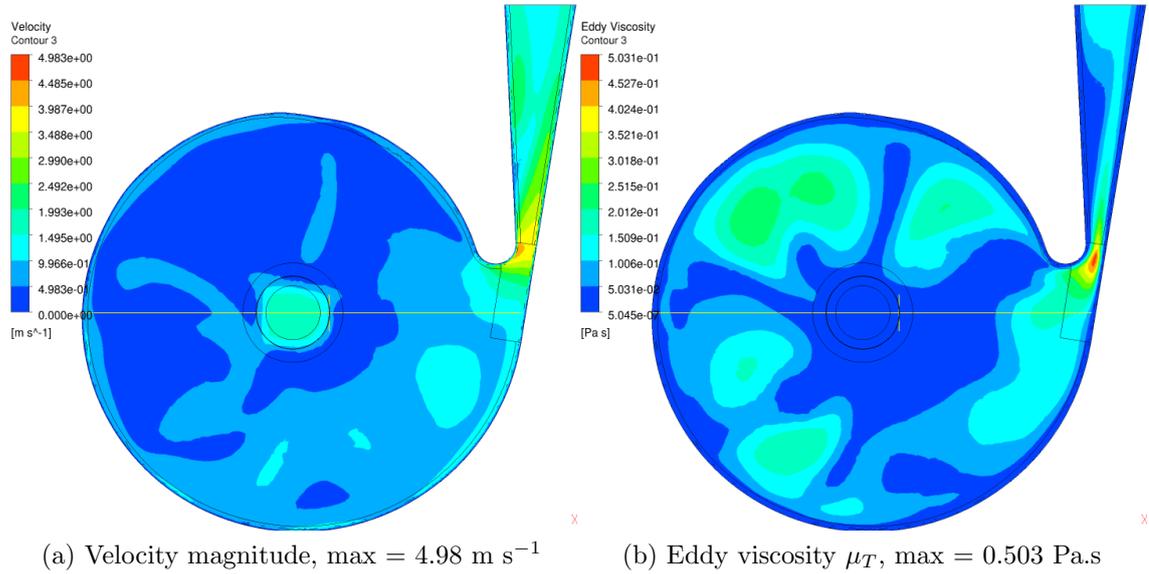


Figure 7.10: Flow dynamics in the reverse direction, CFX, $k - \varepsilon$ model, mass flow rate 0.43 kg s^{-1} .

As with the forward results, maximum eddy viscosity is almost three times smaller than that of the $k - \varepsilon$ results. The reduced dissipation has allowed smaller and more complex turbulent motions to be resolved. However, the result is a slightly higher reverse pressure drop than $k - \varepsilon$ (see Figure 7.12). More work is needed to determine whether a turbulence-resolving model is suitable for reverse mode, since the aim is to obtain a low pressure drop.

7.4.3 Pressure Drop vs. Mass Flow Rate: Forward

Figure 7.12 shows a log-log plot of pressure drop across the VD inlet and exit against mass flow rate in the forward direction for all CFX and Fluidity simulations. Data from the three experiments are marked in black and the forward flow correlation (7.1a) is plotted as a solid black line with $\pm 15\%$ uncertainty bounds shown as dash-dotted lines.

7.4.3.1 Steady State vs. Transient: CFX

Steady-state CFX results are marked with hollow green symbols and transient CFX results by filled green symbols. Running a steady-state simulation in CFX clearly resulted in a much lower vortex pressure drop compared to transient for all turbulence models tested. A fully implicit first-order time

7.4 Results

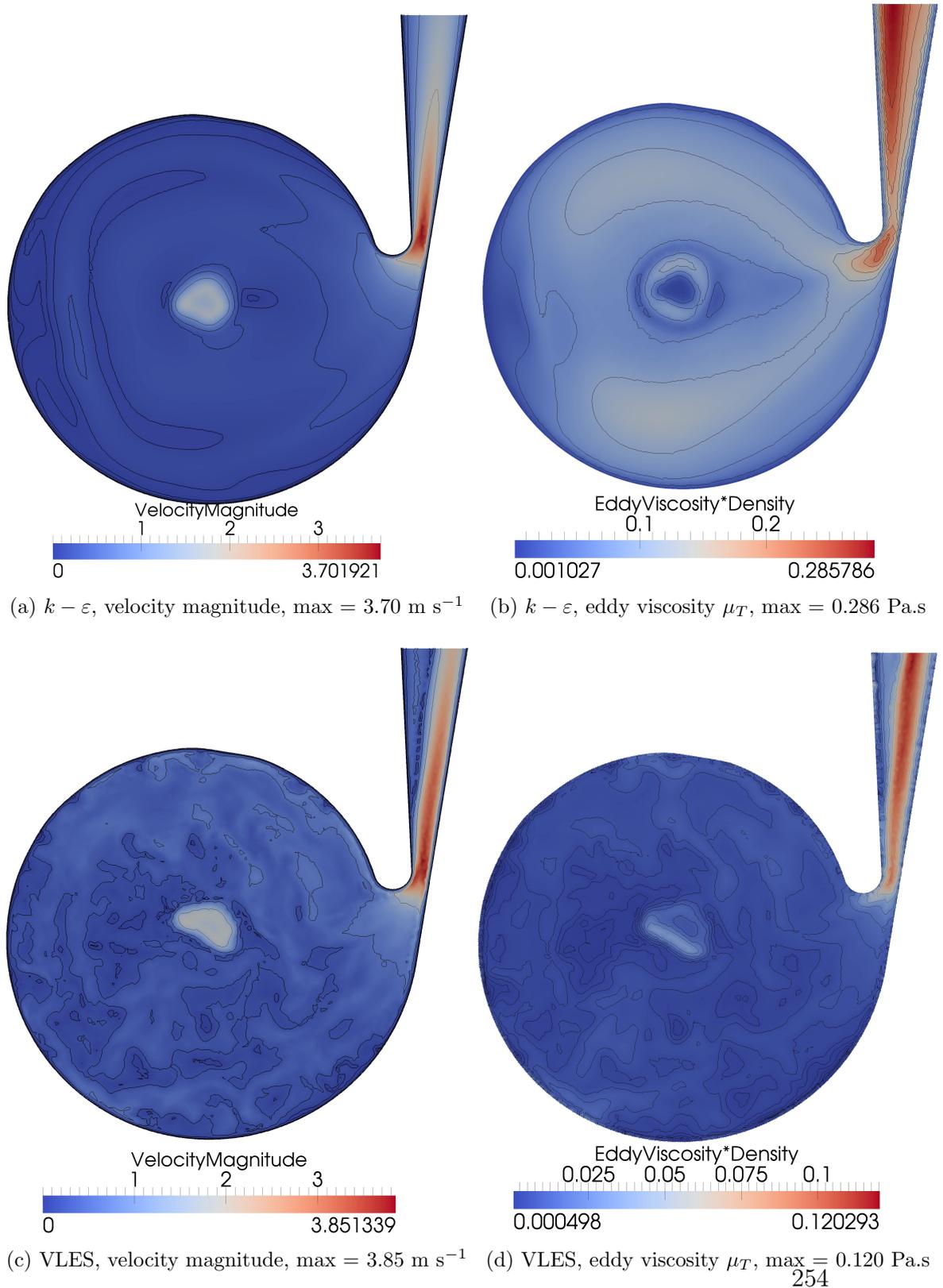


Figure 7.11: Flow dynamics in the reverse direction, Fluidity, $k - \varepsilon$ and VLES models, mass flow rate 0.25 kg s⁻¹. Pressure and eddy viscosity have been multiplied by density to recover true values. Contours are shown in black for clarity.

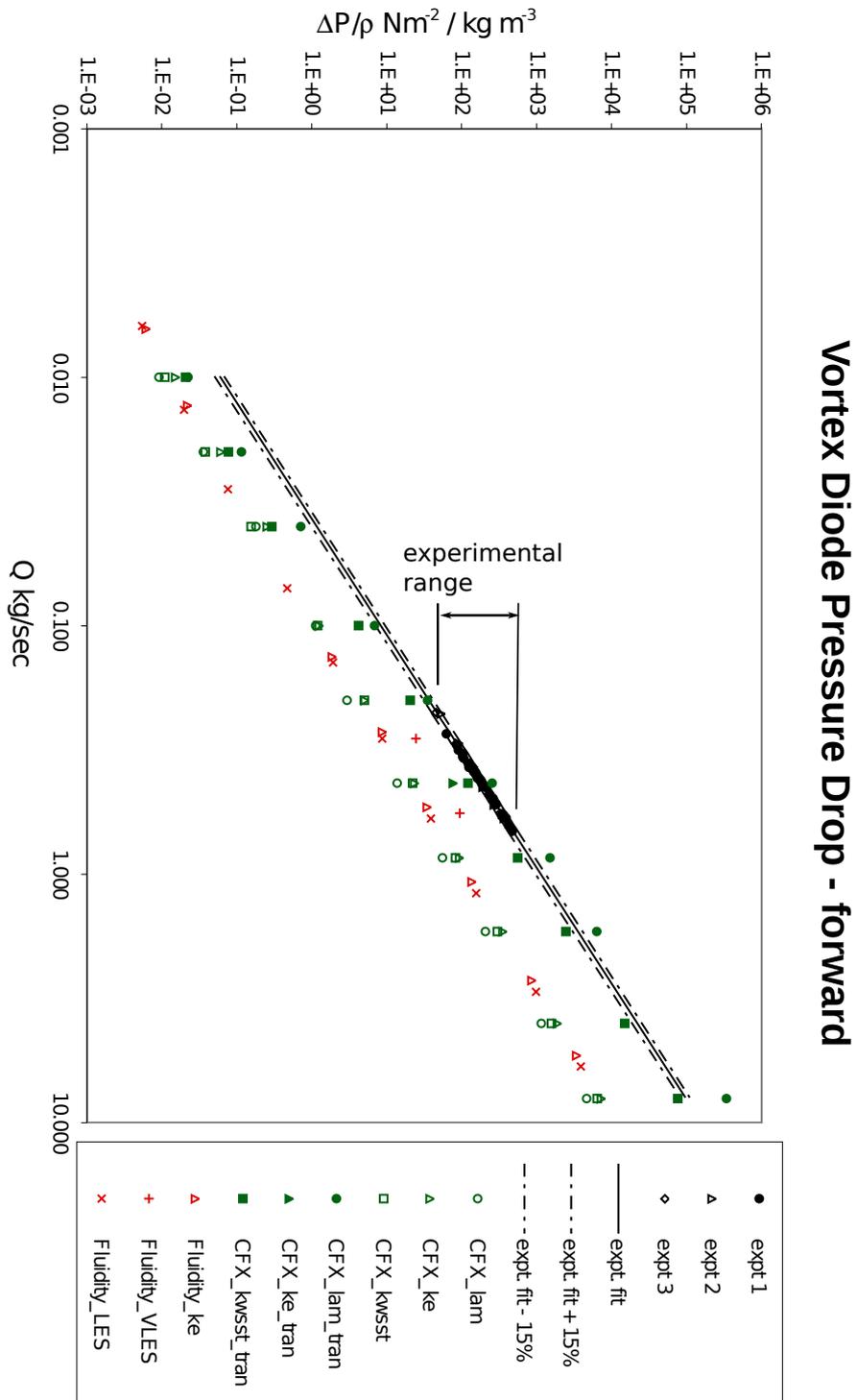


Figure 7.12: Forward pressure drop vs. mass flow rate compared to experimental data. ‘lam’ = no model; ‘ke’ = $k - \epsilon$; ‘kwsst’ = $k - \omega$ SST; ‘tran’ = transient; ‘expt 1, 2, 3’ = data from three experiments; ‘expt fit’ = best-fit power law to experimental data with $\pm 15\%$ uncertainty bounds on pressure drop values.

discretisation is used in steady-state mode which (although unconditionally stable) is numerically dissipative (Zienkiewicz and Taylor, 1997), counteracting vortex development. The second-order scheme used in transient mode is also dissipative but its higher level of accuracy may improve vortex development.

7.4.3.2 Effect of Turbulence Model: CFX

It is clear from Figure 7.12 that the strongest vortex is generated by the transient CFX run with no model but at higher flow rates ($Q > 1 \text{ kg s}^{-1}$) the pressure drop is significantly over-predicted, suggesting that some stabilising dissipation is needed. All the other CFX turbulence models tested appear to stabilise the flow at higher Re. Only one result (at 0.43 kg s^{-1}) has been obtained with the transient $k - \varepsilon$ model and more results are needed before its behaviour can be assessed properly.

The transient CFX run with the $k - \omega$ SST model is consistently closest to the data, even at high flow rates. This result helps to confirm the statement in §4.2.4 that $k - \omega$ models are superior to $k - \varepsilon$ models on coarse near-wall meshes and in strongly swirling flows. However, the quantitative comparison of pressure drop values is unreliable for the reasons given above relating to the eddy-viscosity hypothesis and the location of experimental pressure tappings. We must compare the reverse flow results and diodicity results before drawing conclusions.

At low flow rates ($Q < 0.1 \text{ kg s}^{-1}$) the $k - \omega$ SST and no-model plots display a change of behaviour, with pressure drops significantly under the experimental trend-line. This corresponds to $\text{Re} \approx 5000$ at the inlet or $\text{Re} \approx 1600$ at the narrowest flow cross-section (the tangential entrance to the vortex chamber) based on the width of the duct. It is possible that the change in behaviour corresponds to the laminar-turbulent transition point in the inlet duct. Unfortunately, no experimental data is available at such low flow rates so it is impossible to confirm whether the model is behaving correctly or if the low flow rates are simply outside of the model's range of applicability.

7.4.3.3 Effect of Turbulence Model: Fluidity

The $k - \varepsilon$ model (red triangles) and tensor Smagorinsky LES model (red x) in Fluidity predicted low pressure drops, corresponding to weak vortex formation, and closer to the steady-state CFX results than the transient CFX results. The low pressure drops predicted by the LES model may be a consequence of the forward flow being non-turbulent: the vortex motion

suppresses or ‘laminarises’ any turbulent motion entering the chamber. The VLES model results (red +) are much better than Fluidity $k - \varepsilon$ results and compare well with transient CFX $k - \varepsilon$ results, potentially because VLES is less dissipative than $k - \varepsilon$.

Figures 7.13a and 7.13b show the time series of pressure drop from simulations with VLES and $k - \varepsilon$ models respectively. A much higher pressure drop is generated by VLES with more complex pressure fluctuations. Both time series display oscillation indicating the presence of vortex precession. Oscillations become smaller as time passes, suggesting that during the initial ‘spin-up’ period the vortex centre is more mobile, settling into a smaller orbit as the flow nears a pseudo-steady state.

7.4.3.4 Effect of Inlet Turbulence: CFX

The effect of reducing turbulent intensity in CFX from 5% to 0% by changing the values of k and ε was to cause the forward simulations to diverge (reverse simulations with this condition were not carried out). It was apparent that some inlet turbulence was necessary to generate eddy viscosity in order to stabilise the flow at high flow rates where the flow separates off the curved surface of the chamber inlet.

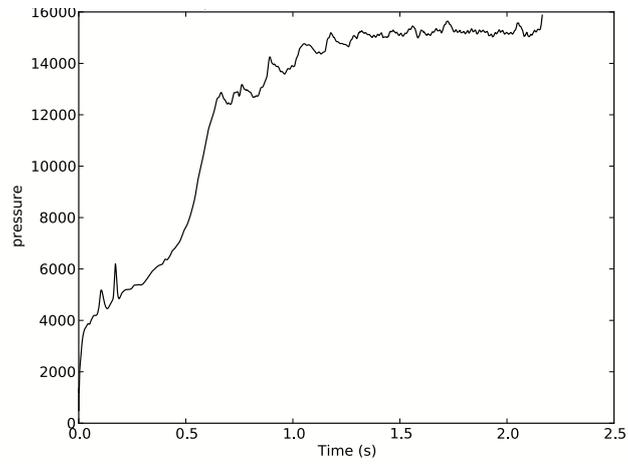
7.4.3.5 Effect of Inlet Turbulence: Fluidity

The effect on the pressure drop of adding synthetic inlet turbulence (SEM) to the $k - \varepsilon$ model is shown in Figure 7.13c. The addition of SEM does not change the pressure drop across the VD compared to Figure 7.13b, but merely adds high-frequency low-amplitude pressure fluctuations. It is likely that these pressure fluctuations are damped out by the vortex. Changing the inlet turbulent intensity in the same manner as for CFX (see previous section) did not affect the pressure drop either (results not shown).

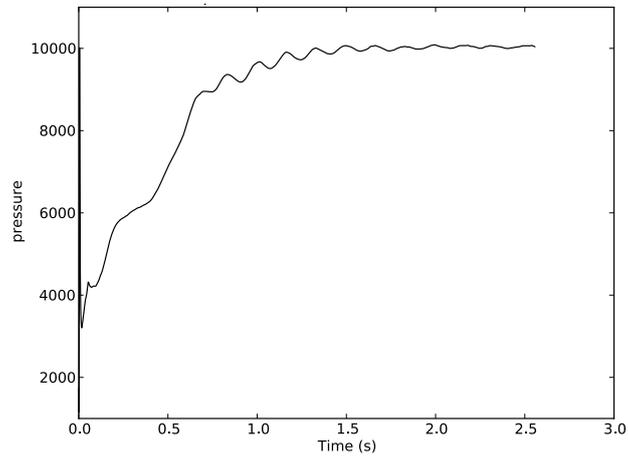
7.4.4 Pressure Drop vs. Mass Flow Rate: Reverse

Figure 7.14 shows a log-log plot of pressure drop across the VD against mass flow rate in the reverse direction for all CFX and Fluidity simulations. Data from the three experiments are marked in black and the reverse flow correlation (7.1b) is also plotted as a solid black line with uncertainty bounds plotted as dash-dotted lines.

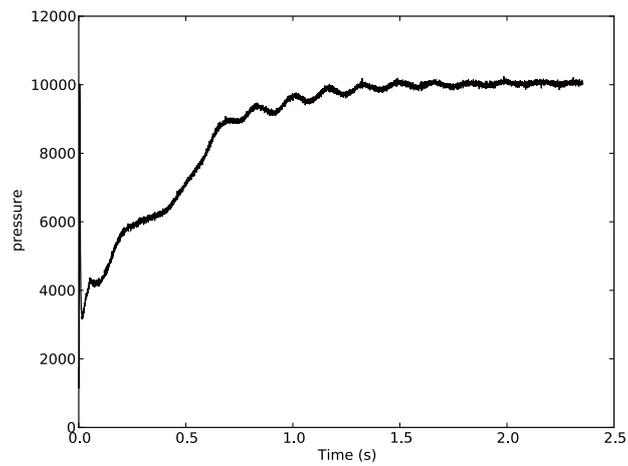
7.4 Results



(a) VLES model pressure drop



(b) $k-\epsilon$ model pressure drop



(c) $k-\epsilon$ model + SEM pressure drop

Figure 7.13: Effect of turbulence model on forward pressure drop, Fluidity, mass flow rate 0.29 kg s^{-1} .

Vortex Diode Pressure Drop - Reverse

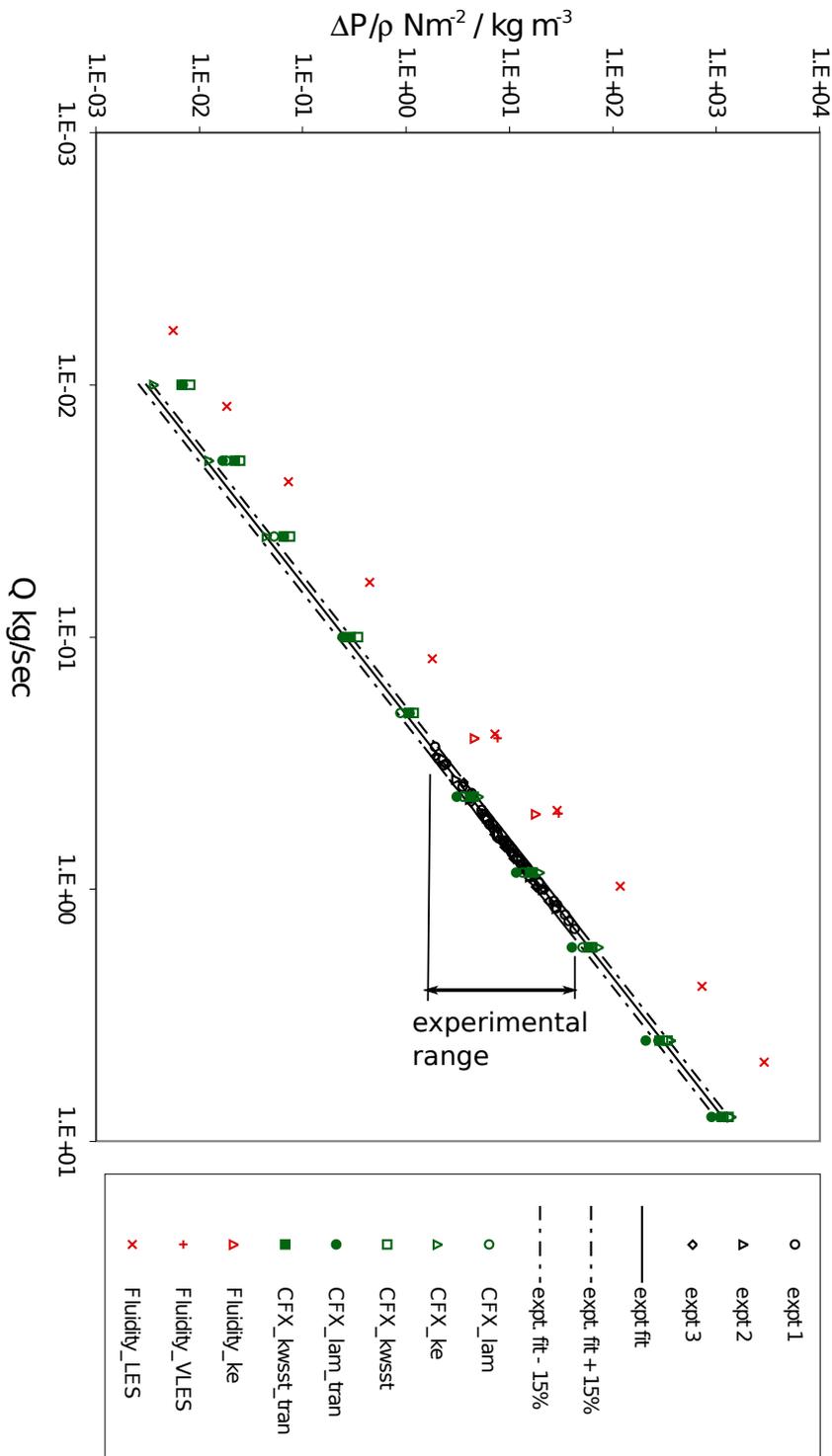


Figure 7.14: Reverse pressure drop vs. mass flow rate compared to experimental data. ‘lam’ = no model; ‘ke’ = $k - \epsilon$; ‘kwsst’ = $k - \omega$ SST; ‘tran’ = transient; ‘expt 1, 2, 3’ = data from three experiments; ‘expt fit’ = best-fit power law to experimental data with $\pm 15\%$ uncertainty bounds on pressure drop values. 259

7.4.4.1 Steady State vs. Transient: CFX

Both steady-state and transient CFX runs collapsed approximately onto the experimental trend line indicating that transient effects may not be as strong or affect the pressure drop to the same degree as in forward flow. Quantitative agreement with the experimental data was not anticipated for the reasons already given and further steps are needed to check whether these results are by pure coincidence. If diodicity measurements (presented in the next section) for the CFX results also agree with the data, then it is likely that the results are indeed accurate.

7.4.4.2 Effect of Turbulence Model: CFX

Compared to the forward flow the choice of turbulence model does not affect pressure drop to the same degree. The strong forward vortex appears to be a more challenging test of the models. Further tests are required to obtain transient $k - \varepsilon$ results in reverse flow.

7.4.4.3 Effect of Turbulence Model: Fluidity

Fluidity has predicted quite different reverse pressure drops to CFX. The $k - \varepsilon$ model is closest while LES and VLES models give identical pressure drop predictions.

7.4.5 Diodicity

Figure 7.15 plots the diodicity, or ratio of forward to reverse pressure drops, for the CFX and Fluidity results against the experimental data with $\pm 15\%$ error bars included. The experimental data covered a narrow range of flow rates with a large uncertainty so it was decided not to include a trend-line. Diodicity must be equal to one below a critical value of flow rate at which no vortices are formed and the flow in both directions is completely laminar. This baseline is marked by a dashed line in Figure 7.15.

It is notable that all the simulated diodicity results tend towards one at low mass flow rates. Steady-state CFX models predicted low values of diodicity which levelled off at high flow rates. Transient CFX models predicted ever-increasing diodicity at higher flow rates, suggesting that transient effects become more influential on pressure drop as flow rate increases.

The transient CFX simulation with the $k - \omega$ SST model has come close to the actual diodicity while the transient CFX simulation with no model over-predicts diodicity because the vortex strength was over-predicted in forward flow mode. Neither of these results was within experimental uncertainty

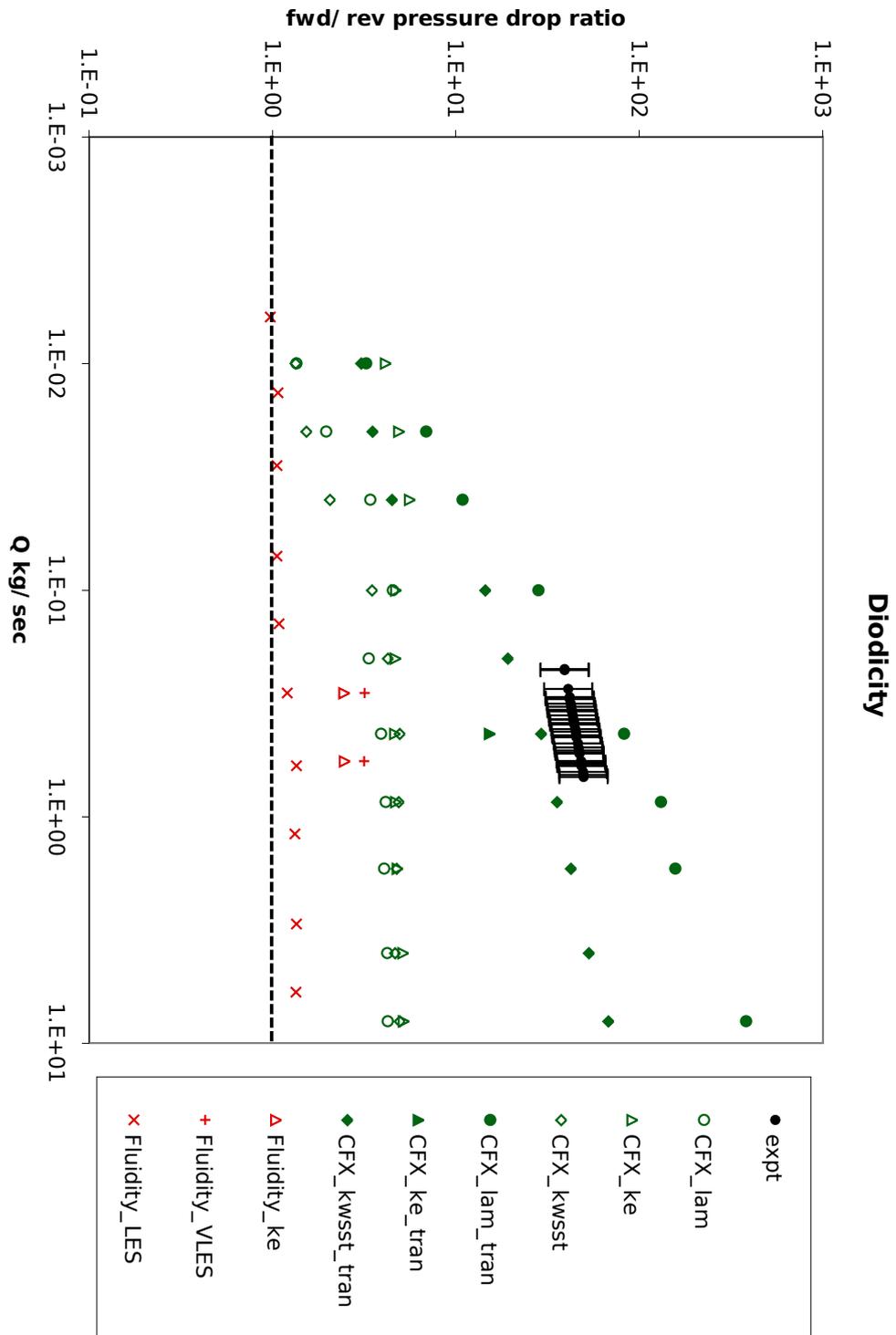


Figure 7.15: Diodicity predictions from CFX and Fluidity compared to experimental data. ‘lam’ = no model; ‘tran’ = transient; ‘expt’ = data from all experiments with $\pm 15\%$ error bars. Minimum possible diodicity of one is marked with a dashed line.

bounds, however. No other model, either in CFX or Fluidity, has come close to the actual diodicity. Fluidity LES results are close to one, meaning that the forward and reverse pressure drops are almost identical. No correction was made for the fact that mass flow rates in forward and reverse Fluidity simulations were not quite the same, but it is unlikely to make a big positive difference to these results.

7.5 Discussion

7.5.1 Summary of Findings

The purpose of this research was to provide information to Rolls-Royce Submarines on the behaviour of the vortex diode flow and on the abilities of two codes, using various turbulence models, to simulate it. Simulations have been performed using the open-source code Fluidity, developed by AMCG at Imperial College, and the commercial code CFX. Pressure drops were calculated across the diode in both forward and reverse flow modes and compared to experimental data. Uniform meshes with relatively coarse wall resolution were used to reduce simulation times.

In the forward direction, transient simulations in CFX gave better predictions than steady-state, possibly on account of their ability to capture vortex precession. The best fit to the experimental data was obtained with the $k - \omega$ SST model in a transient simulation in CFX. The $k - \varepsilon$ model was not a good choice for such strongly swirling flow, its over-dissipative nature sapping energy from the vortex, causing low pressure drops. CFX runs with no turbulence model were more unstable and over-predicted pressure drop. Fluidity runs with no turbulence model were also unstable, but more so than in CFX so no results have been presented.

In the reverse direction, several turbulence models in CFX came very close to the experimental data: $k - \omega$ SST (steady-state and transient), no model (steady-state and transient), and $k - \varepsilon$ (steady-state). The reverse flow resembled an impinging jet and vortices were much weaker than in the forward direction, presenting less of a modelling problem.

In Fluidity, the highest forward pressure drops were predicted by the scale-adaptive VLES model. This model curbed the excessive dissipation of the $k - \varepsilon$ model upon which it is based and consequently allowed a stronger vortex to form. The $k - \varepsilon$ model behaved similarly to the same model in CFX, being generally over-dissipative and preventing the vortex from forming properly. The VLES model demonstrated the ability to scale the turbulent contribution to dissipation in response to mesh size. In this way it compen-

sated for the failure of the $k - \varepsilon$ model rather than modelling the correct flow dynamics. The Smagorinsky LES model behaviour in this flow requires further investigation, being far too dissipative near the wall and preventing a strong vortex from forming.

Fluidity made poor quantitative predictions of reverse pressure drop compared to CFX. The best agreement with data was with the $k - \varepsilon$ model. The VLES model generated complex flow dynamics which increased reverse pressure drop. LES predicted similarly low pressure drops to VLES.

7.5.2 Conclusions and Recommendations

The simulations presented here did not faithfully follow ‘best-practice’ guidelines regarding mesh resolution owing to tight constraints on CFX licences and time on the company computing cluster. Near-wall regions were not highly resolved and mesh-refinement studies were not attempted. The vortex diode is a particularly challenging test case for a number of reasons. Measurement uncertainties in the experimental data and the pressure modification inherent in eddy-viscosity turbulence models made quantitative pressure drop predictions impossible. The true behaviour at low flow rates is unknown and if the flow regime does indeed change at lower Reynolds numbers, as suggested by the $k - \omega$ SST forward results, more data is needed to validate these results. A second source of data, particularly DNS, would be very helpful in future CFD studies of this flow.

Lower pressure drop predictions by Fluidity compared to CFX may be down to differences in solution procedure. CFX uses different spatial and temporal discretisations, a different timestepping algorithm and a coupled rather than segregated solver. These confounding variables are impossible to isolate, but further research into the properties of the schemes employed would be worthwhile. In particular, the dissipative properties of the second-order backward Euler scheme used in CFX may have stabilised the vortex flow in the absence of turbulence models. Other dissipative schemes, such as the stabilised finite element methods described in §3.5.6, are worth considering in future studies.

Differences in mesh generation may have favoured CFX too; it is very difficult to generate identical meshes in different software packages. The GiD package used for Fluidity meshes is quite a basic tool compared to ICEM-CFD, making high-quality mesh generation in complex geometry very difficult. Unfortunately neither of the meshing packages was compatible with the alternate CFD code. The use of an adapted mesh to generate the Fluidity results presented did not make a significant improvement on the GiD-generated initial mesh and further work is required to find the optimal adaptivity set-

tings for the flow.

Fixed meshes could be improved and refined in future studies, particularly the near-wall resolution which may play a part in the generation of a strong vortex. Techniques for generating prismatic boundary layers in a tetrahedra-only mesh suitable for Fluidity should be investigated. A translator could be written to allow meshes generated in ICEMCFD or a similarly sophisticated geometry and mesh generation package to be used in Fluidity.

Eddy viscosity models are known to encounter problems in strongly swirling flows or those with strong streamline curvature (Wilcox, 1998). This fact may well explain the failure of the tested turbulence models to correctly predict the forward pressure drop. Further research into VD simulations without a turbulence model would be worthwhile, in view of the fact that the forward vortex laminarises the flow. Such an approach would have to be dissipative enough to stabilise the vortex without damping it excessively. In the case of CFX, the second-order backward Euler time scheme may have stabilised the no-model simulations at moderate flow rates, but simulations at high flow rates ($Q > 1 \text{ kg s}^{-1}$) over-predicted pressure drop due to instability.

The numerical stabilisation techniques outlined in Chapter 3, such as the Discontinuous Galerkin method or SUPG, may add just enough stabilisation in Fluidity without suppressing vortex formation. Alternatively, a more complex model not based on the Boussinesq hypothesis, such as a Reynolds stress model (RSM), might be appropriate (Yang and Priestman, 1991; Chabard and Laurence, 2009). For example, Leschziner and Hogg (1989) found an RSM to be superior to the $k - \varepsilon$ model in highly swirling confined flows because it accounted for suppression of turbulence while the $k - \varepsilon$ generated too much dissipation, as observed here. It may also be worth implementing the $k - \omega$ SST model in Fluidity in order to take advantage of its improved behaviour in swirling flows over the $k - \varepsilon$ model.

8 Epilogue

Contents

8.1	Summary	266
8.2	Conclusions	268
8.2.1	Towards a Range of Robust Turbulence Models . . .	268
8.2.2	Efficient Use of Computational Power	271
8.2.3	Estimating and Providing Appropriate Resolution in LES	272
8.2.4	Model Validation in the Vortex Diode	273
8.3	Future Work	273
8.3.1	Revised Filter Width Definitions	273
8.3.2	Towards Complete LES Modelling	274

8.1 Summary

In Chapter 2, the use of CFD in the nuclear energy industry was discussed in order to give context to this research. The distinction between ‘conservative’ and ‘best-estimate’ models was made in the context of nuclear safety analysis and best-practice in performing industrial CFD studies was outlined. Costs and constraints associated with running a high-performance computing (HPC) facility were discussed.

Chapter 3 covered the fundamentals of turbulence, turbulence modelling, discretisation and adaptive mesh techniques. Numerically-derived stabilisation methods were compared to turbulence models derived from physical arguments.

In Chapter 4 the implementation of the standard $k - \varepsilon$ model by the control volume-based finite element (CV-FE) method was described. A law-of-the-wall turbulent boundary condition in variational form was developed for use in high-Reynolds-number flows. Numerical properties of the $k - \varepsilon$ model were verified. Its performance with various boundary conditions on fixed and adaptive meshes was validated using the 2D backward-facing step against experimental data.

Convergence was demonstrated on fixed meshes up to 44,000 nodes and on various adaptive meshes with 50-80% fewer nodes, suggesting that mesh adaptivity is an efficient alternative to conventional demonstrations of solution convergence by uniform grid-refinement. From five different interpolation error measures, the $k - \varepsilon$ model converged to a single mesh-independent solution. However, reattachment length and mean velocity profiles were not as accurately predicted as on a uniform fine mesh, possibly due to the log-law wall function being applied on too fine a mesh.

A scale-adaptive modified $k - \varepsilon$ model, the VLES model, was also implemented and validated using 2D backward-facing step. While the reattachment length predictions were not accurate, the flow dynamics displayed a split between large resolved scales and uniform RANS regions as expected of such a model. This ability is potentially useful but only in a ‘true’ i.e. 3D turbulent flow, evidenced by highly accurate preliminary 3D VLES results presented at the end of Chapter 6.

In Chapter 5 a tensorial dynamic LES model was developed in response to the deficiencies of the standard Smagorinsky LES model. The new model was designed specifically for use with unstructured mesh adaptivity and addressed the problems associated with LES on unstructured inhomogeneous meshes. Particular attention was given to filtering and the resulting commutation errors. Properties of the inverse Helmholtz filter, modified to accept a tensorial filter width and implemented in finite element form, were verified in

simple tests in order to justify its use in the dynamic LES model. The chapter concluded with a discussion of strategies for marrying LES to adaptive mesh techniques.

The tensorial dynamic LES model was validated on fixed and adaptive meshes using the 3D backward-facing step in Chapter 6. Results show that dynamic LES with mesh adaptivity was a highly effective combination, able to resolve the important flow details with minimal resolution and provide sufficient sub-filter-scale dissipation in response to the level of resolution. Comparable results were obtained on a fixed mesh with 600,000 nodes and adaptive meshes with 60% fewer nodes while only adapting every 100 to 200 timesteps, with each adapt costing approximately the same as one timestep. This demonstrated both the efficiency of mesh adaptivity and the reduction of required *a priori* knowledge of flow dynamics in order to design the mesh. Some of the interpolation error-based adaptivity strategies discussed in Chapter 5 were investigated and improvements in the results were obtained by changing the interpolation error measures used.

The original form of the tensorial filter width was found to be too dissipative (cf. §5.2.4) compared to a standard scalar definition, the cube root of element volume (5.7), so its width was reduced by a normalisation procedure (5.45). It was found that the normalised tensorial definition gave comparable, but not superior, results to the scalar filter on various meshes, perhaps because the definition of magnitude was inappropriate. To see if the tensor filter was superior on meshes with greater anisotropy, adaptive meshes with higher aspect ratios (A4 and A5) were tested but no significant differences were observed. The scalar filter is suitable for use on these meshes, giving support to the finding of Oshima et al. (1997) on Cartesian meshes.

Significant improvements were obtained by applying the Synthetic Eddy Method (SEM) at the inflow to seed turbulence. Dynamic LES results obtained with CFX were inferior to Fluidity because CFX lacked a comparable method to SEM, resulting in a quasi-laminar flow. Fluidity results without SEM were still better than CFX; in particular the reattachment length was predicted with far greater accuracy, though this may be due to cancellation of errors meaning that one cannot draw firm conclusions.

Chapter 7 compared Fluidity and CFX in a challenging industrial test case, the vortex diode. It was found that the prediction of pressure drop in a strong vortex was highly sensitive to the choice of turbulence model, discretisation and solution algorithm. Eddy-viscosity-based models were unable to make quantitative predictions of pressure drop because the exact pressure is unavailable in these models, although the $k - \omega$ SST model in CFX gave reasonable predictions. The potential of simulating the vortex diode forward flow without a turbulence model was demonstrated in CFX.

8.2 Conclusions

Three research challenges were presented in the introduction which face CFD for nuclear reactor safety analysis. These were to maximise computational efficiency, improve the range of validity or robustness of turbulence models and develop ways to demonstrate accuracy and convergence of LES solutions. Three research objectives were also stated: to use mesh adaptivity to efficiently simulate turbulent flows, to develop a range of robust turbulence models, and to validate the models in academic and industrial test cases. Research achievements are related to the three research challenges and three objectives here.

8.2.1 Towards a Range of Robust Turbulence Models

This thesis presents the development, verification and validation of several models for modelling turbulence with unstructured mesh adaptivity. These methods are the $k - \varepsilon$ U-RANS and VLES models discretised by the CV-FE method, a near-wall model in variational form and a tensorial dynamic LES model with a differential tensor filter. Robustness is defined here as the ability to be applied accurately in diverse situations without tuning of parameters. In other words, the model has to self-tune or self-adapt to the particular flow.

8.2.1.1 RANS model with adaptivity

The $k - \varepsilon$ model with unstructured mesh adaptivity was found to generate mesh-independent solutions when five different error measures were used to generate the mesh. The five converged meshes were quite similar, suggesting that the error measures were equivalent. Setting an error target in one (or a few) fields is equivalent to setting error targets in all of them because all the prognostic fields are coupled in this model. Solving the model equations followed by regenerating the mesh can be viewed as an iterative solution algorithm towards a converged solution with mesh regeneration as part of the model.

Although the $k - \varepsilon$ model is not a very accurate RANS model by itself, its accuracy is improved in combination with adaptive meshing, being able to supply optimal resolution for the flow tested. Separating and reattaching flow (i.e. the 2D backward-facing step) is a well-known case where the $k - \varepsilon$ model is inaccurate, making it difficult to assess the full potential of this combination. In future, either a more accurate RANS model such as the

$k - \omega$ SST model should be used with mesh adaptivity, or a test case in which the $k - \varepsilon$ model is accurate such as a 2D shear layer should be studied.

Reattachment length predictions on a fine fixed mesh were slightly closer to the benchmark data than on adaptive meshes, primarily due to the log-law wall function being applied too close to the wall in the latter, resulting in incorrect near-wall velocity profiles. Adaptive wall functions are needed to improve accuracy on meshes where near-wall resolution is variable.

8.2.1.2 Scale-adaptive VLES model

The VLES model was obtained by scaling the eddy viscosity in the $k - \varepsilon$ model by the relative size of the mesh Δ to the smallest flow scale, the Kolmogorov lengthscale η and the largest scale, the integral (model) lengthscale l_0 . The result was to produce a scale-sensitive model similar to detached eddy simulation (DES) in which a sharp transition was created between a full RANS region where viscosity was not scaled, and an LES-like region where it was.

In 2D validation tests the RANS region the flow was very uniform and steady-state, while the flow in the LES-like region contained large coherent – but non-turbulent – scales. With increasingly fine uniform fixed meshes, the large scales became more complex and finer scales were manifested. As expected for a scale-resolving model in 2D, reattachment length predictions were very poor, albeit consistent across different mesh resolutions.

In the 3D backward-facing step on a uniform fixed mesh of 600,000 nodes the flow was very accurately simulated in the shear layer and reattachment region behind the step. Reattachment length was slightly under-predicted, suggesting that too much dissipation is generated. By resolving only the large scales, the dissipative effect of the smaller scales on the large may be incorrectly accounted for by the model. Further research is needed to find out if the division of the turbulent spectrum by the VLES model is a valid representation of turbulence in terms of the coupling between resolved and modelled scales.

The flow was under-resolved near the bottom wall owing to the presence of a RANS region along the wall, which offers an alternative explanation of the low reattachment length: the large near-wall velocity gradients typical of the reattachment region were not accurately captured in the RANS region. A RANS region was also generated in the inlet and above the shear layer, in which the flow was accurately represented, although the much lower velocity gradients here made the job much easier than near the wall.

To save computational effort the VLES model could be used with mesh adaptivity. However, it may be that an adaptive model and an adaptive

mesh would try to perform the same task of concentrating on the significant dynamics, with neither achieving ideal results. On the other hand, it can be argued that the VLES model does not know which areas of the flow are dynamically significant, and needs a well-designed mesh in order to generate accurate results. Further demonstrations of the model's quantitative predictive capabilities on fixed homogeneous and inhomogeneous meshes are needed before attempting adaptive simulations.

The VLES model (or a similar scale-resolving method) may have considerable value in complex simulations if it can be demonstrated that it gives more detailed and accurate flow representations than RANS without going to the expense on LES. Its range of applicability is limited by the type of quantities that can be predicted accurately in the RANS and LES regions.

8.2.1.3 Tensorial Dynamic LES Model with Adaptivity

The novel combination of a tensorial eddy-viscosity LES model, based on a tensorial filter width designed for unstructured elements, with the dynamic method and unstructured mesh adaptivity has been introduced. It is a robust model for several reasons:

- SFS stress is tuned to local mesh resolution by means of the dynamic Smagorinsky coefficient,
- SFS stress is tuned in each direction independently by means of the tensorial filter width,
- the dynamic Smagorinsky coefficient does not need an *ad hoc* upper limit to stabilise the model (only $c_S \geq 0$ is imposed),
- averaging in the expression for c_S is not needed for stability, removing another *ad hoc* operation from the original dynamic model,
- mesh resolution is automatically adapted to resolve the flow within a specified discretisation error bound,
- no wall functions are necessary thanks to the dynamic method.

The only user-defined parameters in the model are the filter width ratios α and β and the mesh adaptivity settings. Although the aim was to develop a parameter-free model which would be truly robust, the filter width ratios may not need to be changed from the optimum values of $\alpha = 1$ and $\beta = 1$ found in the validation exercise. Additionally, the user expertise required to generate a good adaptive mesh is less than that required to generate a good

fixed mesh. Therefore, this model is an improvement on current methods for dynamic LES on fixed meshes. Robustness of the new model needs to be confirmed by simulating other flows.

In the validation tests it became apparent that the tensor filter was not superior to a simple scalar width definition. This finding is consistent with [Oshima et al. \(1997\)](#), who found that scalar and tensor filter widths gave comparable results on anisotropic structured (Cartesian) meshes. The extension of this finding to unstructured meshes is not illogical.

Furthermore, simple analysis (cf. §5.2.1) has shown that the tensor LES model is not invariant under simple transformations, rendering it inconsistent with the Navier-Stokes equations. A subject for further research could be the construction of a frame-invariant model incorporating the tensor filter width. If one cannot be found, it is probably not worth pursuing this line of enquiry, making a scalar filter width the preferred option.

One question regarding the tensorial filter width remains: can the definition be revised such that the normalisation procedure (5.45) is unnecessary, while maintaining or improving its performance in the dynamic LES model? The tensor width was scaled down to give it the same magnitude as the scalar filter, reducing the magnitude of the resultant SFS stress term. The measure of tensor magnitude used was the Frobenius norm, which may not be the most appropriate geometrical measure of the magnitude of an ellipsoid tensor. Alternative filter width definitions are discussed in §8.3.1.

8.2.2 Efficient Use of Computational Power

In the 2D and 3D simulations presented, savings of 50-80% in terms of reduced node numbers were achieved by employing mesh adaptivity, while producing comparable or better results. Whilst the overhead of mesh generation has not been rigorously quantified, it is certainly outweighed by the savings in computational time achieved by reduced mesh size. [Hiester et al. \(2011\)](#) found that one mesh adapt cost approximately one timestep in terms of computing time.

The exact overhead depends on the frequency of mesh adaptations. In the 3D validation tests, the best results were obtained by adapting the mesh every 200 timesteps to the errors in the time-averaged velocity and fluctuating velocity. By adapting to time-averaged velocity, meshes captured most of the turbulence in the separation region behind the step. Although the timescales on which the fluctuating velocity evolved was on the order of ten timesteps, adapting to the Reynolds stress every 200 timesteps did impact on overall accuracy: the mesh was considerably finer as a result. However, the exact distribution of mesh resolution was non-optimal for representing

the fluctuations from about ten timesteps after each adapt. Here the use of a robust LES model may have been important: where resolution was not fine enough to represent fine-scale velocity fluctuations, the model could provide an appropriate level of dissipation in their place.

The backward-facing step flow converges to a statistical steady state, making it amenable to the mixed average/fluctuating velocity adaptivity strategy. In other cases the flow evolution may be quite different and a new strategy might have to be found. The flow in a complex case such as a nuclear accident scenario may continually evolve from an initial event to a final accident. Here, an alternative adaptivity strategy may be needed, such as adapting to the instantaneous velocity field at frequent intervals. The adaptivity overhead will certainly be higher in this case. Different strategies for adapting the mesh may have to be developed for different classes of problem. Whether this requires less effort than designing fixed meshes is a matter for further investigation.

8.2.3 Estimating and Providing Appropriate Resolution in LES

Achieving sufficient resolution of the turbulence in the 3D backward-facing step necessitated trial-and-error tuning of the error-based adaptivity algorithm to the specific qualities of the flow. Without mesh adaptivity the difficulties are even greater. Repetition of simulations is the only way to iterate towards an ideal mesh, which is inefficient. Clearly, a more advanced technique is needed for estimating the required resolution.

A putative method, developed from a suggestion by Pope (2004), was described in Chapter 5 for determining whether resolution was fine enough to satisfy the rule of thumb that 80% of the total kinetic energy is resolved globally. A relatively simple technique incorporating interpolation error based and goal-based adaptivity was proposed. Although the method has not yet been implemented, it has the potential to address a major challenge facing LES, namely, how do we know how fine to make the mesh? The proposed method is an attractive first step towards more advanced methods for determining appropriate resolution. Future work will also look into pure goal-based adaptivity, where the goal is not a generic measure of turbulence resolution like the proportion of energy, but a quantity directly related to the purpose of the simulation. These ideas are explored in §8.3.2 below.

8.2.4 Model Validation in the Vortex Diode

The turbulence models used to simulate the vortex diode forward flow were inadequate as they did not account for the laminar nature of the vortex and were generally too dissipative. Simulations at low flow rates deserve further attention to establish if there is a change of flow regime from turbulent to laminar in the inlet and outlet ducts, and which models are valid at these flow rates. However, there is no experimental data against which to validate their behaviour; other VD geometries could be simulated instead.

Simulations without a model were moderately successful, as found by [Kulkarni et al. \(2008\)](#). Further research on the use of numerical stabilisation methods in this flow is needed, particularly if Fluidity is to be used for this problem without a model. For example, the Discontinuous Galerkin method or SUPG may be appropriate for this flow, adding sufficient stabilisation without overly damping vortex formation. A deeper analysis of vortex formation in CFD may provide clues as to what the best method is. Reynolds stress models (RSM) should be tested, given their superior abilities to eddy-viscosity models in strongly swirling flow.

Corrections to eddy-viscosity models to recover the true pressure are needed to achieve quantitative predictive capability. Greater resolution is also advisable for the vortex diode if quantitative predictions are to be made. Better mesh generation methods for complex internal geometry may be required in Fluidity if other industrial CFD problems are to be studied. Prismatic boundary layers may be useful to provide small near-wall resolution perpendicular to the wall while maintaining coarser spacing parallel to the wall. Further work on the use of adaptive meshing in this problem is recommended; a suitable choice of error measure might be the pressure in this case.

8.3 Future Work

8.3.1 Revised Filter Width Definitions

Despite its positive effect (a reduction of model dissipation), the tensor normalisation (5.45) is an undesirable *ad hoc* procedure. Instead, a more consistent scalar definition can be obtained from the tensor definition itself which has the same magnitude as the tensor. While this does not address the problem of the magnitude of the tensor filter width, it at least places the scalar and tensor definitions on an equal footing.

The metric \mathcal{M} is an ellipsoid with principal dimensions given by the eigenvalues λ_i of the metric and oriented according to the eigenvectors \mathbf{V} .

Its volume is given by

$$V_{\mathcal{M}} = \frac{4}{3}\pi\lambda_{\zeta}\lambda_{\eta}\lambda_{\xi}. \quad (8.1)$$

Anisotropy is measured by the ratios of the eigenvalues. A regular or isotropic tetrahedron has equal eigenvalues, and its metric is a sphere, with volume given by the formula $V = \frac{4}{3}\pi r^3$. A spherical metric results in isotropic viscosity. Therefore, isotropy in the viscosity tensor can be created by distorting the metric to a space where it is a sphere in a similar process to the merging of two metrics as described in §3.6.1. The spherical metric \mathcal{M}_{iso} has equal eigenvalues Λ_{iso} and its eigenvectors \mathbf{V}_{iso} are the identity matrix. The eigenvalues are scaled so that the sphere's volume is equal to the original metric:

$$\Lambda_{iso} = \begin{bmatrix} \lambda_{iso} & 0 & 0 \\ 0 & \lambda_{iso} & 0 \\ 0 & 0 & \lambda_{iso} \end{bmatrix}, \quad (8.2)$$

$$\lambda_{iso} = (\lambda_{\zeta}\lambda_{\eta}\lambda_{\xi})^{1/3}. \quad (8.3)$$

The scaled spherical metric \mathcal{M}_{iso} is constructed from its eigenvalues and eigenvectors:

$$\mathcal{M}_{iso} = \mathbf{V}_{iso}\Lambda_{iso}\mathbf{V}_{iso}. \quad (8.4)$$

Initial tests of this revised definition show that it generates too much dissipation. The optimal values of α and β with these new definitions are lower as a result; values of less than one are permissible with the differential filtering operation. Future research will examine this question in greater depth and look for other mathematically and physically valid scalar and tensor definitions.

8.3.2 Towards Complete LES Modelling

The positive results obtained with the dynamic LES model and mesh adaptivity have given confidence that this is a robust and efficient CFD model for diverse industrial fluid flow problems. The method is good at resolving turbulence in complex geometry, but in future a more complete strategy could be investigated, stemming from the discussion of LES with adaptivity in §5.4. There a putative improvement to the method was outlined based on measuring the proportions of SFS and resolved kinetic energy. This could indicate whether the solution is well-resolved.

However, it is not necessarily the case that sufficiently resolving all of the turbulent dynamics is a pre-requisite to obtaining a trustworthy solution. Only some areas of the flow are influential over the quantity which we actually want to predict. Goal-based adaptivity can be focussed on the pertinent

details of the problem, namely, what is it we are measuring and what are the parameters to which the measurement is sensitive? Future research could be directed to developing goal-based methods for complex high-Re turbulent flow. These methods have enormous potential in industrial CFD. The problems faced in their development may include:

1. What are the types of questions which industry want to answer?
2. What functionals are reliable indicators of solution accuracy in a particular problem?
3. On which scales of motion do particular functionals depend?
4. How do functionals vary in time in a turbulent flow? To optimise resolution to accurately predict a time-varying functional, how long should the interval between mesh adapts be?
5. How do the functionals depend on the choice of discretisation, turbulence model etc? i.e. are the measurements we are interested in robust to variations in the numerical method?
6. Which turbulence models are appropriate for computing functionals? (As opposed to ‘conventional’ turbulence modelling where common questions pertain to physical considerations such as correctly modelling energy flux, wall scaling etc.) For example, any turbulence model which can be expressed as a variational principle can be differentiated, and can therefore be incorporated in an adjoint model ([Farrell, 2012](#)).
7. What role could/should mesh anisotropy play in improving the accuracy of functional computations?

Bibliography

- Addad, Y., Laurence, D., Talotte, C., Jacob, M., 2003. Large eddy simulation of a forward-backward facing step for acoustic source identification. *International Journal of Heat and Fluid Flow* 24, 562–571. 129
- AIAA, 1998. Guide for the verification of computational fluid dynamics simulations. Tech. Rep. AIAA G-077-1998, AIAA. 194
- Aider, J.-L., Danet, A., 2006. Large-eddy simulation study of upstream boundary conditions influence upon a backward-facing step flow. *Comptes Rendus Mecanique* 334 (7), 447–453. 67, 195
- Alauzet, F., Shephard, M., Seol, E., Li, X., 2006. Parallel anisotropic 3D mesh adaptation by mesh modification. *Engineering with Computers* 21 (3), 247–258. 87
- AMCG, April 2012. Fluidity Manual. Applied Modelling & Computation Group (AMCG), Department of Earth Science and Engineering, Imperial College London, London, version 4.1.6 Edition.
URL <https://launchpadlibrarian.net/99636503/fluidity-manual-4.1.6.pdf> 74, 76, 80, 81, 105, 234
- Ameri, A., Bunker, R., 2000. Heat transfer and flow on the first-stage blade tip of a power generation gas turbine: Part 2 – simulation results. *Journal of Turbomachinery* 122 (2), 272–277. 68
- ANSYS, 2011. ANSYS CFX User Guide. ANSYS. 12, 98, 99, 195, 204, 234, 242, 243
- Aristodemou, E., Bentham, T., Pain, C., Colvile, R., Robins, A., Apsimon, H., 2009. A comparison of mesh-adaptive LES with wind tunnel data for flow past buildings: Mean flows and velocity fluctuations. *Atmospheric Environment* 43, 6238–6253. 139

BIBLIOGRAPHY

- Arnold, D., Brezzi, F., Cockburn, B., Marini, L., 2002. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM Journal on Numerical Analysis* 39 (5), 1749–1779. [85](#)
- Balaras, E., Benocci, C., Piomelli, U., 1995. Finite-difference computations of high Reynolds number flows using the dynamic subgrid-scale model. *Theoretical and Computational Fluid Dynamics* 7, 207–216. [146](#)
- Balaras, E., Benocci, C., Piomelli, U., 1996. Two-layer approximate boundary conditions for large-eddy simulations. *AIAA journal* 34 (6), 1111–1119. [141](#)
- Baldwin, B., Lomax, H., 1978. Thin-layer approximation and algebraic model for separated turbulent flows. In: 16th AIAA Aerospace Sciences Meeting. AIAA, pp. 1–8. [95](#)
- Bardina, J., Ferziger, J., Reynolds, W., 1980. Improved subgrid-scale models for large-eddy simulation. In: 13th AIAA Fluid and Plasma Dynamics Conference. Snowmass, Colorado. [136](#), [138](#), [139](#), [144](#)
- Bardina, J., Huang, P., Coakley, T., 1997. Turbulence modeling, validation, testing and development. Technical memorandum 110446, NASA. [99](#)
- Bazilevs, Y., Hughes, T., 2007. Weak imposition of Dirichlet boundary conditions in fluid mechanics. *Computers and Fluids* 36 (1), 12–26. [66](#), [75](#)
- Bazilevs, Y., Michler, C., Calo, V., Hughes, T., 2007. Weak Dirichlet boundary conditions for wall-bounded turbulent flows. *Computer Methods in Applied Mechanics and Engineering* 196 (49-52), 4853–4862. [75](#), [101](#)
- Becker, R., Rannacher, R., 2001. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica* 10, 1–102. [90](#), [182](#), [186](#)
- Belme, A., Dervieux, A., Alauzet, F., 2012. Time accurate anisotropic goal-oriented mesh adaptation for unsteady flows. *Journal of Computational Physics* 231 (19), 6323–6348. [186](#)
- Benhamadouche, S., Laurence, D., 2003. LES, coarse LES, and transient RANS comparisons on the flow across a tube bundle. *International Journal of Heat and Fluid Flow* 24, 470–479. [26](#)
- Bentham, T., 2003. Microscale modelling of air flow and pollutant dispersion in the urban environment. Ph.D. thesis, Imperial College London. [12](#), [14](#), [135](#), [137](#), [138](#), [140](#), [142](#), [143](#), [150](#), [180](#), [212](#)

BIBLIOGRAPHY

- Berselli, L. C., Ilescu, T., Layton, W. J., 2006. Mathematics of Large Eddy Simulation of Turbulent Flows. Springer. [61](#), [132](#), [134](#), [141](#)
- Bestion, D., Martin, A., Menter, F., 2004. Recommendation on use of CFD codes for nuclear reactor safety analysis. Technical memorandum EVOL-ECORA – D14, ECORA. [8](#)
- Bieder, U., Fauchet, G., Béтин, S., Kolev, N., Popov, D., 2007. Simulation of mixing effects in a VVER-1000 reactor. Nuclear Engineering and Design [237](#), 1718–1728. [9](#), [30](#)
- Bischof, C., an Mey, D., Iwainsky, C., 2011. Brainware for green HPC. Computer Science – Research and Development [27](#) (4), 227–233. [41](#)
- Boris, J., Grinstein, F., Oran, E., Kolbe, R., 1992. New insights into large eddy simulation. Fluid Dynamics Research [10](#). [85](#)
- Bradshaw, P., 1973. Effects of streamline curvature on turbulent flow. In: AGARDograph 169. AGARD, Paris. [58](#)
- Bredburg, J., 2000. On the wall boundary condition for turbulence models. internal report 00/4, Department of Thermo and Fluid Dynamics, Chalmers University of Technology, Göteborg, Sweden. [66](#)
- Brenner, S., Scott, L., 2002. The mathematical theory of finite element methods, 2nd Edition. Springer. [87](#)
- Brezzi, F., Franca, L., Russo, A., 1998. Further considerations on residual-free bubbles for advective-diffusive equations. Computer Methods in Applied Mechanics and Engineering [166](#), 25–33. [84](#)
- Bull, J., Piggott, M., Pain, C., 2012. A finite element LES methodology for anisotropic inhomogeneous meshes. In: Hanjalić, K., Nagano, Y., Borello, D., Jakirlić, S. (Eds.), Turbulence, Heat and Mass Transfer [7](#). pp. 317–320. [127](#), [191](#)
- Burbidge, S., 2011. Management of Imperial College London HPC facilities, personal communication. [41](#), [42](#)
- Burgess, D., 1972. Analysis of data from vortex diode experiments. RRA3144, Rolls-Royce Submarines. [238](#)
- Burgess, D., Tough, W., 1971. Experimental investigation of flow through a vortex diode. Technical memorandum RRA2764, Rolls-Royce Submarines. [238](#), [239](#), [240](#)

BIBLIOGRAPHY

- Candy, A., 2008. Subgrid scale modelling of transport processes. Ph.D. thesis, Imperial College London. 84, 114, 140, 183
- Cant, S., April 2002. High-performance computing in computational fluid dynamics: progress and challenges. *Philosophical Transactions A: Royal Society London* 360, 1211–1225. 135
- Carati, D., Cabot, W., 1996. Anisotropic eddy viscosity models. In: *Proceedings of the summer program 1996*. Center for Turbulence Research, pp. 249–258. 12, 138
- Carati, D., van den Eijnden, E., 1997. On the self-similarity assumption in dynamic models for large eddy simulations. *Physics of Fluids* 9 (2165). 148
- Carati, D., Winckelmans, G. S., Jeanmart, H., 2001. On the modelling of the subgrid-scale and filtered-scale stress tensors in large-eddy simulation. *Journal of Fluid Mechanics* 441, 119–138. 147
- Çengel, Y., Turner, R., 2001. *Fundamentals of Thermal-Fluid Sciences*, international edition Edition. McGraw-Hill. 237
- Cebeci, T., Smith, A., 1974. *Analysis of turbulent boundary layers*. NASA STI/Recon Technical Report A 75. 95
- Cenedese, A., Romano, G., 1991. Experimental testing of Taylor’s hypothesis by L.D.A. in highly turbulent flow. *Experiments in Fluids* 11, 351–358. 55
- Cerutti, S., Meneveau, C., Knio, O., 2000. Spectral and hyper eddy viscosity in high-Reynolds-number turbulence. *Journal of Fluid Mechanics* 421, 307–338. 140
- Chabard, J., Laurence, D., 2009. Heat and fluid flow simulations for deciding tomorrow’s energies. In: *6th International Symposium on Turbulence, Heat and Mass Transfer (THMT6)*. 17, 36, 130, 264
- Chambers, E., 1728. *Cyclopaedia, or, An Universal Dictionary of Arts and Sciences: Containing the Definitions of the Terms, and Accounts of the Things Signify’d Thereby, in the Several Arts, both Liberal and Mechanical, and the Several Sciences, Human and Divine: the Figures, Kinds, Properties, Productions, Preparations, and Uses, of Things Natural and Artificial; the Rise, Progress, and State of Things Ecclesiastical, Civil, Military, and Commercial: with the Several Systems, Sects, Opinions, etc; among Philosophers, Divines, Mathematicians, Physicians, Antiquaries, Criticks, etc.: The Whole Intended as a Course of Ancient and Modern Learning*. Chambers. iii

BIBLIOGRAPHY

- Chen, Q., Otte, M., Sullivan, P., Tong, C., 2008. A posteriori subgrid-scale model tests based on the conditional means of subgrid-scale stress and its production rate. In: 18th Symposium on Boundary Layers and Turbulence. 134
- Ciarlet, P., Lions, J., 2000. Handbook of numerical analysis volume 7. Elsevier. 106
- Clark, R., Ferziger, J., Reynolds, W., 1979. Evaluation of subgrid-scale models using an accurately simulated turbulent flow. *Journal of Fluid Mechanics* 91 (1), 1–16. 58, 139, 147
- Codina, R., Blasco, J., 2002. Analysis of a stabilized finite element approximation of the transient convection-diffusion-reaction equation using orthogonal subscales. *Computing and Visualization in Science* 4, 167–174. 84
- Dantinne, G., Jeanmart, H., Winckelmans, G. S., Legat, V., 1997. Hyper viscosity and vorticity-based models for subgrid scale modelling. *Applied Scientific Research* 59 (4), 409–420. 139
- Darlington, R., McAbee, T., Rodrigue, G., 2002. Large eddy simulation and ALE mesh motion in Rayleigh–Taylor instability simulation. *Computer Physics Communications* 144, 261–276. 135
- Davidson, L., 2009. Large eddy simulations: how to evaluate resolution. *International Journal of Heat and Fluid Flow* 30 (5), 1016–1025. 188
- Davidson, L., 2011. How to estimate the resolution of an LES of recirculating flow. In: *Quality and Reliability of Large-Eddy Simulations II*. Vol. 16 of ERCOFTAC series. Springer, pp. 269–286. 130, 183, 186, 190
- Deardorff, J., 1970. A numerical study of three-dimensional turbulent channel flow at large Reynolds numbers. *Journal of Fluid Mechanics* 41 (2), 453–480. 101, 136, 147
- Diurno, G., Balaras, E., Piomelli, U., 2001. Wall-layer models for LES of separated flows. *Modern Simulation Strategies for Turbulent Flows*, 207–222. 142, 194
- Donea, J., Huerta, A., 2003. *Finite element methods for flow problems*. John Wiley & Sons, Ltd. 69, 71, 74, 77, 81, 82, 83, 104

BIBLIOGRAPHY

- Dongarra, J., Beckman, P., 2011. The international exascale software roadmap. *International Journal of High Performance Computer Applications* 25 (1). 38
- Dunca, A., John, V., 2003. The commutation error of the space averaged Navier-Stokes equations on a bounded domain. *Advances in Mathematical Fluid Mechanics*, 53–78. 155, 159, 163, 164
- Dunca, A., John, V., 2004. Finite element error analysis of space averaged flow fields defined by a differential filter. *Mathematical Models and Methods in Applied Sciences* 14 (4), 603–618. 163, 164
- Edwards, M., 2002. Unstructured, control-volume distributed, full-tensor finite-volume schemes with flow based grids. *Computational Geosciences* 6, 433–452. 79
- Ercan, Erturk, 2008. Numerical solutions of 2-D steady incompressible flow over a backward-facing step, part I: High Reynolds number solutions. *Computers and Fluids* 37 (6), 633–655. 113
- ERCOFTAC, 2012. ERCOFTAC “classic collection” database.
URL <http://cfd.mace.manchester.ac.uk/ercoftac/> 200
- Eriksson, L., Johansson, E., Kettaneh-Wold, N., Wikström, C., Wold, S., 2008. *Design of Experiments: Principles and Applications*, 3rd Edition. Umetrics Academy. 37
- Fang, F., Piggott, M., Pain, C., Gorman, G., Goddard, A., 2006. An adaptive mesh adjoint data assimilation method. *Ocean Modelling* 15, 39–55. 91
- Farrell, P., 2009. Galerkin projection of discrete fields via supermesh construction. Ph.D. thesis, Imperial College London Dept. of Earth Science and Engineering. 87, 89, 90, 91, 163
- Farrell, P., 2012. Strategies for mesh adaptivity, personal communication. 183, 185, 275
- Farrell, P., Piggott, M., Gorman, G., Ham, D., Wilson, C., Bond, T., 2011. Automated continuous verification for numerical simulation. *Geoscientific Model Development* 4, 435–449. 34
- Farrell, P., Piggott, M., Gorman, G., Pain, C., Wilson, C., 2009. Conservative interpolation between unstructured meshes via supermesh construction. *Computer Methods in Applied Mechanics and Engineering* 198, 2632–2642. 109

BIBLIOGRAPHY

- Farrell, P., Piggott, M., Gorman, G., Wilson, C., Ham, D., 2010. Automated continuous verification and validation for numerical simulation. In: Geoscientific Model Development Discussions. Vol. 3. pp. 1587–1623. [34](#), [129](#)
- Ferziger, J., Peric, M., Morton, K., 1999. Computational methods for fluid dynamics. Springer Berlin. [32](#), [59](#), [76](#), [77](#), [79](#), [97](#), [101](#), [103](#), [108](#), [131](#)
- Fischer, P., Lottes, J., Pointer, D., Siegel, A., 2008. Petascale algorithms for reactor hydrodynamics. Journal of Physics: Conference Series 125 (012076). [17](#), [41](#), [59](#)
- Ford, R., Pain, C., Piggott, M., Goddard, A., Umpleby, A., De Oliveira, C., 2004. A nonhydrostatic finite-element model for three-dimensional stratified oceanic flows. part 1: Model formulation. Monthly Weather Review 132, 2816–2831. [103](#)
- Friess, C., Manceau, R., 2012. Investigation of the equivalence of two hybrid temporal-LES methods based on elliptic blending in separated flows. In: Hanjalić, K., Nagano, Y., Borello, D., Jakirlić, S. (Eds.), Turbulence, Heat and Mass Transfer 7. [9](#), [62](#)
- Frisch, U., 1995. Turbulence: the legacy of A. N. Kolmogorov. Cambridge Univ Press. [52](#)
- Fröhlich, J., Rodi, W., Kessler, P., Parpais, S., Bertoglio, J., Laurence, D., 1998. Large eddy simulation of flow around circular cylinders on structured and unstructured grids. Notes on Numerical Fluid Mechanics 66, 319–338. [11](#)
- Fröhlich, J., von Terzi, D., 2008. Hybrid LES/RANS methods for the simulation of turbulent flows. Progress in Aerospace Sciences 44 (5), 349–377. [62](#), [63](#), [125](#), [141](#)
- Fureby, C., Grinstein, F., 2002. Large eddy simulation of high-Reynolds-number free and wall-bounded flows. Journal of Computational Physics 181, 68–97. [10](#), [137](#)
- Fureby, C., Tabor, G., 1997. Mathematical and physical constraints on large eddy simulations. Theoretical and Computational Fluid Dynamics 9, 85–102. [129](#), [159](#)
- Gallerano, F., Pasero, E., Cannata, G., 2005. A dynamic two-equation sub grid scale model. Continuum Mechanics and Thermodynamics 17 (2), 101–123. [12](#), [134](#), [135](#), [138](#), [141](#), [189](#)

BIBLIOGRAPHY

- Garmann, D., Visbal, M., Orkwis, P., 2012. Comparative study of implicit and subgrid-scale model large-eddy simulation techniques for low-Reynolds number airfoil applications. *International Journal for Numerical Methods in Fluids* online only. [150](#)
- Garnier, E., Mossi, M., Sagaut, P., Comte, P., Deville, M., 1999. On the use of shock-capturing schemes for large-eddy simulation. *Journal of Computational Physics* 153 (2), 273–311. [85](#)
- Gaston, L., 1997. Simulation numérique par éléments finis bidimensionnels du remplissage de moules de fonderie et étude expérimentale sur maquette hydraulique. Ph.D. thesis, Ecole Nationale Supérieure des Mines de Paris. [107](#)
- Gavrilas, M., Kiger, K., 2001. ISP-43: Rapid boron dilution transient experiment comparison report. Tech. Rep. NEA/CSNI/R(2000)22, OECD-Nuclear Energy Agency. [26](#)
- George, W., Hussein, H., 1991. Locally axisymmetric turbulence. *Journal of Fluid Mechanics* 233, 1–23. [50](#)
- Georgiadis, N., Fureby, C., Rizzetti, D., 2009. Large-eddy simulation: Current capabilities, recommended practices, and future research. In: 47th AIAA Aerospace Sciences Meeting. AIAA. [141](#)
- Germano, M., 1986a. Differential filters for the large eddy numerical simulation of turbulent flows. *Physics of Fluids* 29 (6), 1755–1756. [142](#), [155](#)
- Germano, M., 1986b. Differential filters of elliptic type. *Physics of Fluids* 29 (6), 1757–1758. [160](#)
- Germano, M., 2001. LES overview. In: Liu, C., Sakell, L., Beutner, T. (Eds.), *DNS/LES Progress and Challenges: Third AFOSR International Conference*. [53](#)
- Germano, M., Piomelli, U., Moin, P., Cabot, W. H., 1991. A dynamic subgrid-scale eddy viscosity model. *Physics of Fluids* 3 (7), 1760–1765. [12](#), [133](#), [135](#), [138](#), [141](#), [144](#), [145](#), [146](#), [147](#), [149](#), [153](#), [194](#)
- Geurts, B. J., Fröhlich, J., 2001. Numerical effects contaminating LES: a mixed story. In: Geurts, B. J. (Ed.), *Modern Simulation Strategies for Turbulent Flow*. Edwards. [131](#), [132](#), [180](#)
- Geurts, B. J., Holm, D. D., 2004. Alpha-modelling strategy for les of turbulent mixing. *Turbulent Flow Computation* 66, 237–278. [132](#), [155](#)

BIBLIOGRAPHY

- Geurts, B. J., Leonard, A., 2000. Is LES ready for complex flows? In: Launder, B., Sandham, N. (Eds.), *Strategies for Turbulent and Transitional Flows*. Cambridge Univ Pr. 159
- Geuzaine, C., Remacle, J.-F., 2009. *Gmsh reference manual*. Montefiore Institute, University of Liège Dept. of Electrical Engineering and Computer Science, Liège, Belgium.
URL www.geuz.org/gmsh/ 86
- Ghosal, S., Lund, T. S., Moin, P., 1995. A dynamic localization model for large-eddy simulation of turbulent flows. *Journal of Fluid Mechanics* 286, 229–255. 135, 141, 145, 147, 189
- GiD, 2012. *GiD reference manual*. GiD.
URL www.gidhome.com/support/manuals 86
- Göddecke, D., Buijssen, S., Wobker, H., Turek, S., 2009. GPU acceleration of an unmodified parallel finite element Navier–Stokes solver. In: *International Conference on High Performance Computing and Simulation*. pp. 12–21. 41
- Gresho, P., Chan, S., 1988. Solving the incompressible Navier-Stokes equations using consistent mass and a pressure poisson equation. In: *ASME Symposium on Recent Development in CFD*. Chicago, pp. 51–73. 78
- Gropp, W., Kaushik, D., Keyes, D., Smith, B., 2001. Latency, bandwidth, and concurrent issue limitations in high-performance CFD. *Computational Fluid and Solid Mechanics*, 839–841. 40
- Grotjans, H., Menter, F., 1998. Wall function for general application CFD codes. In: *Computational fluid dynamics '98: Proceedings of the 4th European Computational Fluid Dynamics Conference*. ECCOMAS. 101
- Grötzbach, G., 1981. Numerical simulation of turbulent temperature fluctuations in liquid metals. *International Journal of Heat and Mass Transfer* 24, 475–490. 66
- Guermond, J.-L., Oden, J. T., Prudhomme, S., 2004. Mathematical perspectives on large eddy simulation models for turbulent flows. *Journal of Mathematical Fluid Mechanics* 6, 194–248. 160
- Gullbrand, J., 2003a. The effect of numerical errors and turbulence models in large-eddy simulations of channel flow, with and without explicit filtering. *Journal of Fluid Mechanics* 495, 323–341. 132

BIBLIOGRAPHY

- Gullbrand, J., 2003b. Grid-independent large-eddy simulation in turbulent channel flow using three-dimensional explicit filtering. In: Annual Research Briefs 2003. Center for Turbulence Research, pp. 331–342. [160](#)
- Guo, X., Gorman, G., Kramer, S., Piggott, M., Sunderland, A., 2010. High performance computing driven software development for next-generation modelling of the worlds oceans. Tech. rep., Distributed Computational Science and Engineering Support, Numerical Algorithms Group.
URL www.hector.ac.uk/cse/distributedcse/reports/fluidity-icom01/ [40](#)
- Hachem, M., 2009. Stabilized finite element method for heat transfer and turbulent flows inside industrial furnaces. Ph.D. thesis, Ecole Nationale Supérieure des Mines de Paris. [104](#), [105](#), [107](#), [108](#), [117](#), [118](#)
- Han, X., Krajnović, S., 2012. An efficient very large eddy simulation model for simulation of turbulent flow. International Journal for Numerical Methods in Fluids unpublished. [14](#), [63](#), [109](#), [110](#), [125](#)
- Hartmann, R., Held, J., Leicht, T., 2011. Adjoint-based error estimation and adaptive mesh refinement for the RANS and k- ω turbulence model equations. Journal of Computational Physics 230, 4268–4284. [186](#)
- Heinrichs, W., 1993. Spectral multigrid techniques for the Navier-Stokes equations. Computer Methods in Applied Mechanics and Engineering 106, 297–314. [243](#)
- Henson, V., Yang, U., 2002. BoomerAMG: a parallel algebraic multigrid solver and preconditioner. Applied Numerical Mathematics 41, 155–177. [107](#)
- Hewitt, G., 2010. Ntec module: Water reactor performance and safety lecture 1: Introduction to water reactors, lecture notes. [18](#), [20](#)
- Hickel, S., Cottet, G.-H., Weynans, L., Adams, N., 2007. Towards implicit subgrid-scale modeling by particle methods. In: ESAIM: proceedings. No. 16. pp. 77–88. [85](#), [131](#)
- Hieber, H., Piggott, M., Allison, P., 2011. The impact of mesh adaptivity on the gravity current front speed in a two-dimensional lock-exchange. Ocean Modelling 38, 1–21. [109](#), [135](#), [185](#), [271](#)
- Hill, J., Piggott, M., Ham, D., Popova, E., Srokosz, M., 2012. On the performance of a generic length scale turbulence model within an adaptive finite element ocean model. Ocean Modelling 56, 1–15. [90](#), [109](#)

BIBLIOGRAPHY

- Hoffman, J., Johnson, C., 2006. Computational turbulent incompressible flow. Springer. 51
- Höhne, T., 412 2007. Numerical simulation of coolant mixing in a pressurised water reactor with different CFD methods based on complex meshes. International Journal of Nuclear Energy Science and Technology 3 (4), 399. 32
- Höhne, T., Kliem, S., Bieder, U., 2006. Modeling of a buoyancy-driven flow experiment at the ROCOM test facility using the CFD codes CFX-5 and Trio-U. Nuclear Engineering and Design 236, 1309–1325. 8, 26, 30, 33
- Holmes, P., Berkooz, G., Lumley, J. L., 1993. The proper orthogonal decomposition in the analysis of turbulent flows. Annual Review of Fluid Mechanics 25, 539–575. 64
- Horiuti, K., 1989. The role of the Bardina model in large eddy simulation of turbulent channel flow. Physics of Fluids A: Fluid Dynamics 1, 426–428. 134
- Horiuti, K., 2003. Roles of non-aligned eigenvectors of strain-rate and subgrid-scale stress tensors in turbulence generation. Journal of Fluid Mechanics 491, 65–100. 139
- Howes, G., 2010. Parallel performance and optimization. Lecture notes. 40
- HPCAC, 2009. Interconnect analysis: 10GigE and InfiniBand in high performance computing. Tech. rep., HPC Advisory Council.
URL www.hpcadvisorycouncil.com/ 40
- HSE, 2006. Safety assessment principles for nuclear facilities. Regulatory guidelines, Health and Safety Executive, UK Government. 7
- Hughes, T., 1995. Multiscale phenomena: Green’s functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods. Computer Methods in Applied Mechanics and Engineering 127 (1-4), 387–401. 84
- Hughes, T., Brooks, A., 1982. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. Computer Methods in Applied Mechanics and Engineering 32, 199–259. 83

BIBLIOGRAPHY

- Hughes, T., Brooks, A., Liu, W., 1979. Finite element analysis of incompressible viscous flows by the penalty function formulation. *Journal of Computational Physics* 30 (1), 1–60. [83](#)
- Hughes, T., Mazzei, L., Jansen, K., 2000. Large eddy simulation and the variational multiscale method. *Computing and visualization in science* 3, 47–59. [84](#)
- Hughes, T., Mazzei, L., Oberai, A., Wray, A. A., 2001. The multiscale formulation of large eddy simulation: decay of homogeneous isotropic turbulence. *Physics of Fluids* 13 (2), 505–512. [84](#)
- Hutton, A., Szczypura, R., 1987. turbulent flow and heat transfer in a sudden pipe expansion. In: *Proceedings of the IAHR congress*. [101](#)
- IAEA, 2003. Use of computational fluid dynamics codes for safety analysis of nuclear reactor systems. Tech. Rep. IAEA-TECDOC-1379, OECD-Nuclear Energy Agency and International Atomic Energy Agency. [22](#), [25](#), [27](#), [28](#)
- IAEA, 2009. Deterministic safety analysis for nuclear power plants. Tech. Rep. Specific Safety Guide No. SSG-2, OECD-Nuclear Energy Agency and International Atomic Energy Agency. [23](#), [24](#), [25](#)
- Ignat, L., Pelletier, D., Ilinca, F., 1998. Adaptive computations of turbulent forced convection. *Numerical Heat Transfer, Part A: Applications* 34 (8), 847–871. [101](#)
- Ilinca, F., Pelletier, D., 1997. An adaptive finite element method for a two-equation turbulence model in wall-bounded flows. *International Journal for Numerical Methods in Fluids* 24, 101–120. [xvii](#), [94](#), [104](#), [108](#), [113](#), [114](#), [115](#), [116](#), [117](#), [120](#), [122](#), [124](#)
- Ioveno, M., Tordella, D., 2003. Variable scale filtered Navier–Stokes equations: A new procedure to deal with the associated commutation error. *Physics of Fluids* 15 (7), 1926–1936. [164](#)
- Ioveno, M., Tordella, D., Passoni, G., 2004. A new large-eddy simulation near-wall treatment. *Physics of Fluids* 16 (11), 3935–3944. [160](#)
- Jameson, A., 1991. Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings. Technical memorandum 91-1596, AIAA. [244](#)

BIBLIOGRAPHY

- Jansen, K., 1999. A stabilized finite element method for computing turbulence. *Computer Methods in Applied Mechanics and Engineering* 174 (3), 299–317. [164](#), [165](#)
- Jarrin, N., Benhamadouche, S., Laurence, D., Prosser, R., 2006. A synthetic-eddy-method for generating inflow conditions for large-eddy simulations. *International Journal of Heat and Fluid Flow* 27 (4), 585–593. [13](#), [67](#), [68](#), [142](#), [195](#)
- Jarrin, N., Prosser, R., Uribe, J.-C., Benhamadouche, S., Laurence, D., 2009. Reconstruction of turbulent fluctuations for hybrid RANS/LES simulations using a synthetic-eddy method. *International Journal of Heat and Fluid Flow* 30 (3), 435–442. [129](#)
- John, V., 2006. On large eddy simulation and variational multiscale methods in the numerical simulation of turbulent incompressible flows. *Applications of Mathematics* 51 (4), 321–353. [153](#)
- Johnstone, R. and Coleman, G. N., Spalart, P., 2010. The resilience of the logarithmic law to pressure gradients: evidence from direct numerical simulation. *Journal of Fluid Mechanics* 643, 163–175. [194](#)
- Jovic, S., Driver, D., 1994. Backward-facing step measurements at low Reynolds number, $Re=5000$. Technical memorandum 108807, NASA. [195](#), [223](#)
- KAERI, 2000. Nuclear power reactor technology lecture notes, module 2: Reactor core and components.
URL <http://www.kntc.re.kr/openlec/nuc/NPRT/> [19](#)
- Kaneda, Y., Leslie, D., 1983. Tests of subgrid models in the nearwall region using represented velocity fields. *Journal of Fluid Mechanics* 132, 349–373. [140](#)
- Kiltie, I., 2011. Rolls-Royce Submarines design principles, personal communication. [23](#), [25](#), [30](#), [32](#)
- Kim, J., 1978. Investigation of separation and reattachment of turbulence shear layer: flow over a backward facing step. Ph.D. thesis, Stanford University. [113](#), [117](#)
- Kim, J., Moin, P., Moser, R., 1987. Turbulence statistics in fully developed channel flow at low Reynolds number. *Journal of Fluid Mechanics* 177, 133–166. [115](#)

BIBLIOGRAPHY

- Kim, J.-Y., Ghajar, A. J., Tang, C., Foutch, G. L., Oct. 2005. Comparison of near-wall treatment methods for high Reynolds number backward-facing step flow. *International Journal of Computational Fluid Dynamics* 19, 493–500. [194](#)
- Klein, M., 2005. An attempt to assess the quality of large eddy simulations in the context of implicit filtering. *Flow, Turbulence and Combustion* 75, 131–147. [10](#), [34](#), [130](#), [131](#), [189](#)
- Klein, M., Sadiki, A., Janicka, J., 2003. A digital filter based generation of inflow data for spatially developing direct numerical or large eddy simulations. *Journal of Computational Physics* 183, 652–665. [67](#), [68](#)
- Kliem, S., Rohde, U., Höhne, T., Weiss, F.-P., 1999. Main steam line break analysis of a VVER-440 reactor using the coupled thermohydraulics system/3D-neutron kinetics code DYN3D/ATHLET in combination with the CFD code CFX-4. In: *Ninth International Topical Meeting on Nuclear Reactor Thermal Hydraulics (NURETH-9)*. San Francisco, California. [26](#), [29](#)
- Knopp, T., Zhang, X., Kessler, R., Lube, G., 2010. Enhancement of an industrial finite-volume code for large-eddy-type simulation of incompressible high Reynolds number flow using near-wall modelling. *Computer Methods in Applied Mechanics and Engineering* 199 (13-16), 890–902. [130](#), [188](#), [189](#), [194](#)
- Kollmann, W., McCallen, R., Leone Jr., J., 2002. An examination of LES filtering within the finite element method. *Communications in Numerical Methods in Engineering* 18, 512–528. [160](#)
- Kolmogorov, A., 1941. The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers. *Doklady Akademiia Nauk SSSR* 30 (301-305). [48](#), [95](#)
- Kuczaj, A., Komen, E., Loginov, M., 2010. Large-eddy simulation study of turbulent mixing in a T-junction. *Nuclear Engineering and Design* 240, 2116–2122. [9](#), [51](#), [129](#)
- Kulkarni, A., Ranade, V., Rajeev, R., Koganti, S., 2008. CFD simulation of flow in vortex diodes. *American Institute of Chemical Engineers Journal* 54 (5), 1139–1152. [235](#), [236](#), [237](#), [238](#), [242](#), [243](#), [245](#), [273](#)

BIBLIOGRAPHY

- Kwak, D., Reynolds, W., Ferziger, J., 1975. Three dimensional time-dependent computation of turbulent flow. Tech. Rep. TF-5, Thermosciences Div., Stanford University, Dept. Mech. Eng., Stanford, CA, USA. [147](#)
- Lacasse, D., Turgeon, E., Pelletier, D., 2004. On the judicious use of the k-epsilon model, wall functions and adaptivity. *International Journal of Thermal Sciences* 43 (10), 925–938. [97](#), [101](#), [105](#), [116](#)
- Langford, J., Moser, R., 2001. Breakdown of continuity in large-eddy simulation. *Physics of Fluids* 13 (5), 1524–1527. [159](#)
- Launder, B., Spalding, D., 1972. *Mathematical models of turbulence*. Academic Press. [100](#)
- Launder, B., Spalding, D., 1974. The numerical computation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering* 3 (2), 269–289. [97](#), [103](#), [104](#)
- Laurence, D., 2013. Personal communication. [81](#), [150](#)
- Laurence, D., Uribe, J., Utyuzhnikov, S., 2005. A robust formulation of the v2-f model. *Flow, Turbulence and Combustion* 73, 169–185. [141](#)
- Layton, W., 1999. Weak imposition of “no-slip” conditions in finite element methods. *Computers and Mathematics with Applications* 38 (5-6), 129–142. [75](#)
- Le, H., Moin, P., 1992. Direct numerical simulation of turbulent flow over a backward-facing step. In: *Annual Research Briefs*. Center for Turbulence Research. [200](#), [201](#)
- Le, H., Moin, P., Kim, J., 1997. Direct numerical simulation of turbulent flow over a backward-facing step. *Journal of Fluid Mechanics* 330, 349–374. [14](#), [129](#), [193](#), [195](#), [196](#), [201](#), [204](#), [214](#), [215](#), [221](#)
- Lee, S., Lele, S., Moin, P., 1992. Simulation of spatially evolving turbulence and the applicability of Taylor’s hypothesis in compressible flow. *Physics of Fluids* 4 (7), 1521–1530. [67](#)
- Leonard, A., 1974. Energy cascade in large eddy simulation of turbulent fluid flow. *Advances in Geophysics* 18A, 237–248. [56](#), [150](#)
- Leonard, S., Terracol, M., Sagaut, P., 2006. A wavelet-based adaptive mesh refinement method for large eddy simulation. *Journal of Turbulence* 7. [184](#)

BIBLIOGRAPHY

- Leschziner, M., 1989. Modeling turbulent reirculating flows by finite-volume methods - current status and future directions. *International Journal of Heat and Fluid Flow* 10 (3), 186–202. [104](#)
- Leschziner, M., Hogg, S., 1989. Computation of highly swirling confined flow with a Reynolds stress turbulence model. *AIAA Journal* 27 (1), 57–63. [264](#)
- Leveque, R., 2002. Finite-volume methods for hyperbolic problems. Cambridge Univ Press. [108](#)
- Lew, A., Buscaglia, G., Carrica, P., 2001. A note on the numerical treatment of the k-epsilon turbulence model. *International Journal of Computational Fluid Dynamics* 14 (3), 201–209. [97](#), [104](#), [113](#)
- Lilly, D. K., 1967. the representation of small-scale turbulence in numerical simulation experiments. In: *Proceedings of the IBM scientific computing symposium on environmental sciences*. [133](#), [147](#)
- Lilly, D. K., March 1992. A proposed modification of the Germano subgrid-scale closure method. *Physics of Fluids* 4 (3), 633–635. [135](#), [146](#), [147](#), [151](#), [230](#)
- Lin, C., Ebadian, M., 1999. The effects of inlet turbulence on the development of fluid flow and heat transfer in a helically coiled pipe. *International Journal of Heat and Mass Transfer* 42, 739–751. [79](#), [105](#)
- Liu, S., Katz, J., Meneveau, C., 1994. On the properties of similarity subgrid-scale models as deduced from measurements in a turbulent jet. *Journal of Fluid Mechanics* 275, 83–119. [58](#), [150](#)
- Liu, S., Katz, J., Meneveau, C., 1999. Evolution and modelling of subgrid scales during rapid straining of turbulence. *Journal of Fluid Mechanics* 387, 281–320. [134](#)
- Love, M., 1980. Subgrid modelling studies with Burgers' equation. *Journal of Fluid Mechanics* 100, 87–110. [154](#)
- Lund, T. S., Novikov, E., 1992. Parameterization of subgrid-scale stree by the velocity gradient tensor. In: *Annual Research Briefs 1992*. Center for Turbulence Research. [139](#), [146](#)
- Lund, T. S., Wu, X., Squires, K., 1998. Generation of turbulent inflow data for spatially-developing boundary layer simulations. *Physics of Fluids* 140, 233–238. [68](#)

BIBLIOGRAPHY

- Macek, J., Kral, P., Mühlbauer, P., 1997. Thermal hydraulic analyses of NPPs with VVER-440/213 for the PTS condition evaluation. In: NURETH-8: International Topical Meeting on Nuclear Reactor Thermal-Hydraulics. Atomic Energy Society of Japan, pp. 440–448. [26](#)
- Macian, R., 2010. Uncertainty quantification of best estimate nuclear safety analyses. In: GoNERI Workshop. [24](#)
- Manica, C. C., July 2006. Numerical methods in turbulence. Ph.D. thesis, Univeristy of Pittsburgh. [152](#), [155](#), [159](#)
- Markall, G., Ham, D., Kelly, P., 2010. Towards generating optimised finite element solvers for GPUs from high-level specifications. *Procedia Computer Science* 1 (1), 1815–1823. [39](#)
- Mason, P. J., 1994. Large-eddy simulation: A critical review of the technique. *Quarterly Journal of the Royal Meteorological Society* 120 (515), 1–26. [130](#)
- Mathieu, J., Scott, J., 2000. An introduction to turbulent flow. Cambridge Univ Pr. [48](#), [97](#)
- Meneveau, C., Bou-Zeid, E., Parlange, M. B., 2005. A scale-dependent Lagrangian dynamic model for large eddy simulation of complex turbulent flows. *Physics of Fluids* 17. [134](#), [145](#), [148](#)
- Meneveau, C., Katz, J., 2000. Scale-invariance and turbulence models for large-eddy simulation. *Annual review of fluid mechanics* 32, 1–32. [15](#), [50](#), [134](#)
- Meneveau, C., Lund, T. S., 1996. A Lagrangian dynamic subgrid-scale model of turbulence. *Journal of Fluid Mechanics* 319, 353–385. [135](#), [146](#)
- Menter, F., 1994. Two-equation eddy-viscosity models for engineering applications. *AIAA journal* 32, 1598–1605. [98](#), [99](#)
- Menter, F., 2003. Ten years of industrial experience with the SST model. In: Hanjalić, K., Nagano, Y., Tummers, M. (Eds.), *Turbulence, Heat and Mass Transfer* 4. Begell House Inc. [94](#)
- Menter, F., 2011. Turbulence modeling for engineering flows. Commercial technical paper, ANSYS Inc. [62](#), [98](#)
- Menter, F., Hemstrom, B., 2002. CFD best practice guidelines for CFD code validation for reactor safety applications. Deliverable D01, ECORA. [10](#)

BIBLIOGRAPHY

- Mestayer, P., 1982. Local isotropy and anisotropy in a high Reynolds number turbulent boundary layer. *Journal of Fluid Mechanics* 125, 475–503. 50
- Metais, O., Lesieur, M., 1986. Statistical predictability of decaying turbulence. *Journal of Atmospheric Sciences* 43 (9), 857–870. 181
- Metais, O., Lesieur, M., 1996. New trends in large-eddy simulations of turbulence. *Annual Review of Fluid Mechanics* 28, 45–82. 147
- Meyers, J., 2008. *Quality and Reliability of Large-Eddy Simulations*. Springer. 130
- Meyers, J., Geurts, B. J., Baelmans, M., 2003. Database analysis or errors in large-eddy simulation. *Physics of Fluids* 15 (9), 2740–2755. 10, 30, 131, 180
- Mitran, S., 2001. A comparison of adaptive mesh refinement approaches for large eddy simulation. In: Liu, C., Sakell, L., Beutner, T. (Eds.), *DNS/LES progress and challenges: Third AFOSR international conference*. 184
- Moin, P., Kim, J., 1982. Numerical investigation of turbulent channel flow. *Journal of Fluid Mechanics* 118, 341–377. 129
- Moin, P., Kim, J., 1997. Tackling turbulence with supercomputers. *Scientific American* 276 (1), 46–52. 51
- Moretti, F., Höhne, T., Melideo, D., Del Nevo, A., D’Auria, F., Lisenkov, E., 2009. CFD analysis of a slug mixing experiment conducted on a VVER-1000 model. *Science and Technology of Nuclear Installations*, 1–12. 26, 29
- Moulinec, C., Benhamadouche, S., Laurence, D., Perić, M., 2005. LES in a U-bend pipe meshed by polyhedral cells. In: Rodi, W., Mulas, M. (Eds.), *Engineering Turbulence Modelling and Experiments* 6. pp. 237–246. 68
- Mullen, J., Fischer, P. F., January 1999. Filtering technique for complex geometry flows. *Communications in Numerical Methods in Engineering* 15 (1), 9–18. 155
- Muntean, S., Ruprecht, A., Buntic, I., Susan-Resiga, R., 2005. A numerical investigation of the 3d swirling flow in a pipe with constant diameter. part 2: turbulent computation. In: *Workshop on Vortex Dominated Flows – Achievements and Open Problems*. pp. 1–10. 237

BIBLIOGRAPHY

- Mydlarski, L., Warhaft, Z., 1998. Three-point statistics and the anisotropy of a turbulent passive scalar. *Physics of Fluids* 10, 2885–2894. 50, 188
- Najjar, F., Tafti, D., 1996. Study of discrete test filters and finite difference approximations for the dynamic subgrid-scale stress model. *Physics of Fluids* 8, 1076–1088. 150
- NEA, 2007a. Best practice guidelines for the use of CFD in nuclear reactor safety applications. Tech. Rep. NEA/CSNI/R(2007)5, OECD-Nuclear Energy Agency. 9, 10, 22, 25, 27, 28, 29, 30, 34, 35, 97
- NEA, 2007b. CSNI technical opinion papers no. 9: Level-2 PSA for nuclear power plants. Tech. Rep. NEA No. 5352, OECD-Nuclear Energy Agency. 22
- NEA, 2008. Assessment of computational fluid dynamics (CFD) for nuclear reactor safety problems. Tech. Rep. NEA/CSNI/R(2007)13, OECD-Nuclear Energy Agency. 8, 22, 27, 28, 29, 32
- Nieuwstadt, F., de Valk, J., 1987. A large eddy simulation of buoyant and non-buoyant plume dispersion in the atmospheric boundary layer. *Atmospheric Environment* (1967) 21 (12), 2573–2587. 130
- NREL, 2012. U.S. Department of Energy National Renewable Energy Laboratory High Performance Computing Data Center, information Sheet. 42
- NVIDIA, 2012. NVIDIA Tesla GPUs power world’s fastest supercomputer. URL http://pressroom.nvidia.com/easyir/customrel.do?easyirid=A0D622CE9F579F09&prid=678988&releasejsp=release_157 42
- Oberkampf, W., Roy, C., 2010. A complete framework for verification, validation, and uncertainty quantification in scientific computing. In: 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition. No. AIAA 2010-124. pp. 1–15. 35
- Oberkampf, W., Trucano, T., 2002. Verification and validation in computational fluid dynamics. Tech. Rep. SAND2002-0529, Sandia National Laboratories. 34, 35
- Oberleithner, K., Sieber, M., Wignanski, I., 2011. Three-dimensional coherent structures in a swirling jet undergoing vortex breakdown: stability analysis and empirical mode construction. *Journal of Fluid Mechanics* 679, 383–414. 51

BIBLIOGRAPHY

- Oshima, M., Kobayashi, T., Taniguchi, N., Tsubokura, M., 1997. Development of a filtering operation for dynamic SGS model using the finite element method. In: Chollet, J. P., Voke, P. R., Kleiser, L. (Eds.), *Direct and Large-Eddy Simulation II. ERCOFTAC*, Kluwer Academic Publishers, pp. 57–68. [137](#), [149](#), [267](#), [271](#)
- Oster, D., Wygnanski, I., 1982. The forced mixing layer between parallel streams. *Journal of Fluid Mechanics* 123, 91–130. [51](#)
- Pain, C., Piggott, M., Goddard, A., Fang, F., Gorman, G., Marshall, D., Eaton, M., Power, P., Oliveira, C., 2005. Three-dimensional unstructured mesh ocean modelling. *Ocean Modelling* 10, 5–33. [73](#), [81](#)
- Pain, C., Umpleby, A., De Oliveira, C., Goddard, A., 2001. Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations. *Computer Methods in Applied Mechanics and Engineering* 190 (29-30), 3771–3796. [11](#), [86](#), [88](#), [89](#), [135](#)
- Panjwani, B., Ertesvag, I. S., Gruber, A., Rian, K. E., 2009. Large eddy simulation of backward facing step flow. In: *5th National Conference on Computational Mechanics*. [14](#), [195](#), [214](#), [215](#), [221](#), [226](#), [229](#)
- Papadimitriou, D., Giannakoglou, K., 2008. Aerodynamic shape optimization using first and second order adjoint and direct approaches. *Archive of Computational Methods in Engineering* 15, 447–488. [186](#)
- Park, K., Oh, P.-K., Lim, H.-J., 2006. The application of the CFD and Kriging method to an optimization of heat sink. *International Journal of Heat and Mass Transfer* 49, 3439–3447. [36](#)
- Patankar, S., Baliga, B., 1983. A control-volume finite-element method for two-dimensional fluid flow and heat transfer. *Numerical Heat Transfer, Part A: Applications* 6 (3), 245–262. [104](#)
- Pavlidis, D., Gorman, G., Gomes, J., Pain, C., ApSimon, H., 2009. Synthetic eddy method for urban atmospheric flow modelling. *Boundary-Layer Meteorology*. [139](#), [142](#), [195](#)
- Petry, A., Awruch, A., 2006. Large eddy simulation of three-dimensional turbulent flows by the finite element method. *J. of the Braz. Soc. of Mech. Sci. and Eng.* 28 (2), 224–232. [165](#)
- Pierce, H., Stevens, A., 1975. PWR – flow model scaling. Technical memorandum, Rolls-Royce Submarines. [23](#), [32](#), [33](#)

BIBLIOGRAPHY

- Piggott, M., Farrell, P., Gorman, G., Wilson, C., Pain, C., 2009. Anisotropic mesh adaptivity for multi-scale ocean modelling. *Phil. Trans. R. Soc.* 367, 4591–4611. [86](#), [87](#), [88](#)
- Piomelli, U., 1999. Large-eddy simulation: achievements and challenges. *Progress in Aerospace Sciences* 35, 335–362. [135](#)
- Piomelli, U., Balaras, E., 2002. Wall-layer models for large-eddy simulations. *Annual Review of Fluid Mechanics* 34 (1), 349–374. [141](#)
- Piomelli, U., Balaras, E., Pasinato, H., Squires, K., Spalart, P., 2003. The inner-outer layer interface in large-eddy simulations with wall-layer models. *International Journal of Heat and Fluid Flow* 24 (4), 538–550. [141](#)
- Piomelli, U., Cabot, W., Moin, P., Lee, S., 1991. Subgrid-scale backscatter in turbulent and transitional flows. *Physics of Fluids A: Fluid Dynamics* 3, 1766–1771. [134](#)
- Pope, S. B., 1999. A perspective on turbulence modeling. In: Salas, M. (Ed.), *Modeling Complex Turbulent Flows*. Kluwer Academic Publishers, pp. 53–67. [62](#)
- Pope, S. B., 2000. *Turbulent Flows*. Cambridge Univ Press. [9](#), [10](#), [48](#), [49](#), [50](#), [51](#), [53](#), [55](#), [58](#), [60](#), [64](#), [65](#), [66](#), [95](#), [96](#), [97](#), [98](#), [100](#), [101](#), [130](#), [131](#), [134](#), [137](#), [139](#), [140](#), [144](#), [188](#)
- Pope, S. B., 2001. Large-eddy simulation using projection onto local basis functions. In: Lumley, J. L. (Ed.), *Lecture Notes in Physics*. Vol. 566. Springer, pp. 239–265. [164](#), [182](#)
- Pope, S. B., 2004. Ten questions concerning the large-eddy simulation of turbulent flows. *New Journal of Physics* 6 (35), 1–24. [35](#), [181](#), [183](#), [272](#)
- Porte-Agel, F., 2004. A scale-dependent dynamic model for scalar transport in large-eddy simulations of the atmospheric boundary layer. *Boundary-Layer Meteorology* 112, 81–105. [130](#)
- Porte-Agel, F., Meneveau, C., Parlange, M. B., 2000. A scale-dependent dynamic model for large-eddy simulation: application to a neutral atmospheric boundary layer. *Journal of Fluid Mechanics* 415, 261–284. [148](#)
- Posey, S., Loewe, B., Mathis, H., Calleja, P., 2009. Scalability of transient CFD on large-scale Linux clusters with parallel file systems. Tech. rep., Panasas Inc. and Intel Corporation (UK) Ltd. and University of Cambridge. [40](#)

BIBLIOGRAPHY

- Power, P., Piggott, M., Pain, C., Fang, F., Marshall, C., Goddard, A., Navon, I., 2006. Adjoint goal-based error norms for adaptive mesh ocean modelling. *Ocean Modelling* 15, 3–38. [186](#)
- Prandtl, L., 1925. Bericht über die Entstehung der Turbulenz. *Angew. Math. Mech.* 5, 136–139. [66](#), [95](#)
- Prandtl, L., 1945. Über ein neues Formelsystem für die ausgebildete Turbulenz. *Nachr. Akad. Wiss. Göttingen Math-Phys. Kl.* 6–19. [95](#)
- Queipo, N., Haftka, R., Shyy, W., Goel, T., Vaidyanathan, R., Tucker, K., 2005. Surrogate-based analysis and optimization. *Progress in Aerospace Sciences* 41, 1–28. [37](#)
- Ragab, S., Sreedhar, M., 1995. Numerical simulation of vortices with axial velocity deficits. *Physics of Fluids* 7, 549–558. [236](#)
- Ramakrishnan, S., Collis, S., 2005. The local variational multi-scale method for turbulence simulation. Sandia Report, SAND2005-2733. [84](#)
- Raphson, J., 1690. *Analysis Aequationum Universalis*. London. [186](#)
- Reynolds, W., 1990. The potential and limitations of direct and large eddy simulations. *Lecture Notes in Physics* 357, 313–343. [12](#)
- Richardson, L., 1922. *Weather prediction by numerical process*. Cambridge Univ Press. [48](#)
- Roache, P., 2002. Code verification by the method of manufactured solutions. *Transactions of the ASME* 124, 4–10. [33](#), [110](#), [165](#)
- Rohde, U., Höhne, T., Kliem, S., Scheuerer, M., 2007. Fluid mixing and flow distribution in a primary circuit of a nuclear pressurized water reactor – validation of CFD codes. *Nuclear Engineering and Design* 237, 1639–1655. [26](#), [29](#)
- Rolfo, S., Uribe, J. C., Laurence, D., 2010. LES and hybrid RANS/LES of turbulent flow in fuel rod bundle arranged with a triangular array. In: Armenio, V., Geurts, B., Fröhlich, J. (Eds.), *Direct and Large-Eddy Simulation VII*. Vol. 13 of ERCOFTAC Series. Springer Netherlands, pp. 409–414. [26](#), [141](#)
- Rollet-Miet, P., Laurence, D., Ferziger, J., 1999. LES and RANS of turbulent flow in tube bundles. *International Journal of Heat and Fluid Flow* 20, 241–254. [77](#), [81](#)

BIBLIOGRAPHY

- Rolls-Royce, 2010. CFD best practice guidelines. Tech. rep., Rolls-Royce Submarines. 10, 23, 29, 31, 34
- Rolls-Royce, 2012. Rolls-Royce Submarines homepage.
URL http://www.rolls-royce.com/marine/about/market_sectors/submarines/ 7, 18
- Roy, C., Tendean, E., Veluri, S., Rifki, R., Luke, E., Hebert, S., 2007. Verification of RANS turbulence models in Loci-CHEM using the method of manufactured solutions. In: 18th AIAA Computational Fluid Dynamics Conference. AIAA. 111
- Sagaut, P., 1999. Discrete filters for large eddy simulation. *International Journal for Numerical Methods in Fluids* 31, 1195–1220. 154
- Sagaut, P., 2006. Multiscale and multiresolution approaches in turbulence. Imperial College Press. 10, 85, 137, 147, 181, 184, 185
- Sagaut, P., Ménéveau, C., 2006. Large eddy simulation for incompressible flows: an introduction. Springer Verlag. 50, 61, 62, 134, 138, 141, 193, 194
- Sage, 2011. Sage manual.
URL <http://www.sagemath.org/doc/reference/index.html> 111
- Sahni, O., Carothers, C., Shephard, M., Jansen, K., 2009. Strong scaling analysis of a parallel, unstructured, implicit solver and the influence of the operating system. *Scientific Programming* 17 (3), 261–274. 17, 40
- Sampaio, P., Hallak, P., Coutinho, A., Pfeil, M., 2004. A stabilized finite element procedure for turbulent fluid–structure interaction using adaptive time–space refinement. *International Journal for Numerical Methods in Fluids* 44, 673–693. 28, 129
- Scheuerer, M., Andreani, M., Bestion, D., Egorov, Y., Heitsch, M., Menter, F., Muhlbauer, P., Pigny, S., Schwäger, C., Willemsen, S., 2005a. Condensed final summary report. Technical memorandum EVOL– ECORA – D 17, ECORA. 11
- Scheuerer, M., Menter, F., Bestion, D., Smith, B., 2005b. Evaluation of computational fluid dynamic methods for reactor safety analysis (ECORA). *Nuclear Engineering and Design* 235 (2-4), 359–368. 8, 26
- Schmidt, R., Schumann, U., 1989. Coherent structure of the convective boundary layer derived from large-eddy simulations. *Journal of Fluid Mechanics* 200, 511–562. 189

BIBLIOGRAPHY

- Schneider, G., Raw, M., 1987. Control volume finite-element method for heat transfer and fluid flow using colocated variables - 1. computational procedure. *Numerical Heat Transfer: An International Journal of Computation and Methodology* 11 (4), 363–390. [104](#), [105](#)
- Schumann, U., 1975. Subgrid scale model for finite difference simulations of turbulent flows in plane channels and annuli. *Journal of Computational Physics* 18, 376–404. [101](#), [102](#), [147](#)
- Scotti, A., Meneveau, C., Lilly, D. K., 1993. Generalized Smagorinsky model for anisotropic grids. *Physics of Fluids* 5 (9), 2306–2308. [137](#)
- Sehgal, B., 2006. Light water reactor (LWR) safety. *Nuclear Engineering and Technology* 38 (8), 697–732. [8](#), [32](#)
- SGI, 2009. Industrial CFD application performance on SGI® Altix® ICE 8200EX, product brochure. [40](#)
- Shih, T.-H., Liou, W. W., Shabbir, A., Yang, Z., Zhu, J., 1995. A new k-epsilon eddy viscosity model for high Reynolds number turbulent flows. *Computers and Fluids* 24 (3), 227–238. [104](#)
- Simpson, R., 1996. Aspects of turbulent boundary-layer separation. *Progress in Aerospace Sciences* 32, 457–521. [193](#)
- Smagorinsky, J., 1963. General circulation experiments with the primitive equations. *Monthly Weather Review* 91 (3), 99–164. [95](#), [133](#)
- Smith, B., 2010. Assessment of CFD codes used in nuclear reactor safety simulations. *Nuclear Engineering and Technology* 42 (4), 339–364. [8](#), [9](#), [25](#), [27](#), [28](#)
- Sohankar, A., Davidson, L., Norberg, C., 2000. Large eddy simulation of flow past a square cylinder: comparison of different subgrid scale models. *Journal of Fluids Engineering* 122, 39–47. [129](#)
- Spalart, P., 2009. Detached eddy simulation. *Annual Review of Fluid Mechanics* 41, 181–202. [30](#), [63](#), [110](#)
- Spalding, D., 1961. A single formula for the law of the wall. *Journal of Applied Mechanics* 28 (3), 444–458. [194](#)
- Speziale, C., 1997. Lecture notes in physics: Computing non-equilibrium turbulent flows with time-dependent RANS and VLES. In: *Fifteenth International Conference on Numerical Methods in Fluid Dynamics*. Vol. 490. Springer, pp. 123–129. [63](#), [110](#)

BIBLIOGRAPHY

- Speziale, C., Thangam, S., 1992. Analysis of an RNG based turbulence model for separated flows. *International Journal of Engineering Science* 30 (10), 1379–1388. [114](#), [194](#)
- Sreenivasan, K., 1991. On the local isotropy of passive scalars in turbulent shear flows. *Proceedings of the Royal Society of London* 434, 165–182. [50](#)
- Susan-Resiga, R., Avellan, F., Muntean, S., Ciocan, D., 2005. Mathematical and numerical modeling of swirling flow in Francis turbine draft tube cone. In: *Workshop on Vortex Dominated Flows - Achievements and Open Problems*. pp. 1–16. [237](#)
- Sweby, P., 1984. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM Journal on Numerical Analysis* 21 (5), 995–1011. [105](#)
- Takata, T., Yamaguchi, A., Tanaka, M., Ohshima, H., October 2007. Application of LES with a near-wall model in nuclear plant thermal hydraulics. In: *The 12th International Topical Meeting on Nuclear Reactor Thermal Hydraulics (NURETH-12)*. [129](#)
- Takenaka, N., Fujii, T., Ono, A., Sonoda, K., Nishizaki, K., Asano, H., 1990. Application of neutron radiography to visualization of multiphase flows. *Flow Measurement and Instrumentation* 1 (3), 149–156. [32](#)
- Tao, B., Katz, J., Meneveau, C., 2000. Geometry and scale relationships in high Reynolds number turbulence determined from three-dimensional holographic velocimetry. *Physics of Fluids* 12 (941), 1–6. [134](#), [139](#)
- Tejada-Martinez, A. E., Jansen, K., 2006. A parameter-free dynamic subgrid-scale model for large-eddy simulation. *Computer Methods in Applied Mechanics and Engineering* 195 (23-24), 2919–2938. [149](#)
- Tejada-Martinez, A. E., Jansen, K. E., 2003. Spatial test filters for dynamic model large-eddy simulation with finite elements. *Communications in Numerical Methods in Engineering* 19, 205–213. [149](#), [156](#), [158](#), [164](#), [230](#)
- Tejada-Martinez, A. E., Jansen, K. E., 2004. A dynamic Smagorinsky model with dynamic determination of the filter width ratio. *Physics of Fluids* 16 (7). [148](#), [149](#), [158](#)
- Tejada-Martinez, A. E., Jansen, K. E., 2005. On the interaction between dynamic model dissipation and numerical dissipation due to streamline upwind/Petrov-Galerkin stabilization. *Computer Methods in Applied Mechanics and Engineering* 194, 1225–1248. [150](#)

BIBLIOGRAPHY

- Todreas, N., Kazimi, M., 1990. Nuclear systems 1: thermal hydraulic fundamentals. Taylor & Francis. 20, 21, 23
- Top500, September 2012. Performance development of top 500 hpc facilities. URL http://www.top500.org/lists/2012/06/performance_development 39
- Toschi, F., Kobayashi, T., Piomelli, U., Iaccarino, H., 2006. Backward-facing step calculations using the shear improved Smagorinsky model. In: Proceedings of the summer program 2006. Center for Turbulence Research, pp. 87–97. 129
- Travin, A., Shur, M., Spalart, P., Strelets, M., 2004. On URANS solutions with LES-like behaviour. In: Neittaanmäki, P., Rossi, T., Majava, K., Pironneau, O. (Eds.), Congress on Computational Methods in Applied Sciences and Engineering ECCOMAS 2004. 9
- Turgeon, E., Pelletier, D., Ignat, L., 2000. Effects of adaptivity on finite element schemes for turbulent heat transfer and flow predictions. Numerical Heat Transfer, Part A: Applications 38 (8), 847–868. 97
- Umlauf, L., Burchard, H., 2003. A generic length-scale equation for geophysical turbulence models. Journal of Marine Research 61 (31), 235–265. 109
- Vallet, M.-G., Manole, C.-M., Dompierre, J., Dufour, S., Guibault, F., 2007. Numerical comparison of some Hessian recovery techniques. International Journal for Numerical Methods in Fluids 72, 987–1007. 89
- van der Bos, F., Geurts, B. J., 2005. Commutator errors in the filtering approach to large-eddy simulation. Physics of Fluids 17. 158, 159, 160
- van Driest, E., 1956. On turbulent flow near a wall. Journal of the Aeronautical Sciences 23, 1007–1011. 66
- Vasilevskii, Y., Lipnikov, K., 1999. An adaptive algorithm for quasioptimal mesh generation. Computational Mathematics and Mathematical Physics 39 (9), 1468–1486. 86
- Vasilyev, O. V., Lund, T. S., Moin, P., 1998. A general class of commutative filters for les in complex geometries. Journal of Computational Physics 146, 82–104. 160
- Vasilyev, O. V., Marsden, A. L., Moin, P., 2002. Construction of commutative filters for LES on unstructured meshes. Journal of Computational Physics 175, 584–603. 160, 161

BIBLIOGRAPHY

- Veber, P., Andersson, L., 2004. CFD calculation of flow and thermal mixing in a T-junction - time-dependent calculation part 2 (BG-project). Tech. Rep. Teknisk not 2004/21, Onsala Ingenjörbyrå. 26
- Venditti, D., Darmofal, D., 2003. Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows. *Journal of Computational Physics* 187 (1), 22–46. 186
- Vreman, B., Geurts, B., Kuerten, H., 1994. Realizability conditions for the turbulent stress tensor in large eddy simulation. *Journal of Fluid Mechanics* (339), 357. 140, 189
- Vreman, B., Geurts, B., Kuerten, H., 1996. Large-eddy simulation of the temporal mixing layer using the Clark model. *Theoretical and Computational Fluid Dynamics* 8, 309–324, 10.1007/BF00639698. 150, 155
- VTK, 2012. Vtk visualization toolkit.
URL <http://www.vtk.org/> 202
- Wang, B., Bergstrom, D. J., 2005. A dynamic nonlinear subgrid-scale stress model. *Physics of Fluids* 17. 139
- Wang, M., Moin, P., 2002. Dynamic wall modeling for large-eddy simulation of complex turbulent flows. *Physics of Fluids* 14 (7), 2043–2051. 194
- Weber, R., Visser, B., Boysan, F., 1990. Assessment of turbulence modeling for engineering prediction of swirling vortices in the near burner zone. *International Journal of Heat and Fluid Flow* 11, 225–235. 58
- Werner, H., Wengle, H., 1989. Large-eddy simulation of the flow over a square rib in a channel. In: proceedings of the seventh symposium on turbulent shear flows. Stanford University, pp. 10.2.1–10.2.6. 66
- Westin, J., Alavyoon, F., Andersson, L., Veber, P., Henriksson, M., Andersson, C., 2006. Experiments and unsteady CFD-calculations of thermal mixing in a T-junction. In: Proc. Int. Workshop on Benchmarking of CFD Codes for Application to Nuclear Reactor Safety (CFD4NRS). NEA, pp. 494–508. 30
- Wilcox, D., 1989. Wall matching, a rational alternative to wall functions. In: 27th AIAA Aerospace Sciences Meeting. No. paper 89-611. AIAA. 101
- Wilcox, D., 1998. Turbulence modeling for CFD. DCW Industries Inc., La Cañada, CA. 52, 58, 96, 97, 98, 99, 100, 117, 264

BIBLIOGRAPHY

- Wilson, C., 2009. Modelling multiple-material flows on adaptive unstructured meshes. Ph.D. thesis, Imperial College London Dept. of Earth Science and Engineering. 78, 85, 105, 106
- Winckelmans, G. S., Wray, A. A., Vasilyev, O. V., 1998. Testing of a new mixed model for LES: the Leonard model supplemented by a dynamic Smagorinsky term. In: Proceedings of the summer program 1998. Center for turbulence research, pp. 367–388. 150, 158
- WNA, 2012a. Safety of nuclear reactors. Public information service, World Nuclear Association. 7
- WNA, 2012b. WNA nuclear reactors database. Public information service, World Nuclear Association.
URL www.world-nuclear.org/NuclearDatabase 18
- WTEC, 2009. WTEC panel report on international assessment of research and development in simulation-based engineering and science. Tech. rep., World Technology Evaluation Center. 39
- Yadigaroglu, G., 2005. Computational fluid dynamics for nuclear applications: from CFD to multi-scale CMFD. Nuclear Engineering and Design 235, 153–164. 8
- Yang, Z., Priestman, G., 1991. Internal flow modelling of vortex throttles. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering 205, 405–413. 237, 264
- Yin, J., Liao, L., Wang, L., 2010. Large eddy simulation of unsteady flow in vortex diode. Nuclear Engineering and Design 240, 970–974. 237
- Yoder Jnr., G., Elkassabgi, Y., De Leon, G., Fetterly, C., Ramos, J., Robbins, J., 2011. Vortex diode analysis and testing for fluoride salt-cooled high-temperature reactors. Technical memorandum ORNL/TM-2011/425, Oak Ridge National Laboratory and Texas A&M University. 238
- Yoshizawa, A., 1986. Statistical theory for compressible turbulent shear flows, with the application to subgrid modeling. Physics of Fluids 29, 2152–2164. 141
- Zahrai, S., Bark, F., Karlsson, R., 1995. On anisotropic subgrid modelling. European Journal of Mechanics, B. fluids 14 (4), 459–486. 138

BIBLIOGRAPHY

- Zang, Y., Street, R., Koseff, J., 1993. A dynamic mixed subgrid-scale model and its application to turbulent recirculating flows. *Physics of Fluids A: Fluid Dynamics* 5 (12), 3186–3196. [138](#), [150](#)
- Zienkiewicz, O., Taylor, R., 1997. *The finite element method*, Vol. 1, 4th Edition. Elsevier. [36](#), [76](#), [256](#)