TECHNISCHE UNIVERSITÄT MÜNCHEN

Lehrstuhl für Numerische Mechanik

Wall Modeling via Function Enrichment for Computational Fluid Dynamics

Benjamin Krank

Vollständiger Abdruck der von der Fakultät für Maschinenwesen der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender: Prof. Dr.-Ing. Hans-Jakob Kaltenbach

Prüfer der Dissertation:

1. Prof. Dr.-Ing. Wolfgang A. Wall
2. Prof. Dr. rer. nat. habil. Claus-Dieter Munz
   Universität Stuttgart

Die Dissertation wurde am 28.02.2018 bei der Technischen Universität München eingereicht und durch die Fakultät für Maschinenwesen am 30.01.2019 angenommen.

Abstract

Computational fluid dynamics is a key design tool in the transportation and energy sectors. Today, many engineering flow problems cannot be computed with satisfactory accuracy and/or within an acceptable time scale. The bottleneck in the simulations is the near-wall region, where the grid resolution requirements can be very high, especially with regard to high Reynolds numbers. One reason for these resolution requirements is the high gradient occurring in the velocity variable just at the wall in turbulent boundary layers.

The primary contribution of this work is a novel approach to computing the flow in the near-wall region in a very cost-effective and yet accurate way. The idea is that the velocity profile is modeled inside the cells of the numerical method at the same time as the Navier–Stokes equations are fulfilled discretely in the whole boundary layer, including the no-slip boundary condition. This approach enables the use of coarse meshes in the vicinity of the wall while the method still accurately accounts for high longitudinal pressure gradients and nonequilibrium boundary layer conditions. Therein, the method is based on the concept of function enrichment. The standard polynomial function space of the Galerkin method is enriched by a few additional shape functions, which are constructed using a wall function. With such a function space available, the numerical method automatically finds the optimal solution as a linear combination of the wall function and the polynomial shape functions. It is the wall-function component of the solution that enables the accurate representation of the high gradient at the wall, whereas the polynomial component allows for more general velocity profiles.

The large discretization cells used with this approach imply that the turbulent motions in the near-wall region are not resolved. These turbulent scales are modeled by the classical Reynolds-averaged Navier–Stokes (RANS) or detached-eddy simulation (DES) methodologies in this work. In addition, the composition of the solution of a wall-function component and a polynomial component enables a much more general turbulence model for eddy-resolving simulations. The polynomial component resolves eddies where the mesh is sufficiently fine; then the wall-function component computes the remaining flow in an averaged sense. This idea solves the typical problems associated with the outer edge of the RANS region in DES. All turbulence modeling approaches are assessed by use of the benchmark examples of turbulent channel flow and flow over periodic hills, which provide insight into the performance of the new models in attached and separated boundary layers. The wall model shows excellent results in both of these flow conditions and is capable of reproducing the results of wall-resolved simulations at a greatly lower computational cost – up to two orders of magnitude.

The wall model may be included in any Galerkin method; the model is, in this work, implemented in a continuous finite element method (FEM) and a high-order discontinuous Galerkin (DG) solver. The latter solver targets turbulent incompressible flow and is developed further in this thesis in order to address recently reported instabilities in spatially underresolved simulations as well as small time steps. The key feature of the solver is a consistent penalty term that reduces the local divergence error in order to overcome the instabilities. This penalty method is similar to the grad-div stabilization widely used in the FEM. The numerical method is applied to fully resolved turbulent flow (direct numerical simulation) and underresolved turbulent flow (implicit large-eddy simulation). The high-order DG solver is also employed to compute a new set of reference data of the flow over periodic hills with the highest fidelity to date.

Zusammenfassung

Die numerische Strömungssimulation ist eines der wichtigsten Entwicklungswerkzeuge im Verkehrs- und Energiesektor. Viele strömungsmechanische Problemstellungen können heute nicht mit zufriedenstellender Genauigkeit und/oder innerhalb einer akzeptablen Simulationszeit berechnet werden. Eine der größten Schwierigkeiten in der numerischen Strömungsmechanik ist der wandnahe Bereich, da dort die Anforderungen an die Netzfeinheit insbesondere im Hinblick auf hohe Reynoldszahlen sehr hoch sind. Einer der Gründe für diese Anforderung ist der hohe Gradient in der Geschwindigkeitsvariable, der in turbulenten Grenzschichten in der wandnächsten Schicht auftritt.

Der Hauptbeitrag dieser Arbeit ist ein neuartiger Ansatz zur kostengünstigen und dennoch genauen Berechnung der Strömung im wandnahen Bereich. Die grundlegende Idee besteht darin, das Geschwindigkeitsprofil innerhalb der Zellen der numerischen Methode zu modellieren, gleichzeitig aber die Navier–Stokes Gleichungen in der gesamten Grenzschicht diskret zu erfüllen, inklusive der Haftbedingnung an der Wand. Der Ansatz ermöglicht grobe Netze in Wandnähe. Die Methode berücksichtigt trotzdem hohe Druckgradienten und kann Grenzschichten darstellen, die nicht im Gleichgewichtszustand sind. Das Modell basiert auf dem Konzept der Funktionsanreicherung. Darin wird der gewöhnliche polynomiale Funktionenraum der Galerkinmethode mit zusätzlichen Formfunktionen angereichtert, die mit Hilfe einer Wandfunktion konstruiert werden. Mit dem dadurch zur Verfügung stehenden Funktionenraum findet die numerische Methode automatisch die optimale Lösung als eine Linearkombination aus der Wandfunktion und den polynomialen Formfunktionen. Die Komponente, die auf der Wandfunktion basiert, ermöglicht dabei die genaue Darstellung des hohen Gradienten an der Wand, wohingegen die polynomiale Komponente davon abweichende Geschwindigkeitsprofile berücksichtigen kann.

Da die finiten Elemente der Diskretisierung mit diesem Ansatz sehr grob sind, werden die turbulenten Skalen in Wandnähe nicht aufgelöst. Diese Skalen werden im Rahmen der Arbeit durch klassische Reynolds-Averaged Navier–Stokes (RANS) und Detached-Eddy Simulationsmodelle (DES) berücksichtigt. Zusätzlich ermöglicht die Zusammensetzung der Lösung aus einer Wandfunktions- und einer polynomialen Komponente ein neuartiges und wesentlich genaueres Turbulenzmodell für wirbelauflösende wandmodellierte Strömungssimulationen. Darin repräsentiert die polynomiale Komponente Wirbel dort, wo das Rechengitter ausreichend fein ist. Die restliche Strömung wird gemittelt und durch die Wandfunktionskomponente dargestellt. Das resultierende Turbulenzmodell zeigt im Gegensatz zu DES keine Probleme im Übergangsbereich am Rand des RANS Gebiets. Alle Turbulenzmodellierungsansätze werden mit Hilfe von turbulenten Benchmarkströmungen, bestehend aus einer Strömung in einem Kanal und über periodische Hügel, bewertet. Diese Beispiele ermöglichen einen Einblick in das Verhalten des Wandmodells sowohl in anliegenden als auch abgelösten Grenzschichten. Das Modell zeigt ausgezeichnete Eigenschaften in beiden Strömungszuständen und resultiert in deutlich reduzierten Rechenkosten – um bis zu zwei Größenordnungen im Vergleich zu wandaufgelösten Simulationen.

Das Wandmodell kann in jeden beliebigen Strömungslöser basierend auf der Galerkinmethode implementiert werden. Das Modell wird in dieser Arbeit in einen kontinuierlichen Finite Elemente Methode (FEM) und einen diskontinuierlichen Galerkin (DG) Löser mit hohen Ordnungen integriert. Letzterer wird im Hinblick auf die Berechnung von turbulenten inkompressiblen Strömungen weiterentwickelt, indem Problemlösungen zu kürzlich berichteten Instabilitäten in räumlich unteraufgelösten Simulationen und mit kleinen Zeitschritten vorgestellt werden. Die Schlüsselkomponente des Lösers ist ein konsistenter Bestrafungsterm, der den lokalen Divergenzfehler in effektiver Weise reduziert. Die Funktionsweise dieses Bestrafungsterms ist ähnlich der grad-div Stabilisierung, die im Rahmen der kontinuierlichen FEM vielfach angewendet wird. Die numerische Methode wird zur Berechnung von vollaufgelösten (direkte numerische Simulation) und unteraufgelösten (implizite Grobstruktursimulation) turbulenten Strömungen verwendet. Im Rahmen einer DNS wird dieser DG Löser auch dazu verwendet, einen neuen Referenzdatensatz für die Strömung über periodische Hügel zu berechnen, der alle bisherigen Referenzdaten an Genauigkeit weit übertrifft.

Acknowledgements

First of all, I would like to express my sincere gratitude to Prof. Dr.-Ing. Wolfgang A. Wall for granting me the opportunity to join the Institute for Computational Mechanics at the Technical University of Munich, where I was able to work on the exciting topics of wall modeling and computational fluid dynamics during my Ph.D. studies. I would also like to thank Dr. Martin Kronbichler for the innumerable scientific discussions and his continuous support. Further, Dr.-Ing. Ursula Rasthofer was of exceptional help during my first year at the institute as I became familiar with the preexisting work in turbulence modeling and the simulation code BACI. The flawless operation of the institute's HPC clusters contributed to the present research, therefore, I would like to thank the cluster administrators, Dr.-Ing. Tobias Wiesner and Andreas Rauch, for their tireless effort of keeping the compute systems running. Likewise, computational resources on SuperMUC in Garching, Germany, provided by the Leibniz Supercomputing Centre under the project pr83te, are gratefully acknowledged. Finally, I would like to thank my wonderful colleagues at the Institute for Computational Mechanics and anyone who has not been named in this listing by quoting Antoine de Saint-Exupéry "L'essentiel est invisible pour les yeux".

Munich, February 2019 Benjamin Krank

Contents

1. Introduction 1
- 1.1. Motivation . . . 1
- 1.2. Contribution of this Work . . . 2
- 1.3. Thesis Structure . . . 4

2. A Brief Introduction to Turbulence and its Numerical Simulation 7
- 2.1. Turbulent Flow: A Multiscale Phenomenon . . . 7
  - 2.1.1. Incompressible Navier–Stokes Equations . . . 7
  - 2.1.2. Bulk Turbulence . . . 8
  - 2.1.3. Boundary Layer Turbulence . . . 9
  - 2.1.4. The Multiscale Nature of Turbulence: A Challenge to Simulation and Modeling . . . 12
- 2.2. Overview of Numerical Discretization Schemes . . . 13
  - 2.2.1. Requirements on the Numerical Baseline Method . . . 13
    - 2.2.1.1. Efficient Time Stepping . . . 14
    - 2.2.1.2. High-Order Accuracy . . . 14
    - 2.2.1.3. Complex Geometries . . . 14
    - 2.2.1.4. Convective Stability . . . 15
  - 2.2.2. Numerical Methods . . . 15
    - 2.2.2.1. The Finite Difference Method . . . 15
    - 2.2.2.2. The Finite Volume Method . . . 16
    - 2.2.2.3. The (Spectral) Finite Element Method . . . 16
    - 2.2.2.4. The Discontinuous Galerkin Method . . . 18
  - 2.2.3. Suitability of Standard Schemes for Simulating Multiscale Turbulence . 18
- 2.3. Fundamental Approaches to the Numerical Simulation of Wall-Bounded Turbulence . . . 18
  - 2.3.1. Direct Numerical Simulation . . . 19
  - 2.3.2. Large-Eddy Simulation . . . 19
  - 2.3.3. Reynolds-Averaged Navier–Stokes . . . 20
  - 2.3.4. DES, Wall Modeling, and Hybrid RANS/LES . . . 21
- 2.4. Summary . . . 22

I Wall-Resolved Computational Fluid Dynamics: High-Order DG Methods and Applications 25

3. A High-Order Incompressible Semi-Explicit Discontinuous Galerkin Solver 27
- 3.1. A Review of High-Order DG Methods for Incompressible Flow and Motivation 27
  - 3.1.1. Use Compressible or Incompressible Schemes? . . . 27
  - 3.1.2. DG Schemes for Incompressible Flow . . . 28

3.1.3. Instabilities Reported in the Literature 29
3.2. Temporal Velocity-Correction Scheme 30
3.3. Spatial Discretization 32
3.3.1. Preliminaries 32
3.3.2. Variational Formulation 33
3.3.3. The CFL Condition and Adaptive Time Stepping 36
3.4. Four Variants of the Projection Step and Impact on Stability 37
3.4.1. Sources of Instabilities 38
3.4.1.1. The Small-Time-Step Limit 38
3.4.1.2. Conservation of Mass in the Underresolved Limit 39
3.4.2. Four Variants 41
3.4.2.1. Variant 1 (V1) 41
3.4.2.2. Variant 2 (V2) 42
3.4.2.3. Variant 3 (V3) 42
3.4.2.4. Variant 4 (V4) 45
3.4.3. Conclusion on Variants 1 to 4 47
3.4.4. Matrix Formulation 47
3.5. Implementation 48
3.5.1. Evaluation and Integration of Cells and Faces via Sum Factorization . . 48
3.5.2. Matrix-Free Solution of Linear Systems 49
3.5.3. Performance Evaluation 50
3.6. Verification 52
3.6.1. Vortex Problem 53
3.6.1.1. Temporal Convergence 53
3.6.1.2. Spatial Convergence 54
3.6.1.3. Performance Evaluation 54
3.6.2. Laminar Flow past Cylinder 56
3.7. Summary 58

4. Application to DNS and LES of Turbulent Flow **59**
4.1. DNS and Implicit LES using High-Order DG Methods 59
4.1.1. Resolution Power for DNS 59
4.1.2. Application to LES 60
4.1.3. Criteria for Assessing the Near-Wall Resolution 61
4.2. Turbulent Channel Flow 61
4.2.1. Direct Numerical Simulation 62
4.2.2. Implicit Large-Eddy Simulation 63
4.3. Flow over Periodic Hills 66
4.3.1. Previous Reference Data 67
4.3.2. Extension of the Reference Database 68
4.3.3. Simulation Setup, Resolution Requirements, and Mesh 69
4.3.3.1. Simulation Setup 69
4.3.3.2. Meshes 70
4.3.3.3. Assessment of Resolution 71
4.3.3.4. Computational Cost 73

4.3.4. Direct Numerical Simulation 73
    4.3.4.1. $\mathrm{Re}_H = 5{,}600$: Skin Friction and Pressure Coefficients 74
    4.3.4.2. $\mathrm{Re}_H = 5{,}600$: Velocity Statistics 77
    4.3.4.3. $\mathrm{Re}_H = 10{,}595$: Skin Friction and Pressure Coefficients . . . 78
    4.3.4.4. $\mathrm{Re}_H = 10{,}595$: Velocity Statistics 80
    4.3.4.5. Analysis of the Sampling Error 82
4.3.5. Implicit Large-Eddy Simulation 84
    4.3.5.1. ILES h/p-Refinement Study at $\mathrm{Re}_H = 10{,}595$: Skin Friction and Pressure Coefficients 86
    4.3.5.2. ILES h/p-Refinement Study at $\mathrm{Re}_H = 10{,}595$: Velocity Statistics 88
4.4. Summary 91

5. Extension to RANS and DES 93
5.1. Literature Review and Motivation 93
5.2. Incompressible Navier–Stokes Equations and the Spalart–Allmaras Model . . . 94
    5.2.1. The Spalart–Allmaras Model 94
    5.2.2. Extension of the Velocity-Correction Scheme 95
5.3. Spatial Discretization 96
    5.3.1. Galerkin Formulation 97
    5.3.2. Time Stepping 99
    5.3.3. Matrix Formulation 100
5.4. Numerical Examples 101
    5.4.1. Turbulent Channel Flow 102
    5.4.2. Flow over Periodic Hills 103
5.5. Application to Detached-Eddy Simulation 107
    5.5.1. Turbulent Channel Flow 108
    5.5.2. Flow over Periodic Hills 110
5.6. Summary 111

II Wall Modeling via Function Enrichment 115

6. Wall Modeling via Function Enrichment: Motivation, Formulation, Implementation 117
6.1. The two Challenges in Computing Turbulent Boundary Layers 117
6.2. A Review of Wall Modeling Approaches for CFD 119
    6.2.1. Detached-Eddy Simulation 119
    6.2.2. WMLES Using Wall-Stress Models 121
    6.2.3. WMLES Using the Two-Layer Model 121
    6.2.4. Preview of Wall Modeling via Function Enrichment 122
    6.2.5. Wall Modeling in RANS 122
6.3. Capturing the Gradient via Function Enrichment 122
    6.3.1. Function Enrichment: Applications with High Gradients 123

6.3.2. Wall Modeling via Function Enrichment 124
    6.3.2.1. Enriching the DG Solution Space 125
    6.3.2.2. Enriching the Solution Space of the Continuous FEM 127
    6.3.2.3. Wall Functions as Enrichment Function 128
    6.3.2.4. Spatial and Temporal Adaptation 131
    6.3.2.5. Turbulence Modeling 132
6.3.3. Implementation of High-Gradient Enrichments 133
    6.3.3.1. Evaluation of Weak Forms 134
    6.3.3.2. Matrix-Free Application of the Inverse Mass Matrix 135
    6.3.3.3. Mixed Matrix-Free Projection Step 139
    6.3.3.4. Implementation of Enriched Integrals in C++ 142
6.4. Summary 144

7. Application I: Wall Modeling for LES in the Continuous FEM 145
7.1. Enrichment Space 146
7.2. Subgrid-Scale Modeling 148
    7.2.1. Weighted Residual Formulation 148
    7.2.2. Scale Separation for Large-Eddy Simulation 149
    7.2.3. Subgrid Modeling with Multifractal Subgrid Scales 150
    7.2.4. Residual-Based Modeling 152
    7.2.5. Final Discrete Problem 154
7.3. Numerical Examples 155
    7.3.1. Turbulent Channel Flow 155
    7.3.2. Flow over Periodic Hills 157
    7.3.3. Backward Facing Step Flow 163
7.4. Summary 167

8. Application II: Wall-Modeling for RANS in High-Order Discontinuous Galerkin 169
8.1. Enrichment Space: Enrich Only When Needed 169
    8.1.1. Enrichment Space 170
    8.1.2. Enhancement of the Adaptation Scheme 170
    8.1.3. Implementation 172
8.2. Numerical Examples 172
    8.2.1. Turbulent Channel Flow 172
    8.2.2. Flow over Periodic Hills 177
8.3. Summary 181

9. Application III: Extension to DES 183
9.1. Enrichment Space 184
9.2. Numerical Examples 185
    9.2.1. Turbulent Channel Flow 185
    9.2.2. Flow over Periodic Hills 188
9.3. Summary 191

10. Application IV: A Multiscale Approach to Hybrid RANS/LES Wall Modeling 195
- 10.1. Basic Idea and Comparison to Existing Hybrid RANS/LES Methods 195
  - 10.1.1. Basic Idea 195
  - 10.1.2. Comparison to Other Hybrid RANS/LES Methodologies 196
- 10.2. A Multiscale Approach to Wall Modeling 197
  - 10.2.1. Three-Level Hybrid RANS/LES Filter 198
    - 10.2.1.1. The Classical Two-Level Additive Hybrid RANS/LES Filter 198
    - 10.2.1.2. A New Three-Level Additive Hybrid RANS/LES Filter . . . 199
  - 10.2.2. Variational Multiscale Formulation 202
- 10.3. RANS and LES Velocity Components Using Wall Modeling via Function Enrichment 204
- 10.4. Galerkin Formulation and Implementation 206
  - 10.4.1. Galerkin Formulation 206
  - 10.4.2. Coercivity Analysis 209
  - 10.4.3. Implementation 211
- 10.5. Numerical Examples 211
  - 10.5.1. Turbulent Channel Flow 211
  - 10.5.2. Flow over Periodic Hills 222
- 10.6. Summary 228

11. Summary and Outlook 229

A. Evaluation of Enrichment Functions and Spatial Derivatives 233
- A.1. Numerical Evaluation of Spalding's Law 233
- A.2. Numerical Evaluation of van Driest's Law 233
- A.3. Derivatives of the Enrichment in Cartesian Coordinates 234

Bibliography 237

Nomenclature

Abbreviations

ATAAC	Advanced Turbulence Simulation for Aerodynamic Application Challenges
BDF	Backward Differentiation Formula
CFD	Computational Fluid Dynamics
CFL	Courant–Friedrichs–Lewy
CG	Conjugate Gradient method
DDES	Delayed Detached-Eddy Simulation
DEM	Discontinuous Enrichment Method
DES	Detached-Eddy Simulation
DG	Discontinuous Galerkin
DNS	Direct Numerical Simulation
DOF	Degree Of Freedom
ER	Expansion Ratio
FD	Finite Difference
FEM	Finite Element Method
FV	Finite Volume
GMRES	Generalized Minimum-RESidual method
HDG	Hybridizable Discontinuous Galerkin
HRLES	Highly Resolved Large-Eddy Simulation
ILES	Implicit Large-Eddy Simulation
INDEXA	INcompressible Discontinuous Galerkin towards the EXA scale
LES	Large-Eddy Simulation
LU	Lower Upper
NIPG	Non-Symmetric Interior Penalty Galerkin
NLDE	NonLinear Disturbance Equations
PSPG	Pressure Stabilizing Petrov–Galerkin
PUM	Partition-of-Unity Method
RANS	Reynolds-Averaged Navier–Stokes
RMS	Root-Mean Square $\mathrm{RMS}(\phi) = \sqrt{\langle \phi^2 \rangle - (\langle \phi \rangle)^2}$
RSS	Reynolds Shear Stress
SA	Spalart–Allmaras
SIPG	Symmetric Interior Penalty Galerkin
SUPG	Streamline Upwind Petrov–Galerkin
TBLE	Thin-Boundary Layer Equations
TKE	Turbulence Kinetic Energy $K = 1/2(\langle u'u' \rangle + \langle v'v' \rangle + \langle w'w' \rangle)$
TLM	Two-Layer Model
WME	Wall-Modeling via function Enrichment
WMLES	Wall-Modeled Large-Eddy Simulation
WRLES	Wall-Resolved Large-Eddy Simulation
XFEM	eXtended Finite Element Method

Dimensionless Numbers

Cr	Courant number
D	diffusion number
Ma	Mach number
Re	Reynolds number
Re_τ	friction Reynolds number
Re_H	Reynolds number based on length scale H

Greek Symbols (Lowercase)

α	coarsening factor ($\alpha = 3$)
α_0, α_1, α_2	time integrator constants
β	blending factor ($\beta = 6$)
β_0, β_1, β_2	time integrator/extrapolation constants
γ	mesh stretching factor
γ_0	time integrator constant
δ	channel half-height/boundary layer thickness
ε	mathematical symbol
$\tilde{\varepsilon}$	dissipation rate
$\boldsymbol{\epsilon}$	rate-of-deformation tensor
ζ_C, ζ_D, ζ_C^*, ζ_D^*	proportionality factors in stabilization parameter
η	Kolmogorov length scale
κ	von Kármán constant ($\kappa = 0.41$)
$\varkappa$	wave number
λ^+, λ^-	maximum eigenvalue of flux Jacobian
λ_h	maximum eigenvalue of local generalized eigenvalue problem
λ_ν	viscous length scale
μ_h	weighting function
ν	kinematic viscosity
ν_t	eddy viscosity
ν_t^+	eddy viscosity ratio $\nu_t^+ = \nu_t/\nu$
$\widetilde{\nu}$, $\widetilde{\nu}_h$	eddy-viscosity-like working variable of SA model
$\widetilde{\nu}_0$	initial condition of eddy-viscosity-like working variable of SA model
$\widehat{\nu + \nu_t}$	modified viscosity due to stability considerations
$\boldsymbol{\xi}$	spatial coordinate vector in parameter space
ρ	density
τ_w, $\tau_{w,h}$	wall shear stress
$\boldsymbol{\tau}_w$	wall shear stress vector, components in Cartesian coordinate directions
τ_{IP}	interior penalty stabilization parameter
τ_C	continuity penalty stabilization parameter

τ_D	div-div penalty/grad-div stabilization parameter
τ_M	SUPG, PSPG stabilization parameter
$\boldsymbol{\tau}_{\text{LES}}$	LES subgrid tensor
$\boldsymbol{\tau}_{\text{RANS}}$	Reynolds stress tensor
$\boldsymbol{\tau}_H$	subgrid tensor of hybrid-filtered equations
ϕ	scalar quantity
χ	SA model quantity
ψ	enrichment function
ψ_h	enrichment function interpolated on Gauss–Lobatto nodes
$\boldsymbol{\omega}$, $\boldsymbol{\omega}_h$	vorticity vector

Greek Symbols (Uppercase)

Λ	Lax–Friedrichs flux quantity
$\boldsymbol{\Pi}^*$	velocity jump function
$\boldsymbol{\Omega}$	vorticity tensor
Ω, Ω_h	spatial domain
Ω_e^-, Ω_e^+	element domain
$\widetilde{\Omega}$, $\widetilde{\Omega}_h$	enriched subdomain
$\partial\Omega$, $\partial\Omega_h$	domain boundary
$\partial\Omega^D$, $\partial\Omega_h^D$	Dirichlet boundary
$\partial\Omega^N$, $\partial\Omega_h^N$	Neumann boundary
$\partial\Omega_e^-$, $\partial\Omega_e^+$	element boundary
$\partial\Omega_e^\Gamma$	interior element boundary
$\partial\widetilde{\Omega}^I$	boundary between enriched and nonenriched elements

Roman Symbols (Lowercase)

$a(q_h, \hat{\boldsymbol{u}}_h)$	right-hand side of Poisson equation
$b(\boldsymbol{v}_h, p_h^{n+1})$	right-hand side of projection step
b_1, b_2	gain in controller
c_{b1}, c_{b2}, c_{b3}	SA model constant
c_D	drag coefficient
c_f	skin friction coefficient
c_L	lift coefficient
c_p	pressure coefficient
c_{v1}	SA model constant
c_ν	multifractal subgrid scales model constant
c_{w1}, c_{w2}, c_{w3}	SA model constant
d	number of space dimensions
e	element number
$e^{\text{continuity}}$	error in mass conservation

$\boldsymbol{f}$	volume force
f_1, f_2, f_3	Cartesian components of volume force
f_{v1}, f_{v2}	SA model function
f_w	SA model function
f_d	DDES model function
$g(x_1)$	scalar function defining hill shape of periodic hill flow
g	SA model function
$\boldsymbol{g}_{\tilde{\nu}}$	boundary condition of eddy viscosity working variable of SA model
$\boldsymbol{g}_p$	pressure boundary condition
$\boldsymbol{g}_{\boldsymbol{u}}$	velocity Dirichlet boundary condition
h	grid length
h_1, h_2, h_3	grid length in corresponding space dimension
h_e	characteristic cell size
$\boldsymbol{h}$	Neumann boundary condition
k	polynomial degree of discrete function space
l	polynomial degree of weighting of enrichment function
l_{mix}	mixing length
$\ell(\boldsymbol{v})$	right-hand side variational form of incompressible Navier–Stokes equations
m	polynomial degree of function space of discrete wall distance and wall shear stress
n	time step number
$\boldsymbol{n}$	normal vector
n_1, n_2, n_3	Cartesian components of normal vector
$\boldsymbol{n}_\Gamma$	normal vector pointing from Ω_e^- to Ω_e^+
n_q	number of quadrature points
p	kinematic pressure
p_h	discrete pressure variable
$\bar{p}$	LES-filtered pressure
q, q_h	weighting function
s	sign of viscous interior penalty face term, yielding SIPG or NIPG
r	SA model function
r_d	DDES model function
r_h	ramp function
$\boldsymbol{r}^{\boldsymbol{v}}$	right-hand side of incremental Picard-iteration scheme, rows corresponding to momentum equation
$\boldsymbol{r}^q$	right-hand side of incremental Picard-iteration scheme, rows corresponding to continuity equation
$\boldsymbol{r}_h^{\boldsymbol{v}}$	right-hand side residual vector
$\boldsymbol{r}_{\alpha h}^{\boldsymbol{v}}$	coarsened right-hand side residual vector
$\boldsymbol{r}_{\parallel B}^{\boldsymbol{v}}$	wall-parallel component of right-hand side residual vector
$\boldsymbol{r}_i^{n+1}$	right-hand side vector of residual of incremental Picard-iteration scheme
t	time
Δt	time step size
Δt^n	time step size between t^n and t^{n+1}

Δt_{aver}	statistical averaging time
Δt^{NS}	time step size of Navier–Stokes steps
Δt^{SA}	time step size of SA step
$\boldsymbol{u}$	velocity field vector
$\boldsymbol{u}_0$	initial velocity field vector
u_1, u_2, u_3	Cartesian components of velocity vector
$\boldsymbol{u}_h$	discrete velocity field vector
$\boldsymbol{u}_B$	velocity degrees of freedom of node B
$\bar{\boldsymbol{u}}_B$	polynomial velocity degrees of freedom of node B
$\widetilde{\boldsymbol{u}}_B$	enrichment velocity degrees of freedom of node B
$\bar{\boldsymbol{u}}$	LES-filtered velocity field vector
$\bar{\boldsymbol{u}}_h$	polynomial component of discrete velocity vector
$\widetilde{\boldsymbol{u}}_h$	enrichment component of discrete velocity vector
$\hat{\boldsymbol{u}}$, $\hat{\boldsymbol{u}}_h$	first intermediate velocity
$\hat{\hat{\boldsymbol{u}}}$, $\hat{\hat{\boldsymbol{u}}}_h$	second intermediate velocity
u^+	dimensionless streamwise velocity
u_τ	friction velocity
u_b	bulk velocity
u'	RMS velocity of u_1
$u'v'$	Reynolds shear stress
$\boldsymbol{v}$, $\boldsymbol{v}_h$	weighting function
v'	RMS velocity of u_2
v_h, w_h	solution and weighting function for local eigenvalue problem
w^-, w^+	weight of averaging operator dependent on material parameter, e.g., $\nu_t^{\{-,+\}}$
w'	RMS velocity of u_3
$\boldsymbol{x}$	Cartesian coordinate vector
x_1, x_2, x_3	Cartesian components of coordinate vector
Δx^+	dimensionless grid spacing in streamwise direction
Δx_e^+	dimensionless width of whole element in wall-parallel direction
$x_{1,\text{reatt}}$	location of reattachment point from hill crest
$x_{1,\text{sep}}$	location of separation point from hill crest
y	wall-normal coordinate
y^+	dimensionless wall coordinate
Δy^+	dimensionless grid spacing in wall-normal direction
Δy_{1e}^+	dimensionless width of first off-wall element in wall-normal direction
$(\Delta)y_1^+$	dimensionless location of first off-wall grid point
$(\Delta)y_{1,\text{GL}}^+$	dimensionless location of first off-wall Gauss–Lobatto point
y_c^+	dimensionless grid spacing in channel center in wall-normal direction
y_{max}^+	algorithmic constant for enrichment adaptation algorithm in terms of the dimensionless wall coordinate
y_{DES}	modified wall distance function for DES
y_{DDES}	modified wall distance function for DDES
y_h	discrete wall distance
Δz^+	dimensionless grid spacing in spanwise direction

$\Delta \boldsymbol{z}_{i+1}^{n+1}$	incremental solution vector of Picard-iteration scheme

Roman Symbols (Uppercase)

A^+	model constant of algebraic RANS model
$A(\partial\Omega_e^\Gamma)$	element surface area
$\boldsymbol{A}$	divergence operator of matrix-form
A_B	nodal area
B	constant offset in log-law ($B = 5.17$)
B	coefficient of multifractal subgrid scale model
$\boldsymbol{B}$	gradient operator
$\mathcal{B}_{\text{NS}}(\boldsymbol{v}, q; \boldsymbol{u}, p)$	variational form comprising left-hand side of Navier–Stokes equations
$\mathcal{B}_{\text{NS}}^{\text{lin}}(\boldsymbol{v}, q; \boldsymbol{u}', p')$	variational form of terms linear with respect to unresolved scale quantities
$\boldsymbol{C}$	continuity penalty matrix
C_{DES}	DES model constant
C_{sgs}	model constant of multifractal subgrid scales
$\mathcal{C}(\boldsymbol{v}; \boldsymbol{u}_h, \boldsymbol{u}')$	variational form of cross stress terms
$\mathscr{C}(q, \boldsymbol{u})$	variational form of velocity divergence
$\boldsymbol{D}$	div-div penalty matrix
$E(\varkappa)$	energy spectrum
$\boldsymbol{F}$	matrix-form vector of body force
$\boldsymbol{F}^c$	convective flux matrix
$\widetilde{\boldsymbol{F}}^c$	convective flux matrix of SA model
$\boldsymbol{\mathcal{F}}^c$	convective flux
$\boldsymbol{\mathcal{F}}^{c*}$	numerical convective flux function
$\widetilde{\boldsymbol{\mathcal{F}}}^c$	convective flux of SA model
$\boldsymbol{F}^\nu$, $\boldsymbol{F}_{\text{BC}}^\nu$	viscous flux matrix and boundary terms
$\widetilde{\boldsymbol{F}}^{\widetilde{\nu}}$	diffusive flux matrix of SA model
$\boldsymbol{\mathcal{F}}^\nu$	viscous flux (based on molecular viscosity)
$\boldsymbol{\mathcal{F}}^{\nu*}$	numerical viscous flux function
$\widetilde{\boldsymbol{\mathcal{F}}}^{\widetilde{\nu}}$	diffusive flux of SA model
$\boldsymbol{\mathcal{F}}^{\nu_t}$	eddy viscosity contribution to viscous flux
$\boldsymbol{\mathcal{F}}^{\nu+\nu_t}$	total viscous flux based on molecular and eddy viscosity
$\boldsymbol{\mathcal{F}}^{\nu+\nu_t*}$	numerical viscous flux function including molecular and eddy viscosity
$\mathscr{F}^c(\boldsymbol{v}, \boldsymbol{u})$	variational form of convective term
$\mathscr{F}^\nu(\boldsymbol{v}, \boldsymbol{u})$	variational form of viscous term
$\mathscr{F}^{\nu+\nu_t}(\boldsymbol{v}, \boldsymbol{u})$	variational form of viscous term including eddy viscosity
$\mathscr{F}_e^\nu(\bar{\boldsymbol{v}}, \widetilde{\boldsymbol{v}}, \bar{\boldsymbol{u}}, \widetilde{\boldsymbol{u}})$	variational viscous multiscale term
$\boldsymbol{G}_{\boldsymbol{u}}$	vector of velocity boundary values
$\boldsymbol{G}_p$	vector of pressure boundary values
J	order of accuracy of BDF and EX scheme
$\boldsymbol{J}$	Jacobian
J_p	order of accuracy of pressure boundary condition

K	turbulence kinetic energy
$\boldsymbol{K}^{\bar{V}\bar{U}}$	projection matrix block corresponding to polynomial rows and columns
$\boldsymbol{K}^{\bar{V}\widetilde{U}}$	projection matrix block corresponding to polynomial rows and enrichment columns
$\boldsymbol{K}^{\widetilde{V}\bar{U}}$	projection matrix block corresponding to enrichment rows and polynomial columns
$\boldsymbol{K}^{\widetilde{V}\widetilde{U}}$	projection matrix blocks corresponding to enrichment rows and columns
$\boldsymbol{K}_i^{n+1}$	linearized system matrix
L	characteristic length
$\boldsymbol{L}$, $\boldsymbol{L}_{\mathrm{BC}}$	matrix of pressure Laplace operator
$\boldsymbol{M}$	mass matrix
$\widetilde{\boldsymbol{M}}$	mass matrix of SA model
$\boldsymbol{M}^{\bar{V}\bar{U}}$	mass matrix block corresponding to polynomial rows and columns
$\boldsymbol{M}^{\bar{V}\widetilde{U}}$	mass matrix block corresponding to polynomial rows and enrichment columns
$\boldsymbol{M}^{\widetilde{V}\bar{U}}$	mass matrix block corresponding to enrichment rows and polynomial columns
$\boldsymbol{M}^{\widetilde{V}\widetilde{U}}$	mass matrix blocks corresponding to enrichment rows and columns
$\mathscr{M}(\boldsymbol{v}, \boldsymbol{u})$	variational form of mass matrix
N	total number of grid points/nodes
N	number of cascade steps in multifractal subgrid model
$\widetilde{\boldsymbol{N}}$	vector of degrees of freedom of SA-eddy viscosity working variable
N_e	number of elements
N_{e1}, N_{e2}, N_{e3}	number of elements in corresponding space dimension
N^k, N^l	set of discontinuous Lagrangian shape functions of degree k or l
$N^{c,l}$, $N^{c,m}$	set of continuous Lagrangian shape functions of degree l or m
N_B^k, N_B^l	B^{th} discontinuous Lagrangian shape function of degree k or l
$N_B^{c,l}$, $N_B^{c,m}$	B^{th} continuous Lagrangian shape function of degree l or m
N_{lnm}^k	shape functions defined through tensor product
N^{SA}	number of sub-cycle time steps of SA model
$\boldsymbol{P}$	vector of pressure degrees of freedom
$\boldsymbol{\mathcal{P}}^*$	pressure interior penalty flux function
$P_k(\Omega_e)$	space of polynomials up to tensor degree k
$\boldsymbol{P}_{\alpha h}^h$	prolongation matrix
$\mathscr{P}(\boldsymbol{v}, p)$	variational form of pressure gradient
$\boldsymbol{R}_h^{\alpha h}$	restriction matrix
$\boldsymbol{R}_{M,h}$	discrete momentum residual
$\boldsymbol{R}_{C,h}$	discrete continuity residual
$\boldsymbol{S}_h^{\alpha h}$	scale separation operator
Q	Q-criterion variable
Q	volume flux
Q^0	nominal volume flux
$Q(\boldsymbol{u}, \widetilde{\nu})$	SA source term

$\boldsymbol{Q}$	SA source term vector of matrix form
$\bar{\boldsymbol{R}}$	polynomial part of vector
$\widetilde{\boldsymbol{R}}$	enrichment part of vector
$\mathcal{R}(\boldsymbol{v};\boldsymbol{u}')$	variational form of Reynolds stress terms
$S, \widetilde{S}$	SA model function
$\mathcal{T}$	simulation time
T_{wall}	wall time
T^*	flow-through time
U	characteristic velocity
$\boldsymbol{U}$	vector of velocity degrees of freedom
$\hat{\boldsymbol{U}}, \hat{\hat{\boldsymbol{U}}}$	vector of first and second intermediate velocity degrees of freedom
$\bar{\boldsymbol{U}}$	vector of polynomial velocity degrees of freedom
$\widetilde{\boldsymbol{U}}$	vector of enrichment velocity degrees of freedom
$V(\Omega_e)$	element volume
$\mathcal{V}_h^p$	discrete pressure space
$\mathcal{V}_h^{\boldsymbol{u}}$	discrete velocity space
$\mathcal{V}_h^{\bar{\boldsymbol{u}}}$	polynomial component of discrete velocity space
$\mathcal{V}_h^{\widetilde{\boldsymbol{u}}}$	enrichment component of discrete velocity space
$\mathcal{V}_h^{\widetilde{\nu}}$	discrete space of working variable of SA model
W	width
H	height

Mathematical Operators

$\lvert \cdot \rvert$	absolute value
$\lVert \cdot \rVert$	vector norm
$[\![\cdot]\!]$	jump operator
$[\cdot]$	jump operator
$[\cdot]^{ND}$	jump operator with modification on Dirichlet and Neumann boundaries
$\{\{\cdot\}\}$	weighted average
$\{\{\cdot\}\}^{ND}$	weighted average with modification on Dirichlet and Neumann boundaries
$\lfloor \cdot \rfloor$	round to next smaller integer
$\lceil \cdot \rceil$	round to next larger integer
$\mathrm{sgn}(\cdot)$	sign function

Subscripts, Superscripts, and Accents

$(\cdot)_h$	resolved quantity associated with characteristic mesh length scale h
$(\cdot)_{\alpha h}$	coarse resolved scale associated with characteristic mesh length scale αh for multifractal subgrid scales model
$\delta(\cdot)_h$	fine resolved scale for multifractal subgrid scales model

$\bar{(\cdot)}_h$	polynomial component of resolved quantity
$\widetilde{(\cdot)}_h$	enrichment component of resolved quantity
$(\cdot)_H$	quantity filtered by two-/three-level Germano filter
$\bar{(\cdot)}$	LES-filtered quantity
$\langle\cdot\rangle$	statistical ensemble-averaged quantity
$(\cdot)'$	temporal fluctuation
$(\cdot)'$	subgrid-scale quantity
$(\cdot)^n$	quantity at time step number n
$(\cdot)^-$	current element
$(\cdot)^+$	neighboring element
$(\cdot)^+$	dimensionless quantity in wall units
$(\cdot)_{\text{end}}$	value at end of simulation
$(\cdot)_{\text{ext}}$	extrapolated vector
$(\cdot)_{\text{max}}$	maximum value
$(\cdot)_{\text{min}}$	minimum value
$(\cdot)^{\text{new}}$	discrete velocity vector based on shape functions of new time step
$(\cdot)^{\text{old}}$	discrete velocity vector based on shape functions of old time step
$(\cdot)_{\text{ref}}$	reference value

1

Introduction

Fluid dynamics is one of the most important physical phenomena in engineering and science. Essentially, all engineering products are surrounded by fluid, most commonly air or water, and the resulting fluid dynamic forces govern the design process of these products. Fluid dynamics is not just a spurious phenomenon but necessary for the core functionality in most machines or devices. It enables aircraft to fly, wind energy power plants to produce electricity, and the cooling of engines, batteries, as well as microprocessors, just to name a few applications. Fluid dynamics is also a key factor in the current transition to a sustainable and CO_2-neutral energy economy due to its high relevance in the power-generation and transportation sectors. Such applications include cars, trains, planes, nautical vessels, all kinds of turbomachinery and fans, as well as heat exchanging devices. One question that arises: How can we create products that achieve the highest level of efficiency?

1.1. Motivation

The field of computational fluid dynamics (CFD) provides numerical tools that enable the prediction of three-dimensional flow including the forces of lift and drag. For the design of a product, we have to be able to compute the behavior of the fluid as accurately as possible, but the computations are also time critical in order to evaluate a large number of designs. These two requirements constitute opposing demands on the numerical approach employed for the simulations. On the one hand, direct numerical simulation (DNS) may be used to compute all turbulent motions of the flow in three space dimensions, which gives the highest possible fidelity. However, the computational effort required for such simulations is so high that the simulations take on the order of months and years of computation time, and DNS of most industrial applications with a high Reynolds number is not feasible today even on the largest supercomputers. With such high computational demands, it is not possible to compute a sufficient number of designs in a product development cycle. On the other hand, all turbulent scales may be modeled by using a Reynolds-averaged Navier–Stokes (RANS) model, which often allows the use of simplified two-dimensional geometries and steady-state solvers, providing the fastest solution times. However, the modeling errors of RANS can be severe, such that the best design of an industrial product according to the computational model is not necessarily equal to the best design in reality. A compromise between these two methodologies is to compute only the larger, energetic, anisotropic, turbulent structures of the flow, while the smaller homogeneous ones are modeled according to the paradigm of large-eddy simulation (LES). Owing to the continuous increase of available computing resources, this methodology is currently starting to become feasible for industrial applications at moderate Reynolds numbers, for example LES of internal turbomachinery flows [264]. One goal of the present work is to propose and validate accurate, robust, and efficient numerical methods for DNS, LES, and RANS.

The presence of boundary layers on solid walls significantly complicates the simulation of turbulent flows and makes the balance of the aforementioned requirements of accuracy and efficiency much more difficult, especially at high Reynolds numbers. With respect to LES, the sizes of the energetic turbulent motions vary drastically within the boundary layer due to the principal kinematic condition that the size of the largest turbulent structures is proportional to their distance from the wall [127]. This means that LES has, in the vicinity of walls, essentially similar computational requirements as DNS. With increasing Reynolds number, the cost of resolving the near-wall region dominates the cost of the entire simulation and makes an application of LES infeasible for, e.g., aircraft or wind turbines. Solutions to this problem are sought in the field of wall modeling. Another, essentially independent, challenge in the near-wall area is a very high velocity gradient just at the wall, which comes along with high computational demands in all flow simulation approaches.

Several wall modeling approaches exist, but all of them show drawbacks in at least one flow condition. For example, the near-wall region may be computed using RANS and the region further away from the wall via LES, according to the detached-eddy simulation (DES) methodology [240]. The issue in this approach is that the turbulent boundary-layer motions develop in a nonphysical way in the region between the RANS and LES zones. This problem frequently yields inaccurate results in attached boundary layers. As an alternative, the near-wall region may be modeled by use of wall functions, which apply synthetic boundary conditions at an off-wall location [168]. This method performs well in attached boundary layers, but no wall function exists that can accurately represent general nonequilibrium flows. In this work, a novel approach to wall modeling is developed, which substantially reduces the computational effort of turbulence-resolving simulations, while maintaining the high accuracy and consistency of LES.

Besides the aforementioned requirements for the development of efficient CFD tools, short computation times and high accuracy, two additional requirements have shown to be essential: the stability of the numerical method and the applicability to complex geometries. The requirement of stability is mandatory because an unstable numerical method would not produce trustworthy results if it gives results at all. In addition, it is the ambition of this work to contribute to and propose novel numerical methods that are sufficiently general in order to be applied to real-world industrial problems. These problems consist of complex bodies possibly with curved boundaries, thus demanding simulation tools that are capable of handling complex geometries.

1.2. Contribution of this Work

This work proposes and further develops numerical methods in the field of CFD. These computational tools fulfill the four previously formulated requirements of accuracy, low computational cost, stability, and applicability to complex geometries. The main contributions of this thesis are the following:

Wall Modeling via Function Enrichment. Wall modeling via function enrichment is proposed in this work as a methodology that drastically reduces the computer time and storage requirements of CFD tools (see Krank and Wall [153, 154], Krank et al. [150–152], and Kronbichler et al. [159, 160]). The main idea is that the underlying interpolation stencils of the numerical method are tailored for the specific application of turbulent wall-bounded flow. With such a scheme available, the numerical method is much more efficient in computing turbulent boundary layers in comparison to standard methods, without noticeable drawbacks in accuracy. The method is much more accurate than the previously mentioned wall-function approaches and computationally less expensive than DES.

Since wall modeling via function enrichment is a spatial discretization technique only, the topic of modeling the unresolved turbulence in the near-wall region has to be addressed in addition. To this end, four different turbulence modeling approaches for an application with wall modeling via function enrichment have been developed in this thesis:

- Wall modeling via function enrichment may be used in conjunction with the classical RANS methodology in order to reduce the computational cost arising through the high near-wall velocity gradient present at high-Reynolds-number applications (see Krank et al. [152]).

- Wall modeling via function enrichment is employed in order to reduce the required number of grid points in the near-wall region in DES (see Krank et al. [151]).

- Wall modeling via function enrichment can be used in order to construct a novel multiscale approach for the simulation of turbulent boundary layers (see Krank et al. [150]). The numerical method resolves eddies via LES where the mesh is sufficiently fine. Additionally, a RANS layer overlaps with the LES inside the near-wall region, where the coarseness of the mesh does not allow the resolution of the velocity profile. The tailored stencil enables coarse discretization cells at the wall, so the approach can be best described as wall-modeled LES (WMLES). The model is constructed in such a way that it does not show the problems of DES in the RANS–LES transition region.

- Wall modeling via function enrichment can be used in conjunction with several alternative turbulence modeling approaches. An approach additionally considered due to historical reasons at the Institute for Computational Mechanics is a residual-based variational multiscale method, which is supported by a multifractal subgrid scale model [215] in the bulk flow (see Krank and Wall [153, 154]).

Wall modeling via function enrichment can be included in any Galerkin scheme, such as the continuous finite element method (FEM) and the discontinuous Galerkin (DG) method. As part of this thesis, the wall model has been implemented in two independent simulation programs: a continuous FEM incompressible flow solver available in the simulation software BACI [258] developed at the Institute for Computational Mechanics at the Technical University of Munich and a new high-order incompressible DG solver [71], which has been developed further in this work (see Krank et al. [147]).

A Stabilized High-Order Incompressible DG Scheme for Turbulent Flow. Another major contribution of this thesis is the further development of a high-order DG method for incompressible flow in close collaboration with a second PhD student Niklas Fehn under the supervision of Dr. Martin Kronbichler at the Institute for Computational Mechanics. The results of this collaboration have been published in a joint publication (see Krank et al. [147]). The novelty is a modification of the scheme, which stabilizes issues in mass conservation and small time steps reported in earlier works on high-order incompressible DG methods (see, e.g., [73, 246]), and a new adaptive time stepping algorithm. This numerical method enables one of the first applications of an incompressible high-order DG scheme to underresolved 3D turbulent flow (implicit LES, ILES) (see Krank et al. [147, 149] and Kronbichler et al. [159]).

A New High-Order DG Solver for RANS and DES. The high-order DG method has been extended to the incompressible RANS equations based on the Spalart–Allmaras (SA) model (see Krank et al. [152]). This code is extended to DES (see Krank et al. [151]) as well.

High-Fidelity Reference Data of Flow over Periodic Hills. In order to allow for the validation and quantitative assessment of the preceding methodological developments, the use of canonical benchmark flows is essential. One of the most widely used test cases is a flow over periodically arranged, smoothly curved hills; see, e.g., [93]. Due to significant discrepancies between existing reference databases for this flow, the present high-order DG method was employed to compute new reference results via DNS with the highest fidelity to date at two Reynolds numbers (see Krank et al. [149]). This reference data has been made available on a public repository (see Krank et al. [148]) and is extensively used in the remainder of this thesis.

As indicated, most of this work has already been published in peer-reviewed international journals. The articles that have already been published are used in the remainder of this thesis with permission.

1.3. Thesis Structure

The structure of this thesis is organized as follows. Chapter 2 provides an introduction to the fundamental concepts related to turbulent flows and the available principle approaches in CFD. The remaining chapters are divided into two parts. In Part I, the numerical methods for wall-resolved CFD without specific treatment of the near-wall region are described. Then, the approach of wall modeling via function enrichment and its application in different turbulence modeling scenarios is proposed in Part II of this thesis.

Chapters 3 to 5 form Part I of the thesis. Chapter 3 presents a high-order incompressible DG code and proposes the modifications that stabilize the scheme in underresolved simulations. This numerical method is applied to large-scale DNS and ILES of two canonical benchmark examples, each at several Reynolds numbers, in Chapter 4. Chapter 5 extends the numerical method to RANS by use of the SA model and this implementation is applied to DES as well.

In Part II, we begin in Chapter 6 with the motivation, the mathematical formulation, and the efficient implementation of the new wall model. The subsequent chapters present the four applications using different turbulence models as mentioned earlier. In Chapter 7, wall modeling via function enrichment is applied in the context of the continuous FEM by use of a residual-based turbulence model. Chapter 8 shows how the wall model may be used within the high-order DG method in RANS by extending the RANS baseline solver presented in Chapter 5.

In Chapter 9, it is shown that the DES method provides a suitable turbulence model for the unresolved turbulent motions in wall modeling via function enrichment. Chapter 10 describes a novel multiscale approach to wall modeling, which circumvents the typical problems of hybrid RANS/LES methods that are visible in the results presented in Chapter 9.

Conclusions and suggestions for future research projects close the thesis in Chapter 11.

2

A Brief Introduction to Turbulence and its Numerical Simulation

As a basis for the CFD methods and models developed in this thesis, we introduce the primary concepts related to turbulent flows and their numerical simulation. The first section summarizes the governing equations as well as the phenomenological and statistical nature of turbulence with a particular focus on turbulent boundary layers. In Section 2.2, we give a brief overview of the most common discretization methods used in CFD and discuss arguments for selecting the DG method for the solvers presented in Chapters 3 and 5. Section 2.3 summarizes the available approaches to computing turbulent wall-bounded flows and discusses the range of applicability of each method. The approaches considered in the present thesis are DNS, LES, RANS, and hybrid RANS/LES including wall modeling. Wall modeling is identified as a crucial ingredient of an efficient CFD method targeting turbulent wall-bounded flows at high Reynolds number.

2.1. Turbulent Flow: A Multiscale Phenomenon

The first section of this chapter states the governing equations of fluid dynamics as considered in the present work (Section 2.1.1). Subsequently, a brief overview of the phenomenological nature of turbulence is given in Section 2.1.2 including turbulent boundary layers in Section 2.1.3.

2.1.1. Incompressible Navier–Stokes Equations

The Mach number Ma is defined as the ratio of the fluid velocity magnitude to the speed of sound. If the Mach number is below 0.3, the flow may be considered incompressible and the density ρ is constant, allowing several simplifications of the Navier–Stokes equations. The research presented in this thesis focuses on incompressible flows. However, the concepts of wall modeling discussed herein are applicable to compressible and incompressible flows alike.

The incompressible Navier–Stokes equations are given in conservation form as

$$\frac{\partial \boldsymbol{u}}{\partial t} + \nabla \cdot \big(\boldsymbol{\mathcal{F}}^c(\boldsymbol{u}) + p\boldsymbol{\mathcal{I}} - \boldsymbol{\mathcal{F}}^\nu(\boldsymbol{u})\big) = \boldsymbol{f} \qquad \text{in } \Omega \times [0, \mathcal{T}] \tag{2.1}$$

with the incompressibility condition,

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \text{in } \Omega \times [0, \mathcal{T}], \tag{2.2}$$

ensuring mass conservation, where $\boldsymbol{u} = (u_1, u_2, u_3)^T$ is the velocity, p the kinematic pressure, $\mathcal{T}$ the simulation time, Ω the domain size, and $\boldsymbol{f} = (f_1, f_2, f_3)^T$ the body force vector. The convective flux is defined as $\boldsymbol{\mathcal{F}}^c(\boldsymbol{u}) = \boldsymbol{u} \otimes \boldsymbol{u}$ and the viscous flux as $\boldsymbol{\mathcal{F}}^\nu(\boldsymbol{u}) = 2\nu\boldsymbol{\epsilon}(\boldsymbol{u})$ with the symmetric rate-of-deformation tensor $\boldsymbol{\epsilon}(\boldsymbol{u}) = 1/2(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T)$, where ν is the kinematic viscosity. By use of the incompressibility condition (2.2), the convective term may be rewritten as

$$\nabla \cdot \boldsymbol{\mathcal{F}}^c(\boldsymbol{u}) = \boldsymbol{u} \cdot \nabla \boldsymbol{u}, \tag{2.3}$$

which is denoted by convective form of the convective term, and both the divergence and convective formulations of the convective term are considered in the present work. At $t = 0$, a divergence-free velocity field is imposed as initial condition with

$$\boldsymbol{u}(t = 0) = \boldsymbol{u}_0 \qquad \text{in } \Omega. \tag{2.4}$$

Boundary conditions on the Dirichlet and Neumann boundaries $\partial\Omega^D$ and $\partial\Omega^N$, with $\partial\Omega^D \cup \partial\Omega^N = \partial\Omega$ and $\partial\Omega^D \cap \partial\Omega^N = \emptyset$, are defined as

$$\boldsymbol{u} = \boldsymbol{g}_{\boldsymbol{u}} \qquad \text{on } \partial\Omega^D \tag{2.5}$$

and

$$(-p\boldsymbol{\mathcal{I}} + \boldsymbol{\mathcal{F}}^\nu(\boldsymbol{u})) \cdot \boldsymbol{n} = \boldsymbol{h} \qquad \text{on } \partial\Omega^N, \tag{2.6}$$

where the outward unit normal vector with respect to $\partial\Omega$ is denoted by $\boldsymbol{n}$.

A two-dimensional version of these equations, where the third entry in all vector-valued quantities is canceled, is also considered in some sections of this thesis. However, turbulence is a three-dimensional phenomenon by nature and turbulence-resolving simulations must therefore be performed using the three-dimensional Navier–Stokes equations.

2.1.2. Bulk Turbulence

The incompressible Navier–Stokes equations exhibit solutions of unexpected complexity. Two solution states may be observed: laminar and turbulent flow. While laminar flow is smooth, turbulent flow is irregular and chaotic. Whether a flow is laminar or turbulent may be determined by means of the Reynolds number, $\mathrm{Re} = UL/\nu$, with a characteristic velocity U and length scale L. Small Reynolds numbers indicate the dominance of viscous effects and therefore laminar flow. High Reynolds numbers are a sign of prevailing inertial forces, promoting turbulent flow conditions.

Turbulent flows manifest their chaotic nature through the presence of vortical structures, often with a multitude of spatial and temporal scales, and their complex nonlinear interaction. Eddies may be defined as regions, where the second invariant of the velocity gradient tensor,

$$Q = \frac{1}{2}\left(\boldsymbol{\Omega} : \boldsymbol{\Omega} - \boldsymbol{\epsilon} : \boldsymbol{\epsilon}\right), \tag{2.7}$$

with $\boldsymbol{\Omega} = 1/2(\nabla \boldsymbol{u} - (\nabla \boldsymbol{u})^T)$, is positive ($Q > 0$) [121]. This relation is the so-called Q criterion, which is widely used to visualize vortex tubes by rendering iso-surfaces of $Q > 0$. Such iso-surfaces of a developed turbulent flow, a zoom-in on the flow over periodic hills as investigated in Section 4.3, are visualized in Figure 2.1. The picture shows vortex tubes, which are in complex mutual interactions and in interactions with (invisible) smaller and larger eddies. The isosurfaces in the figure are colored by the velocity magnitude, and the bulk flow is directed

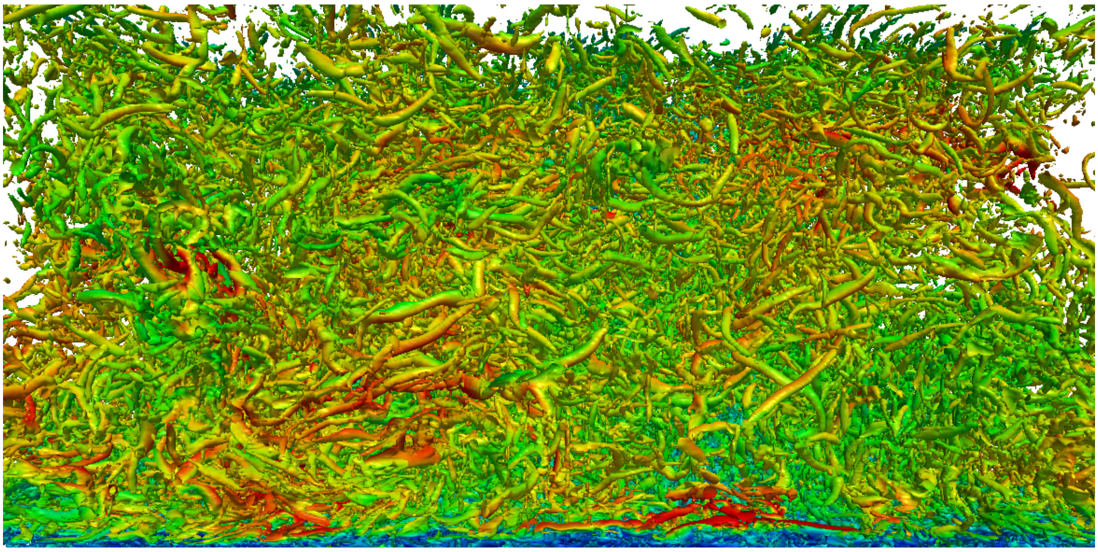


Figure 2.1.: Bulk turbulence visualized via Q criterion. Red indicates a high and blue low velocity magnitude.

into the plane. A strong gradient in color along the circumference of the vortex tubes reveals the significant circumferential velocity induced by the eddies.

A life-cycle of a typical eddy may be sketched as follows. A large vortex is generated by one of the many vortex generating mechanisms, for example via pressure-induced separation or natural boundary layer instability. Through convective interactions with other vortices, the eddy is stretched and folded iteratively, and each stretching-and-folding process reduces the size of the eddy, until it is dissipated by viscous forces. The statistical behavior of many eddies yields the characteristic energy spectrum $E(\varkappa)$ with the wave number $\varkappa$ as sketched in Figure 2.2. Herein, the largest, newly generated, vortices are the energy-carrying scales. The energy is then consecutively transferred to smaller scales in the inertial sub-range until the viscous forces become dominant and the energy is dissipated. The largest energy-carrying scales are typically of similar size as the boundary conditions and the length scale of the smallest vortices may be estimated by Kolmogorov's viscous length scale $\eta = (\nu^3/\widetilde{\varepsilon})^{1/4}$ [144], where $\widetilde{\varepsilon}$ is the rate of energy transfer. It is this multiscale nature of turbulence with broad spectra of length and time scales that makes the direct computation of turbulent flow so challenging and defines the need for turbulence modeling.

Turbulent flows exhibit numerous further phenomenological and statistical properties. For a comprehensive and detailed description, we refer to the textbooks by Pope [206] and Davidson [56]. One such special case is boundary layer turbulence, which is the topic of the next section.

2.1.3. Boundary Layer Turbulence

The presence of walls has a significant influence on the near-wall turbulence, since both the mean velocity and the velocity fluctuations must be zero at the wall, and a vortex size is physically limited by its distance to the no-slip boundary. As a consequence, the flow just at the wall is

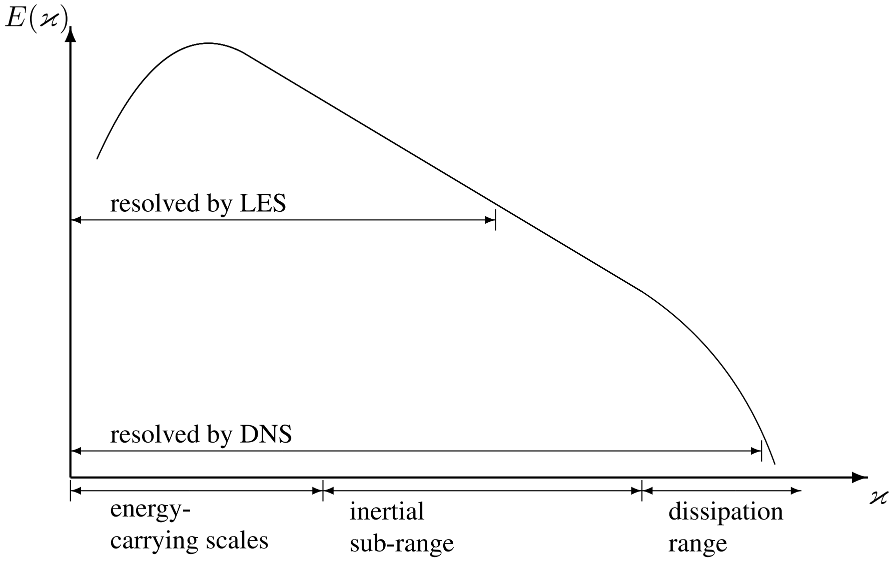


Figure 2.2.: Schematic of a characteristic energy spectrum resulting from the energy cascade.

laminar. In attached boundary layers, the turbulence beyond the laminar sublayer is dominated by two characteristic coherent turbulent structures: thin streamwise vortex tubes, so-called streaks, and hairpin vortices, an arch-like connection of two counter-rotating streaks, which moves away from the wall due to self-induced convection [221]. Such hairpin vortices are visualized in Figure 2.4 via the Q criterion for a plane boundary layer, showing a transitional stage of a channel flow at the friction Reynolds number of $\mathrm{Re}_\tau = u_\tau\delta/\nu = 2{,}000$ using the setup presented in Section 4.2.

Despite the complexity of the turbulent motions, a mean velocity profile may be derived for attached turbulent boundary layers in near-equilibrium conditions[1] with vanishing longitudinal pressure gradient on smooth walls. In the inner layer of the boundary layer, $y/\delta < 0.2$, with the wall distance y and the boundary layer thickness δ, the mean shear stress is approximately constant. Based on this assumption, the velocity is proportional to the wall distance inside the viscous sublayer, just at the wall. Further away from the wall, the viscous effects are negligible and the velocity profile obeys a logarithmic relation up to the outer edge of the inner layer, $y/\delta = 0.2$ [56]. We have

$$u^+ = \begin{cases} y^+ & y^+ < 5, \\ \frac{1}{\kappa}\ln(y^+) + B & y^+ > 40 \text{ and } \frac{y}{\delta} < 0.2, \end{cases} \tag{2.8}$$

and consider $\kappa = 0.41$ and $B = 5.17$ according to Dean [59] throughout this work. Herein, the dimensionless wall coordinate y^+ and the dimensionless velocity u^+ are defined as

$$y^+ = \frac{y u_\tau}{\nu}, \qquad u^+ = \frac{u}{u_\tau},$$

---

[1] The term 'boundary layer in equilibrium state' seems to originate from an analogy proposed by Clauser [47] of the turbulent boundary layer with a dynamic nonlinear mechanical system. A boundary layer is in equilibrium when the dimensionless output variables of the boundary layer system, for example the velocity profile, asymptotically tend to a steady-state with fixed input parameters. The term equilibrium boundary layer commonly refers to the case without pressure gradient, which yields the log-law as a velocity profile in the inner layer as presented herein. However, the discussion in [47] stresses that there are other equilibrium states including pressure gradients or roughness, which yield other velocity profiles.

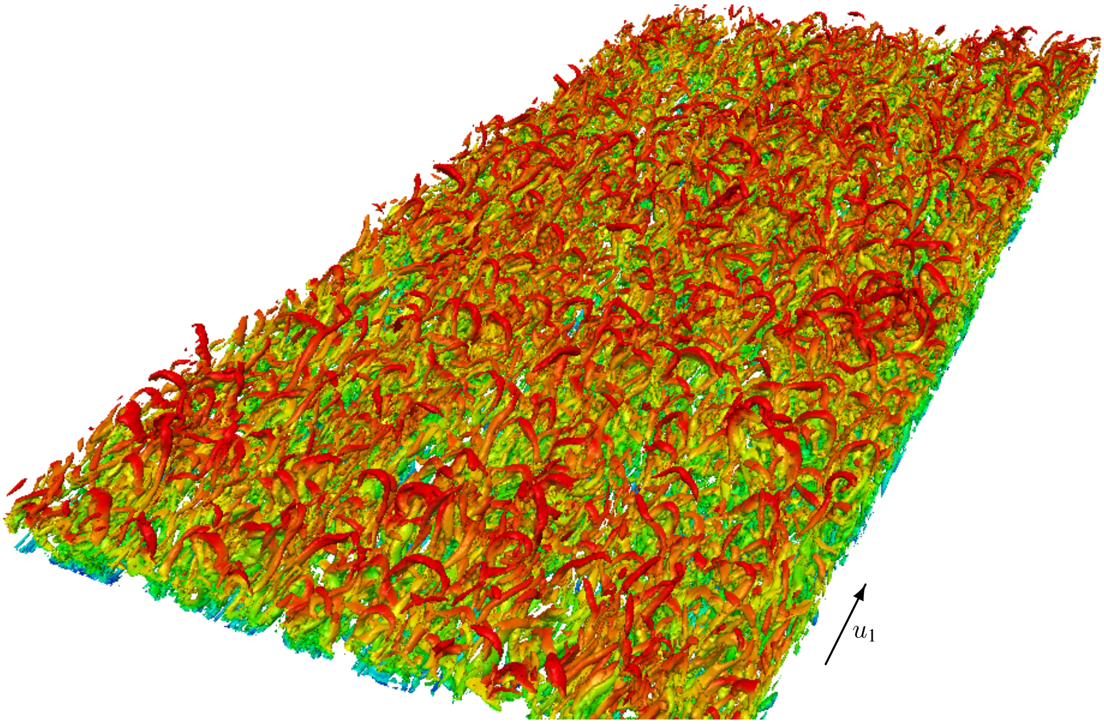


Figure 2.3.: Boundary layer turbulence visualized via Q criterion. Red indicates high and blue low velocity magnitude.

with the statistically averaged velocity u and the friction velocity $u_\tau = \sqrt{\tau_w/\rho}$, where $\tau_w = \rho\nu \left.\frac{\mathrm{d}u}{\mathrm{d}y}\right|_{y=0}$ is the wall shear stress and ρ the density. We note that the law of the viscous sublayer fulfills two boundary conditions, given as no-slip conditions ($u^+|_{y=0} = 0$) and the slope $\left.\frac{\mathrm{d}u+}{\mathrm{d}y+}\right|_{y=0} = 1$, where the latter is responsible for reproducing the correct wall shear stress [58].

The laws of the viscous sublayer and the logarithmic region are plotted in Figure 2.4. In the y^+ region between $5 \leq y^+ \leq 40$, both viscous and inertial forces are relevant and the velocity profile blends from the linear to the logarithmic profile in the so-called buffer layer. Several empirical wall laws have been proposed, which provide a continuous description for the velocity profile in the whole range of the linear and logarithmic region as well as the buffer layer. Probably the most widely known relation is Spalding's law [245], which is also plotted in Figure 2.4 and is discussed in Section 6.3.2.3 alongside several other wall functions. Further away from the wall, beyond $y/\delta \geq 0.2$, the velocity profile deviates from the log-law as the assumption of constant shear stress becomes inaccurate. Several modifactions have been proposed in order to account for the wake of the bulk flow, for example the defect law by Coles [51] or the correction by Dean [58]. For a detailed derivation and a further discussion of the analytical velocity profiles, it is again referred to the textbook by Davidson [56]. The logarithmic character of the velocity profile in the inner layer results in extremely sharp velocity gradients just at the wall, and the mean velocity gradient in the viscous sublayer scales with Re_τ. Therefore, the velocity gradient may become very high and a significant part of the velocity drop towards the no-slip boundary "takes place below $y^+ = 80$" [127].

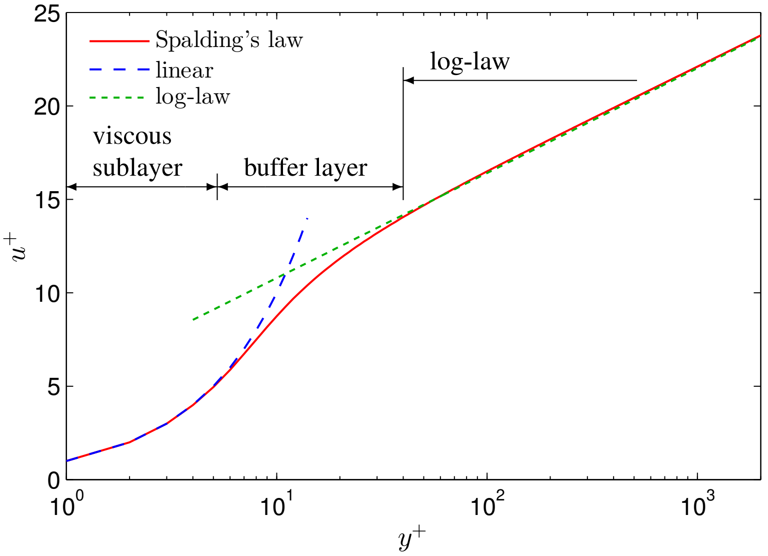


Figure 2.4.: Boundary layer profile consisting of a linear and a logarithmic region, as well as a buffer layer.

What size do the characteristic energy-carrying near-wall eddies have, in relation to the viscous length scale and the bulk turbulence? Jiménez provides a quantitative analysis to this question in a review article [127] and the relevant results are summarized. Figure 2.5 shows the typical sizes of energy-carrying and dissipating scales. Near the wall, the smallest structures and the energy-containing scales are of similar length. In the outer layer, the length scale ratio between the largest and the smallest scales is of the order of $\mathrm{Re}_\tau/100$ wall units, so the ratio increases substantially with increasing Reynolds number. In the logarithmic layer, the size of the energy-carrying eddies is approximately proportional to the wall distance [127].

2.1.4. The Multiscale Nature of Turbulence: A Challenge to Simulation and Modeling

The key conclusions of the latter two sections are summarized in view of the present thesis. The nature of turbulence is governed by several multiscale phenomena and these are responsible for the challenges related to the simulation and modeling of turbulent flow:

- In the bulk of the flow, many different scales exist and the gap between energy-containing and viscous dissipation scales increases dramatically with the Reynolds number. However, it will be sufficient and affordable to compute the largest scales only and model the small scales (see Section 2.3.2).

- Near walls, the largest scales are of similar size as the viscous dissipation scales, rendering a direct computation of these turbulent motions impossible in high-Reynolds-number engineering flows. Yet, the near-wall turbulence has to be resolved or modeled in order

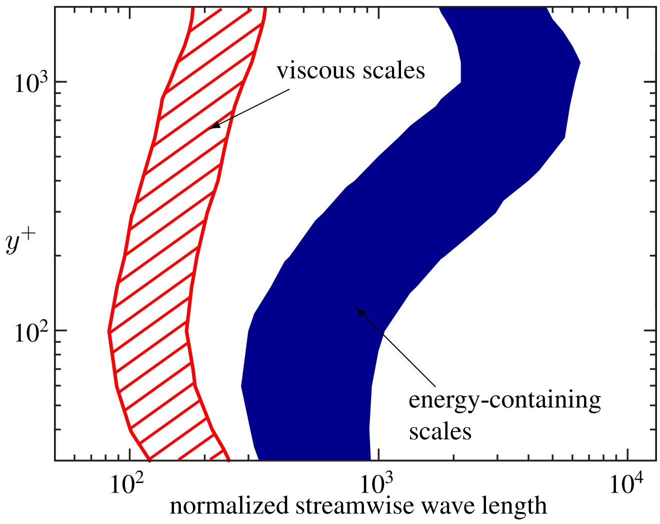


Figure 2.5.: Sketch showing the sizes of the energy-carrying and viscous dissipation scales in a turbulent channel flow at $\mathrm{Re}_\tau = 2{,}000$, based on data from Jiménez [127].

to predict engineering quantities such as the mean velocity profiles and the skin friction. Remedies to this challenge are sought in the field of wall modeling (see Section 2.3.4).

- As the mean velocity gradient at the wall increases with the Reynolds number, it is demanding to accurately compute this gradient. This is another, independent, and more general difficulty and has to be addressed by wall modeling techniques.

In the scope of this work, numerical methods and models are developed, which allow an accurate and fast computation of turbulent wall-bounded flows despite these three challenges.

2.2. Overview of Numerical Discretization Schemes

We give a brief overview of the most commonly used numerical schemes for the spatial discretization of the incompressible Navier–Stokes equations. The discussion is not intended to provide an exhaustive listing of characteristics and techniques, but merely the primary advantages and difficulties are mentioned, serving as a motivation for the numerical methods developed and promoted in the present work. We follow the arguments presented by Hesthaven and Warburton [112], Ferrer [73], and Landmann [166] and it is referred to these publications for further details.

In the first subsection, we introduce requirements on a numerical discretization scheme in view of the efficient computation of time-dependent flow problems. Subsequently, the four most common spatial discretization methods are presented and their characteristics are compared to the requirements.

2.2.1. Requirements on the Numerical Baseline Method

In the introduction (Section 1.1), four basic requirements for the development of an efficient computational method for the simulation of turbulent boundary layers were introduced: high accuracy, low computational cost, stability of the numerical method, and applicability to complex geometries. These requirements may be translated into requirements on the spatial discretization scheme. An efficient time stepping scheme enables a low computational cost. High-order spatial accuracy contributes to an accurate method. We further require the applicability to general geometries and the convective stability of the spatial discretization according to the stability criterion.

2.2.1.1. Efficient Time Stepping

For the discretization of space- and time-dependent partial differential equations, two principal procedures are available: The spatial and temporal domains may be discretized separately, in which case there is a large number of time stepping methods available. As an alternative, a discretization in space and time may be carried out simultaneously by the same numerical method. In Chapter 3, several alternatives are mentioned in the context of the DG method. In the case of an independent discretization of space and time, another subdivision into explicit and implicit time stepping methods may be considered. Explicit time integration schemes do not necessitate the solution of equation systems and exhibit short computation times per time step, but are subject to time step size limitations. Implicit time integration schemes on the other hand necessitate the solution of global equation systems. The question if explicit or implicit time stepping is more efficient in simulating eddy-resolving turbulent flows is the subject of ongoing research. In this work, we consider a semi-explicit time integration method of the incompressible Navier–Stokes equations, which exhibits a wide range of applicability without severe time step restrictions and is presented in Section 3.2.

Due to the explicit character of the time integration method, we prefer a spatial discretization that yields a local mass matrix, which is invertible without the cost of a global equation system.

2.2.1.2. High-Order Accuracy

Early numerical methods have mainly used low-order schemes of first to third order accuracy or linear and quadratic interpolation. The reason for this limitation may lie in the preconception that the cost of wide stencils in high-order methods could annihilate the improved accuracy. In Figure 2.6, the velocity error is plotted over the computational cost using the solver presented in the current work. It may be observed, that the efficiency of the numerical method increases dramatically with the polynomial degree, and thus the order of accuracy. In addition, the numerical dissipation of the discretization scheme decreases with increasing order of accuracy, which is beneficial in eddy-resolving simulations such as presented in Chapter 4. It is noted that high-order methods are expensive in implicit codes when the traditional matrix-based solution procedures are employed. In order to achieve good performance when using iterative solvers in conjunction with a high-order method, matrix-free solution procedures have to be employed. Such algorithms are used and developed in this thesis and exhibit solution times per degree of freedom (DOF), which are almost independent of the spatial order of accuracy.

2.2.1.3. Complex Geometries

Most engineering problems consist of complex geometries with curved boundaries, which cannot be meshed using structured grids without great effort. It is the aim of this work to develop

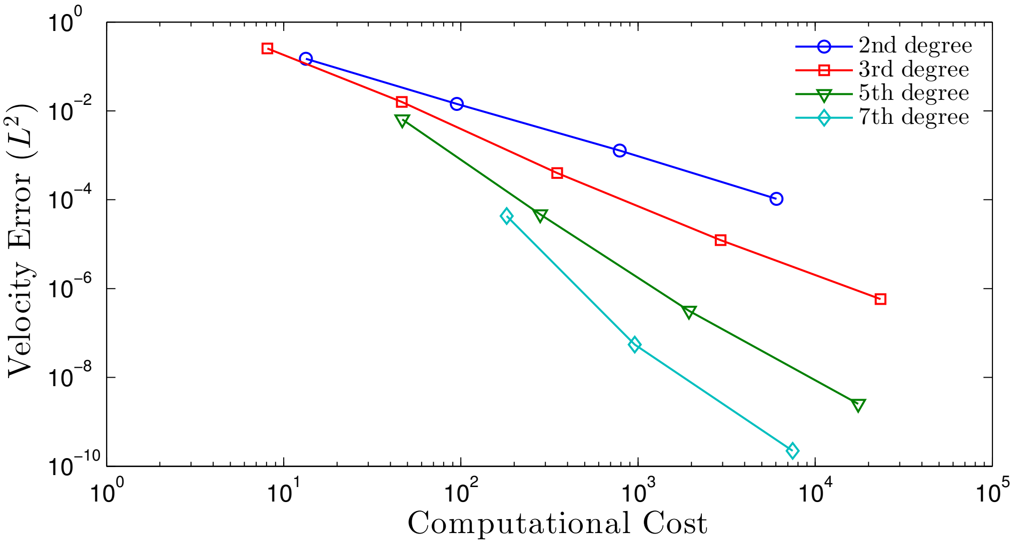


Figure 2.6.: Velocity error versus computational cost in the context of varying grid refinement and polynomial degree of the numerical method.

computational tools usable with complex geometries, and capable to utilize unstructured meshes. Due to this requirement on the numerical method, spectral Fourier methods, which are limited to periodic domains, are not considered.

2.2.1.4. Convective Stability

As the last requirement, we consider the ability to achieve a stable scheme in flows that are dominated by convection. The nonsymmetric character adhering problems with dominating convection causes stability issues in symmetric discretization schemes. We are looking for a numerical method, which provides efficient techniques to tackle this issue with low dissipation properties.

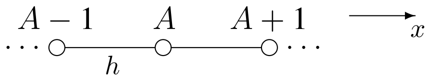
In addition to these four aspects, there is a large number of further requirements that may be relevant, see, for example, the extensive listings presented in [73, 166].

2.2.2. Numerical Methods

The four most widely used spatial discretization methods for flow problems are discussed in view of the formulated requirements.

2.2.2.1. The Finite Difference Method

In the finite difference (FD) method, the spatial domain is discretized by a structured mesh. All terms of the conservation law are explicitly computed on the nodes of the grid. The spatial derivatives are evaluated along one-dimensional stripes via FD stencils, which perform numerical differentiation based on neighboring nodes. For example, the simplest finite difference stencil for evaluation of the node A is based on the two neighboring nodes $A-1$ and $A+1$ in 1D,



allowing an evaluation of the first and second spatial derivatives of a quantity ϕ with

$$\frac{\partial \phi}{\partial x} \approx \frac{\phi_{A+1} - \phi_{A-1}}{2h}, \qquad \frac{\partial^2 \phi}{\partial x^2} \approx \frac{\phi_{A+1} - 2\phi_A + \phi_{A-1}}{h^2},$$

yielding a spatial accuracy of second order. It is straight-forward to extend this approach to higher accuracy by considering more consecutive points in the stencil. The FD approach is considered as one of the fastest methods for the solution of time-dependent partial differential equations, in particular for explicit schemes due to the locality of the mass matrix. Convection-dominated simulations can be treated using upwinding schemes, in which the information “upwind” of the current node is weighted stronger, providing an improved stability of the scheme. A severe limitation of the FD method is that it relies on neighboring nodes for the evaluation of the finite derivatives, which makes the extension to unstructured grids and complex geometries difficult. Since the capability of complex geometries was listed as an essential requirement in the introduction, the FD method is not considered in this thesis.

2.2.2.2. The Finite Volume Method

The finite volume (FV) method is a method that allows the use of unstructured meshes and enables complex geometries. Instead of considering nodes, this method fulfills the conservation laws in finite cells or volumes. The Navier–Stokes equations are integrated over one such volume and the divergence theorem is applied in order to obtain a balance in terms of fluxes across the cell boundaries. The capability of unstructured meshes is achieved by this construction, since no assumptions are made regarding the shape of each volume cell. In its simplest form, the solution is constant in each cell and the solution variables have to be reconstructed at the cell boundary during the evaluation of boundary integrals. If a linear interpolation is chosen, the resulting scheme is on structured meshes essentially equivalent to the classical FD method. In view of a high-order extension of the FV method, interpolation schemes spanning several volume cells would be necessary, similar to the FD method. However, such a high-order stencil would limit the FV method to structured grids; the high-order extension in conjunction with unstructured grids is challenging. Like with the FD method, the FV approach implies local mass matrices, which yield high performance in explicit schemes. The interpolation scheme may be chosen in such a way that the upwind information is weighted stronger, which again yields a method that is stable in the convective regime. Despite the lack of general higher-order capability, the FV method is today the ‘working horse’ in the industrial simulation of turbulent flow. The combined requirements of geometrical flexibility and high accuracy do not seem to be feasible with the FV method.

2.2.2.3. The (Spectral) Finite Element Method

The classical continuous FEM, also denoted by the continuous Galerkin method, is derived by the multiplication of the residual with a weighting function and integration over the whole computational domain, yielding the variational form. The domain is then subdivided into a finite number of elements, and these elements are, with some exceptions, of tetrahedral or hexahedral shape. Inside each cell, the solution and weighting functions are approximated by means of

Table 2.1.: Overview of numerical discretization schemes including advantages and disadvantages, similar to [112].

Method	Unstructured meshes	High-order capable	Mass matrix	Convective stabilization
FD	difficult	good	local	good
FV	good	difficult	local	good
FEM	good	good	global	difficult
Spectral FEM	good	good	local	difficult
(Spectral) DG	good	good	local	good

polynomials of the form:

$$\boldsymbol{u}_h(\boldsymbol{x},t) = \sum_{B\in N} N_B(\boldsymbol{x})\boldsymbol{u}_B(t), \tag{2.9}$$

with shape functions N_B and DOFs $\boldsymbol{u}_B$. The subscript $(\cdot)_h$ indicates the identification of the respective variable with a characteristic element length h. Despite the flexibility regarding unstructured grids, we have not made any assumption on the polynomial degree of the shape functions in each cell. The accuracy of the method per DOF increases generally with the polynomial degree, and the method is denoted by spectral element method if very high degrees are employed. If the shape functions N_B are so-called nodal shape functions, there are points within each element, where all but one shape functions are zero-valued, and these points are called nodes. On the boundary of each cell, the nodes are shared among the two neighboring elements. This connection couples the DOFs with neighboring cells, such that the mass matrix becomes a global matrix and a global matrix system has to be solved in an explicit time stepping scheme. It is noted that this drawback vanishes if an implicit time stepping method is employed. Also, the approximation of lumping the mass matrix provides an engineering solution to the problem; this technique is frequently used in explicit spectral element codes. The issue of convective stabilization is in the context of the FEM commonly treated by literally weighting the residual upstream with a higher weight than downstream trough a modification of the weighting function in the Streamline Upwind Petrov-Galerkin method (SUPG) [34]. The SUPG is quite costly in terms of computational effort in comparison to the other upwinding methods in FD and FV and comes along with a significant amount of artificial dissipation – often more than desired. Therefore, convective stabilization is a real challenge in the FEM, in particular in the spectral regime. Due to the drawbacks regarding the mass matrix and the convective stabilization, the FEM is not considered an ideal candidate for the spatial discretization of the Navier–Stokes equations.

We highlight one particular feature of the Galerkin method, which distinguishes this method from the FD and FV methods: As a user, we choose the shape functions and the Galerkin method automatically tries to find an optimal solution in a least-squares sense. Out of common practice, polynomial shape functions are usually chosen. However, most physical phenomena show solutions of nonpolynomial shape. Therefore, the Galerkin method enables vastly efficient spatial discretizations by employing problem-tailored shape functions using an *a priori* known approximate solution as an enrichment function. This idea is used in Part II of this thesis to develop a novel approach to wall modeling in CFD. Therein, the continuous FEM approach can be used as a baseline method (see Chapter 7).

2.2.2.4. The Discontinuous Galerkin Method

So far, none of the presented methods has shown fully satisfactory characteristics. Table 2.1 gives an overview of the methods discussed and it is noted that each method shows a different drawback. The question is therefore: can a method be constructed that combines the unstructured mesh and high-order capabilities of the FEM with the efficiency of the FV method in explicit schemes including convective stabilization? The answer is the DG method. This method is derived from the partial differential equations by first subdividing the computational domain into finite cells and subsequently deriving variational formulations for each cell. In this manner, the geometric flexibility is maintained while the solution is represented by a (possibly high-order) polynomial in each cell as in the FEM. Between elements, the numerical fluxes may be used to stabilize the method in the convective regime analogous to the FV method. Since the elements are decoupled, the mass matrix becomes block-diagonal such that a local mass matrix can be inverted in each cell. Furthermore, the DG method with a polynomial degree of zero yields for pure advection problems exactly the same equations as the second-order FV method. These characteristics satisfy all the basic requirements for the development of an efficient numerical method for turbulent flow, as noted in the introduction.

Therefore, a high-order DG solver is used and further developed in this study. This solver is described in Chapter 3 for laminar and eddy-resolving simulations and extended in Chapter 5 to ensemble-averaged simulations of turbulent flow. As the DG method exhibits the same flexibility in terms of choosing shape functions outside of the traditional polynomial space, we develop wall modeling approaches via function enrichment based on the DG solver in Chapters 8 to 10.

2.2.3. Suitability of Standard Schemes for Simulating Multiscale Turbulence

All of the presented numerical methods are ideally-suited for simulating single-scale phenomena such as wave propagation with a single wave number. In such simulations, the discretization errors may be quantified and the resolution can be adjusted to match the required accuracy. Due to the multiscale phenomena present in turbulent flows, some scales are inevitably resolved very well while others are computed with marginal resolution, and most of the turbulence is not even resolved in industrial simulations. As a result, there is a necessity for modeling, and several methodologies are summarized in the subsequent section.

2.3. Fundamental Approaches to the Numerical Simulation of Wall-Bounded Turbulence

There is no such CFD method that fulfills all requirements of engineers and researchers. Rather, different applications pose different requirements, with fidelity and computational cost as the basic parameters. In the following, four methodologies for simulating wall-bounded turbulent flows are outlined. In this work, we develop computational tools in all of these fields and perform computations using all four methodologies, with the aim of increasing the ratio of accuracy over speed for engineering-type simulations.

2.3.1. Direct Numerical Simulation

DNS is the simplest and yet most challenging approach to computing turbulent flows and provides the highest level of accuracy. All scales in the flow are resolved, including the viscous dissipation scales, such that modeling is not needed. In Figure 2.2, the computed part of the energy spectrum is labeled. In order to guarantee that sufficient physical dissipation is captured by the scheme, the resolution of a DNS has to be evaluated carefully. According to Pope [206], most of the dissipation occurs at a length scale of approximately 24 Kolmogorov lengths η and Moin and Mahesh [184] state that most of the dissipation is resolved above 15η. Depending on the resolution power of the scheme, the distance between two grid points is usually around 1 to 4η [184]. The resolution capacity of the spectral DG solver of this work is discussed in more detail in Chapter 4.

The sole limitation of DNS is its computational cost. The scaling of the resolution requirements with Kolmogorov units yields a dependence of the required number of grid points with the Reynolds number as $\mathrm{Re}^{2.25}$ according to [40], and as $\mathrm{Re}^{2.65}$ according to [45]. Additional requirements are posed on the time step size in order to resolve the relevant temporal scales. Therefore, DNS is out of reach in the foreseeable future regarding most industrial applications in the moderate to large Reynolds number regime. DNS is still a valuable tool in research and industry, e.g., for computing reference data and investigating selected low-Reynolds-number configurations, see for example [184] for an overview.

In Chapter 4 of this thesis, we perform DNS of two popular reference benchmarks with the high-order DG solver presented: turbulent channel flow and flow over periodic hills. The latter simulations exhibit the highest fidelity to date and provide a new set of reference data for two Reynolds numbers.

2.3.2. Large-Eddy Simulation

The high cost of resolving all turbulent scales suggests a method where only the large, inhomogeneous, energy-containing scales are resolved and the smaller homogeneous scales are modeled. This is the methodology of LES and the respective part of the energy spectrum is labeled in Figure 2.2. Governing equations for LES may be derived by several approaches, for example by the classical filtering [175] or a variational projection onto a given spatial discretization in the context of Galerkin methods [120]. In the case of filtering using a filtering operator $\overline{(\cdot)}$, the incompressible Navier–Stokes equations become under the assumption of commutativity of the gradient and filtering operators [223]:

$$\frac{\partial \bar{\boldsymbol{u}}}{\partial t} + \nabla \cdot \left(\boldsymbol{\mathcal{F}}^{c}(\bar{\boldsymbol{u}}) + \bar{p}\boldsymbol{\mathcal{I}} - \boldsymbol{\mathcal{F}}^{\nu}(\bar{\boldsymbol{u}}) \right) = \boldsymbol{f} - \nabla \cdot \boldsymbol{\tau}_{\mathrm{LES}}, \tag{2.10}$$

$$\nabla \cdot \bar{\boldsymbol{u}} = 0, \tag{2.11}$$

with the LES subgrid tensor $\boldsymbol{\tau}_{\mathrm{LES}} = \overline{\boldsymbol{u} \otimes \boldsymbol{u}} - \bar{\boldsymbol{u}} \otimes \bar{\boldsymbol{u}}$, which has to be modeled.

Sagaut [222] classifies the subgrid models in two groups, functional models on the one hand and structural models on the other hand. Functional models mimic the physical dissipation of the unresolved subgrid scales whereas structural models recover some of the nonlinear interactions between small resolved and subgrid scales. In Chapter 7, a structural LES model will be employed in conjunction with a variational multiscale approach within the continuous FEM. In

Chapters 5 and 9, a functional model is used in the context of DES (see subsequent section). A third approach is the idea of implicit subgrid models, stating that the numerical dissipation required to stabilize the scheme may in fact be appropriate to model the turbulent motions on subgrid level, or at least their physical dissipation. This modeling approach is popular in particular in the context of the DG method, see the discussion and literature review in the Chapters 3 and 4.

LES is recognized as a method, which is well capable of reproducing results of DNS with much coarser grids and its particular strengths lie in the accurate prediction of separated flows. One key advantage of LES in comparison to the subsequent modeling approaches is that grid refinement always yields convergence to DNS.

The primary limitation of LES arises from what is displayed in Figure 2.5: The size of the energy-containing scales is proportional to the distance to the wall and, near the wall, the largest turbulent motions are of a similar size as the viscous dissipation scales. As a result, very fine boundary layer meshes have to be used, and the near-wall spatial and temporal resolution requirements increase dramatically with the Reynolds number. Chapman [42] estimates the scaling of the grid resolution requirements with the Reynolds number as $\mathrm{Re}^{1.8}$, which is only moderately relaxed compared to the DNS requirement, and Baggett et al. [12] derived a scaling in wall units of Re_τ^2. The difference in the exponent between Re and Re_τ arises from the scaling of the skin friction with $\mathrm{Re}^{-0.2}$ [42]. Small computation cells in the boundary layer come along with severe constraints on the time step size to resolve the temporal scales of momentum-transfer mechanisms and to be compliant with time step restrictions if explicit time integration schemes are employed. These computational requirements are out of reach in many industrial applications and there is a need for the further reduction of the computational cost through more invasive modeling. These are the topics of the two subsequent sections: statistical modeling and wall modeling.

2.3.3. Reynolds-Averaged Navier–Stokes

In the past, the available computer resources were by far not sufficient for LES of industrial flows. Even today, when thousands of parameter settings have to be analyzed during a product development cycle, a primary requirement is short run time. Under these circumstances, the method of choice would be the RANS approach. The governing equations are derived by considering a Reynolds decomposition of the velocity and pressure into a statistical ensemble-averaged component, denoted by $\langle\cdot\rangle$, and a fluctuating component, denoted by $(\cdot)'$, yielding

$$\boldsymbol{u} = \langle \boldsymbol{u} \rangle + \boldsymbol{u}'. \tag{2.12}$$

Substituting this decomposition in Equations (2.1) and (2.2) and assuming permutability of gradient and averaging operators [223] results in the RANS equations:

$$\frac{\partial \langle \boldsymbol{u} \rangle}{\partial t} + \nabla \cdot \big(\boldsymbol{\mathcal{F}}^c(\langle \boldsymbol{u} \rangle) + \langle p \rangle \boldsymbol{\mathcal{I}} - \boldsymbol{\mathcal{F}}^\nu(\langle \boldsymbol{u} \rangle) \big) = \boldsymbol{f} - \nabla \cdot \boldsymbol{\tau}_{\mathrm{RANS}}, \tag{2.13}$$

with the Reynolds stress tensor $\boldsymbol{\tau}_{\mathrm{RANS}} = \langle \boldsymbol{u}' \otimes \boldsymbol{u}' \rangle$, which has to be modeled. The continuity equation remains unchanged:

$$\nabla \cdot \langle \boldsymbol{u} \rangle = 0. \tag{2.14}$$

The vast majority of RANS turbulence models considers an eddy viscosity approach, taking into account the Reynolds stress tensor by the Boussinesq approximation,

$$-\langle \boldsymbol{u}' \otimes \boldsymbol{u}' \rangle \approx 2\nu_t \boldsymbol{\epsilon}(\langle \boldsymbol{u} \rangle) = \boldsymbol{\mathcal{F}}^{\nu_t}(\langle \boldsymbol{u} \rangle), \tag{2.15}$$

with the eddy viscosity ν_t. Historically the first and probably the simplest turbulence model is Prandtl's mixing length model, which is applicable in the log-layer of a boundary layer:

$$\nu_t = (\kappa y)^2 |\boldsymbol{\epsilon}(\langle \boldsymbol{u} \rangle)|, \tag{2.16}$$

where $|\boldsymbol{\epsilon}(\langle \boldsymbol{u} \rangle)| = \sqrt{2\boldsymbol{\epsilon}(\langle \boldsymbol{u} \rangle) : \boldsymbol{\epsilon}(\langle \boldsymbol{u} \rangle)}$. There is a large number of advanced one and two-equation models, and their review is beyond the scope of this introduction; see, e.g., [266] for an overview. In Chapters 5 and 8, we consider the SA one-equation turbulence model [242] as an extension of the DG scheme.

RANS models depend heavily on calibrated model constants and their range of applicability is limited. They are generally well-suited for flows with attached boundary layers with moderate pressure gradients. In massively separated flows or flow separation from curved boundaries, RANS models may show deficiencies (see, e.g., [125]). For example, the SA model fails to accurately predict the flow in the recirculation zone of the periodic hill flow (see Section 5.4.2). Also, due to the statistical modeling approach, RANS simulations do not converge to DNS data by means of mesh refinement. Finally, resolving the sharp velocity gradient at the wall with the numerical scheme may be quite costly as well, and requires additional wall modeling at high Reynolds numbers, see for example the wall model developed in Chapter 8.

2.3.4. DES, Wall Modeling, and Hybrid RANS/LES

Due to the high cost of LES in the boundary layer, the concept of wall modeling was introduced in early works on LES by Deardorff [60] and Schumann [231] in an attempt to circumvent the resolution dependence on wall units. Wall modeling implies that near-wall turbulence and the accompanying momentum transfer are not resolved but modeled in a statistical sense. Despite this statistical modeling, wall-modeled LES often exhibits higher accuracy than RANS as the anisotropic turbulent motions of the bulk flow are resolved. With the near-wall turbulence modeled, the size of dominating eddies in the bulk of the flow are governed by geometrical scales of boundary conditions with resolution requirements increasing approximately as $\mathrm{Re}^{0.4}$ [204]. Such a Reynolds number dependence is acceptable in most industrial applications. As a comprehensive review of the literature is beyond the scope of this introduction, we refer to the reviews in [94, 168, 203, 204, 223, 268], and the discussion is limited to the main ideas.

There are two principal approaches to wall modeling: hybrid RANS/LES including DES and wall-stress models.

- Hybrid RANS/LES or detached-eddy simulation (DES) approaches consider a RANS closure in the inner boundary layer, whereas an LES subgrid model is used in the outer layer or the bulk flow. Both methods are applied on the same computational grid and employ the same baseline solver. The traditional detached-eddy simulation approach according to Spalart et al. [244] blends the turbulence models by limiting the wall distance parameter in the SA model (see Section 5.2) by a characteristic grid length scale such that the standard

RANS equations are solved in the boundary layer and the RANS model degenerates to a one-equation LES closure elsewhere. As the boundary layer turbulence is not resolved, the wall-parallel grid spacing can be very large, which achieves a drastic reduction in the grid point requirements. Regarding the application of DES, two main branches are frequently used. If the wall-parallel grid length is chosen of the order of up to 0.1δ, with the boundary layer thickness δ, the SA model acts as a RANS model in the inner boundary layer region only and as an LES subgrid model in the outer layer [195, 204] (see Section 2.1.3), yielding WMLES. As an alternative, the whole boundary layer may be computed in RANS mode, where the wall-parallel grid spacings are of the order of δ such that only the turbulence in 'detached' shear layers is resolved [168, 244]. WMLES is by a factor of 10 to 100 more expensive than classical DES, but still yields substantial savings [168]. Numerous other hybrid RANS/LES approaches have been proposed, for example the method of explicitly blending RANS and LES eddy viscosity models by Baggett [11] or the theoretical framework for blending hybrid RANS/LES filters instead of models by Germano [97]. Applications within DG may be found for example in [179, 267]. We consider DES as WMLES in Section 5.5 and extend the approach by a new wall modeling concept in Chapters 9 and 10.

- Wall-stress models are a second category of wall models. The aim of this approach is to model the inner layer, including both the turbulent motions and the velocity profile in one method. Therein, the no-slip boundary conditions are replaced by traction boundary conditions, i.e., Neumann boundary conditions, such that the velocity gradient at the wall does not have to be resolved by the scheme. In turn, the wall shear stress is computed based on the velocity of a grid point, which usually lies in the logarithmic region. There is a large number of ways for predicting the wall stress in turbulent boundary layers and applying the stresses as boundary conditions, see for example the wall models developed within the high-order DG method [88, 90, 263]. The simplest approach is to employ the relation of the log-law, which poses the requirement that the first off-wall node has to be located inside the log-layer. In addition, the underlying assumptions in such models are significant and they may only be applied in equilibrium boundary layers [203]. These restrictions can be reduced by more advanced wall functions. Another and better consistent approach is the so-called two-layer model (TLM) (see, e.g., the review in [204]), in which simplified thin-boundary layer equations (TBLE) are solved on a separate grid in the inner layer.

We discuss further advantages and disadvantages of these wall modeling approaches in Chapter 6 prior to introducing a novel approach to wall modeling. Wall modeling may also be necessary in RANS simulations of high-Reynolds number, as the resolution of the near-wall velocity gradient would be quite expensive. Further details on wall modeling for RANS are given in Chapters 6 and 8.

2.4. Summary

In this chapter, the chaotic and eddy-dominated nature of turbulence was identified as a challenge for the simulation of turbulent flows. The most accurate simulation approach to turbulence is DNS, where all scales are resolved. As the Reynolds number increases, ratio of the large over small scales grows, making a direct computation of all scales impossible. One possibility is to simulate the large and inhomogeneous eddies only and to model the smaller scales, which yields a similar level of accuracy while reducing the cost. However, this approach is still too computationally demanding for many applications, as the large near-wall eddies are indeed small. A statistical model of the whole flow is possible, however with the drawback of a reduced fidelity of the prediction. A reasonable compromise can be seen in the concept of wall modeling in conjunction with LES, where the large eddies are only computed away from the walls, and the near-wall turbulence is modeled. In this thesis, we develop computational methods for all of these simulation approaches with a strong focus on the latter wall modeling concept. These models are developed within a high-order DG method, which will be introduced in the subsequent chapter.

I

Wall-Resolved Computational Fluid Dynamics: High-Order DG Methods and Applications

3

A High-Order Incompressible Semi-Explicit Discontinuous Galerkin Solver

The description of the numerical methods for the solution of the Navier–Stokes equations in Section 2.2 identified the DG method as an approach that combines four highly desirable characteristics in one computational approach. These characteristics are, according to Section 2.2: stability in the convection-dominated regime with low dissipation properties, high-order capability in conjunction with unstructured meshes, and efficiency in explicit time stepping. In this chapter, a solver for incompressible flows is presented and extended, that makes use of these qualities and is applicable to laminar and turbulent incompressible flow, however with a strong focus on spatially underresolved simulations of turbulence (LES). Instabilities occurring in the limit of small time steps and in marginally resolved flows are analyzed in detail and remedies to these issues are developed.

This solver is applied to DNS and LES of turbulent flow in Chapter 4. An extension to the RANS equations is developed in Chapter 5 and the concept of wall modeling via function enrichment is also implemented in this solver, which is described in Part II of this thesis. In the first section of the following chapter, we give an overview of available DG schemes in the literature. Subsequently, the temporal discretization of a splitting scheme is presented in Section 3.2. Section 3.3 introduces the spatial DG discretization and Section 3.4 compares several modifications of the scheme. Section 3.5 gives an overview of the implementation and Section 3.6 presents laminar verification examples. The present chapter is based on joint work with a second PhD student Niklas Fehn under the supervision of Dr. Martin Kronbichler at the Institute for Computational Mechanics and has been published in Krank et al. [147].

3.1. A Review of High-Order DG Methods for Incompressible Flow and Motivation

3.1.1. Use Compressible or Incompressible Schemes?

High-order DG schemes governed by the compressible Navier–Stokes equations have been developed for many years and can be applied to a wide range of compressible turbulent flows and simulation approaches, e.g., DNS (see Section 2.3.1) [15, 115], LES (see Section 2.3.2) [15, 19, 262], and RANS simulations (see Section 2.3.3) [17, 167, 259]. Applications range from internal turbomachinery flows [264], the side-view mirror of a car [87], a high-lift configuration of an entire aircraft [110], to environmental flows [98, 99].

As fully incompressible DG schemes of an equally mature development level have not been widely available, it is common practice to compute incompressible flows with compressible codes at small Mach number to avoid compressibility effects; see, e.g., [15, 19, 28, 52, 263]. However, the time step restrictions in such schemes can be severe if an explicit approach is used [263], since the time step has to be chosen based on the speed of sound as the convective velocity instead of the fluid velocity. The analysis in [263] concludes that a Mach number of $\mathrm{Ma} = 0.1$ is sufficient to avoid compressibility effects, so an explicit compressible scheme would require approximately ten times more time steps in comparison to an explicit incompressible method. However, the study in [145] compares the results of a compressible and an incompressible approach and their results show noticeable differences even at $\mathrm{Ma} = 0.05$ in the compressible method, so further analysis is required in order to assess the suitability of compressible schemes for incompressible flows. In addition to the time step restriction, the compressible Navier–Stokes equations require the supplementary solution of the energy equation and exhibit a higher degree of nonlinearity, which either results in more expensive terms [140] or requires a higher level of modeling in form of dealiasing [112].

The bottleneck in incompressible schemes is the pressure variable, since this quantity cannot be formulated explicitly unless the partial differential equations are modified, such as in artificial compressibility methods; see, e.g., [181] (LES) and [196] (URANS) for an application to turbulent flow. As a result, the pressure in truly incompressible methods requires the solution of a challenging equation system, which can be quite expensive as well. As a comprehensive comparison of two highly developed compressible and incompressible schemes is not readily available, it is currently impossible to answer the question whether compressible or incompressible high-order DG methods are better suited for incompressible flows. However, we present techniques for incompressible solvers that allow such a comparison in the future.

3.1.2. DG Schemes for Incompressible Flow

Incompressible numerical schemes in the context of eddy-resolving simulations are rare and have for example been employed by [73, 74] within a 2D DG solver coupled with a spectral vanishing viscosity approach in the third space dimension and by [248] using a space-time method. Fully resolved unsteady laminar (DNS-like) simulations and have been carried out by [54] using an artificial compressibility flux method within an incompressible scheme [16]. The latter numerical method was recently applied to turbulent channel flow as well [86]. The related hybridizable discontinuous Galerkin (HDG) method has shown promising results in [193] for simple setups. Furthermore, [174] used HDG for the convective and DG for the viscous term within an operator-splitting technique. In the context of the state-of-the-art matrix-free algorithms used in the present thesis, the symmetric interior penalty (SIPG) DG flavor is much more efficient in comparison to HDG, especially in three space dimensions [162].

An efficient time integration scheme is the key to large-scale simulations of the incompressible Navier–Stokes equations aiming at turbulent flows. Coupled solvers applied within DG for example in [48, 49, 141, 218, 229] require the solution of a saddle point problem and include nonlinear iterations within each time step; they have so far only been applied to small-scale academic examples. On the contrary, temporal splitting schemes allow for tailored solution procedures regarding the respective terms contained in the Navier–Stokes equations, which renders them much more efficient in many applications. In particular, explicit time integration steps can be performed very efficiently in DG (see Section 2.2.1.1). There are four main branches of splitting methods, namely pressure-correction, velocity-correction, algebraic-splitting and consistent-splitting schemes, see [105] for an overview. With respect to DG, a pressure-correction method with discontinuous velocity and continuous pressure has for example been proposed in [30]. Pressure-correction schemes are however limited to second-order accuracy in time [105]. An algebraic splitting scheme has for example been proposed by Shahbazi et al. [234].

In this thesis, we consider the high-order velocity-correction (also termed dual-splitting) scheme by Karniadakis et al. [135], which has been applied within DG in a series of recent publications [66, 67, 72–75, 77, 112, 147, 202, 246] and within the related spectral multidomain penalty method (SMPM) in [68, 129]. The scheme allows equal-order interpolations for velocity and pressure [136] and splits each time step into three substeps: The nonlinearity present in the convective term is first handled explicitly, a Poisson problem is subsequently solved for the pressure which is used to make the velocity divergence-free and the viscous term is taken into account in the third step.

3.1.3. Instabilities Reported in the Literature

A downside of many splitting schemes are limitations coming along with the splitting approach [105]. As a novel contribution of this thesis, remedies are provided to two such limitations observed with the dual-splitting scheme when employed in conjunction with the DG method as reported in a series of recent papers, e.g., [72, 73, 75, 129, 147, 202, 246]. These issues are not related to the well-understood aliasing errors induced by underintegration of nonlinear terms [112] or convection-dominated flow regimes. The two sources of instabilities are:

- Ferrer and Willden [73] and Ferrer et al. [75] discuss instabilities encountered for small time steps with this scheme, both for continuous and discontinuous Galerkin discretizations. In the remainder of this chapter, it is shown that these instabilities arise due to spurious divergence errors as a consequence of the finite spatial resolution. Several remedies to this issue are reviewed, among them a consistent div-div penalty term within the local projection step reducing the point-wise divergence error. This term may be seen as a much simpler variant of the postprocessing proposed in [246] and is similar to the grad-div term frequently used in continuous Galerkin [197].

- Violation of the mass balance through velocity discontinuities across element boundaries triggers another instability recently described by Joshi et al. [129]. This instability becomes especially relevant in spatially underresolved simulations such as LES of turbulent flow. We review and benchmark two remedies to this problem, which are a supplementary jump-penalty term included in the projection according to [129] on the one hand or partial integration of the right-hand side of the Poisson equation on the other hand.

With regard to this time integration scheme, a third issue has been discussed in the literature. The dual-splitting scheme exhibits an inf-sup instability if equal orders for velocity and pressure are used when very small time steps are employed [72, 75]. As we aim at time step sizes close the maximum allowable ones in this work, this issue is not directly relevant for the simulations presented in this thesis.

Based on these enhancements, we develop an efficient high-order DG solver applicable to laminar and turbulent incompressible flow including implicit LES. The spatial DG discretization

Table 3.1.: Time integration parameters for time stepping with constant time step size [135]. The parameters are extended in Section 3.3.3 to adaptive time stepping.

J, J_p	α_0	α_1	α_2	β_0	β_1	β_2	γ_0
1	1	0	0	1	0	0	1
2	2	$-1/2$	0	2	-1	0	$3/2$
3	3	$-3/2$	$1/3$	3	-3	1	$11/6$

follows the approaches proposed in Hesthaven and Warburton [112] and Shahbazi et al. [234] (although the latter publication uses triangular elements and an algebraic splitting scheme), and a preliminary version of the solver used in this work was developed by Fehn [71]. We choose the local Lax–Friedrichs numerical flux for the discretization of the convective term and the interior penalty method [10] for the Poisson problem and the viscous term. The local conservativity of the overall method is attained by use of the divergence form of the incompressible Navier–Stokes equations [234]. The method further employs nodal Lagrangian shape functions. As it is described in [147], multigrid solvers with smoothing based on the point Jacobi method require fewer iterations with nodal polynomials than with modal shape functions, which are used, e.g., in [73, 75, 246]. This is since the underlying matrices are closer to diagonal dominance in the case of nodal shape functions. In addition, face integrals involving shape values of nodal polynomials, where only some shape functions are nonzero on a particular face, are cheaper to evaluate than bases that must be interpolated from all shape functions onto the boundary [147]. The method is implemented based on matrix-free operator evaluation that relies on sum factorization [146, 156–158] in a solver called INDEXA (INcompressible Discontinuous Galerkin towards the EXA scale).

3.2. Temporal Velocity-Correction Scheme

The incompressible Navier–Stokes equations (2.1) and (2.2) are discretized in time by use of the semi-explicit dual-splitting scheme by Karniadakis et al. [135]. Herein, the transient term is discretized by a backward-differencing formula (BDF), the nonlinear convective term is treated with an extrapolation scheme (EX) and the pressure, viscous and body force terms are handled implicitly. The time-discretized momentum equation (2.1) becomes

$$\frac{\gamma_0 \boldsymbol{u}^{n+1} - \sum_{i=0}^{J-1} \left(\alpha_i \boldsymbol{u}^{n-i}\right)}{\Delta t} + \sum_{i=0}^{J-1} \beta_i \nabla \cdot \boldsymbol{\mathcal{F}}^c(\boldsymbol{u}^{n-i}) + \nabla p^{n+1} - \nabla \cdot \boldsymbol{\mathcal{F}}^\nu(\boldsymbol{u}^{n+1}) = \boldsymbol{f}^{n+1}, \quad (3.1)$$

with the solution $\boldsymbol{u}^{n+1}$ and p^{n+1} at time level $t^{n+1} = (n+1)\Delta t$ with n indicating the time step and Δt the increment in time. The time integrator constants γ_0, α_i, and β_i of the BDF and EX schemes are given in [135] and they are listed in Table 3.1. We consider the temporal orders of accuracy $J = \{1, 2, 3\}$. In Section 3.3.3, we give details on how these parameters may be adaptively computed in order to take into account temporally varying time step increments. As the scheme is not self-starting for $J = \{2, 3\}$, the first time steps are performed either by successively increasing the BDF order or by interpolation of the solution to the discrete time instants $t^{n-1}, ..., t^{n-J+1}$ if an analytical solution is available.

In the framework of DG methods, the dual-splitting scheme by Karniadakis et al. [135] has for example been investigated by Hesthaven and Warburton [112], Ferrer and Willden [76], Steinmoeller et al. [246], Ferrer et al. [75], as well as Emamy [66]. It splits Equation (3.1) into three substeps: (i.) the nonlinear convective term is advanced in time explicitly, (ii.) the pressure is computed by solving a pressure Poisson equation and the result is used to project the velocity onto a solenoidal space, and (iii.) the viscous term is handled implicitly. The substeps are:

i. *Explicit convective step*

In the first substep, the nonlinear convective term is handled efficiently by explicit time advancement,

$$\frac{\gamma_0 \hat{\boldsymbol{u}} - \sum_{i=0}^{J-1} \left(\alpha_i \boldsymbol{u}^{n-i}\right)}{\Delta t} = -\sum_{i=0}^{J-1} \beta_i \nabla \cdot \boldsymbol{\mathcal{F}}^c(\boldsymbol{u}^{n-i}) + \boldsymbol{f}^{n+1}, \tag{3.2}$$

yielding the first intermediate velocity $\hat{\boldsymbol{u}}$.

ii. *Pressure Poisson equation and projection*

The pressure step consists of solving a Poisson equation for the pressure at time t^{n+1} given as

$$-\nabla^2 p^{n+1} = -\frac{\gamma_0}{\Delta t} \nabla \cdot \hat{\boldsymbol{u}}. \tag{3.3}$$

Consistent boundary conditions for this problem on $\partial\Omega^D$ may be derived according to [135, 199] by multiplication of the momentum equation (2.1) with the normal vector. The resulting transient term is treated using the Dirichlet values given, likewise the body force, while the convective and viscous contributions are handled explicitly to avoid a dependency on the velocity solution, yielding

$$\begin{aligned} &\nabla p^{n+1} \cdot \boldsymbol{n} = \\ &-\left(\frac{\partial \boldsymbol{g}_{\boldsymbol{u}}(t^{n+1})}{\partial t} + \sum_{i=0}^{J_p-1} \beta_i \big(\nabla \cdot \boldsymbol{\mathcal{F}}^c(\boldsymbol{u}_h^{n-i}) + \nu \nabla \times \boldsymbol{\omega}^{n-i}\big) - \boldsymbol{f}^{n+1}\right) \cdot \boldsymbol{n} \qquad \text{on } \partial\Omega^D. \end{aligned} \tag{3.4}$$

Herein, $\boldsymbol{\omega}$ denotes the vorticity and J_p the extrapolation order of the convective and viscous terms in the Neumann pressure boundary condition. Throughout this work, we take $J_p = J$ in order to obtain optimal temporal convergence rates both in velocity and pressure. On the contrary, the mixed-order case $J_p = J - 1$ discussed for example in [106, 177] yields sub-optimal convergence rates for the pressure [106, 177], see also the discussion in [71]. Note that only the solenoidal part in form of the rotational formulation of the viscous term is accounted for, which has been reported to be essential for reducing boundary divergence errors as well as high-order temporal accuracy of the overall methodology [135, 199]. A suitable boundary condition on $\partial\Omega^N$ for Neumann outflow is $p^{n+1} = g_p(t^{n+1})$, prescribing the desired pressure value $g_p(t^{n+1})$ directly.

Utilizing the new pressure field p^{n+1}, the first intermediate velocity $\hat{\boldsymbol{u}}$ is projected onto the space of divergence-free vectors by

$$\hat{\hat{\boldsymbol{u}}} = \hat{\boldsymbol{u}} - \frac{\Delta t}{\gamma_0} \nabla p^{n+1}, \tag{3.5}$$

resulting in the second intermediate velocity $\hat{\hat{\boldsymbol{u}}}$.

iii. *Implicit viscous step*

The final solution $\boldsymbol{u}^{n+1}$ at time t^{n+1} is computed implicitly due to stability considerations by a Helmholtz-like equation reading

$$\frac{\gamma_0}{\Delta t}(\boldsymbol{u}^{n+1} - \hat{\hat{\boldsymbol{u}}}) = \nabla \cdot \boldsymbol{\mathcal{F}}^{\nu}(\boldsymbol{u}^{n+1}). \tag{3.6}$$

The system is closed by specifying boundary conditions for the velocity according to

$$\begin{aligned} &\boldsymbol{u}^{n+1} = \boldsymbol{g}_{\boldsymbol{u}}(t^{n+1}) && \text{on } \partial\Omega^{D} \text{ and} \\ &\boldsymbol{\mathcal{F}}^{\nu}(\boldsymbol{u}^{n+1}) \cdot \boldsymbol{n} = \boldsymbol{h}(t^{n+1}) + g_p(t^{n+1})\boldsymbol{n} && \text{on } \partial\Omega^{N}. \end{aligned} \tag{3.7}$$

3.3. Spatial Discretization

Let us commence the discussion on the spatial discretization in the first Subsection 3.3.1 by introducing the notation used. Subsequently, the variational formulation is presented in Subsection 3.3.2. Several alternative variants of the resulting weak form are discussed and compared with regard to stability in marginally resolved simulations as well as for small time steps in the subsequent Section 3.4.

3.3.1. Preliminaries

We consider a tessellation of the d-dimensional domain $\Omega \subset \mathbb{R}^d$ into N_e nonoverlapping quadrilateral/hexahedral finite elements $\Omega_h = \bigcup_{e=1}^{N_e} \Omega_e$. The exterior boundaries of Ω_h are denoted by $\partial\Omega_h$. They are partitioned into a Dirichlet and Neumann boundary $\partial\Omega_h = \partial\Omega_h^D \cup \partial\Omega_h^N$ with $\partial\Omega_h^D \cap \partial\Omega_h^N = \emptyset$. Interior boundaries $\partial\Omega_e^- \cap \partial\Omega_e^+$ between two adjacent elements Ω_e^- and Ω_e^+ are named $\partial\Omega_e^\Gamma$. The unit normal vectors of such interior boundaries are oriented outwards of the respective element yielding $\boldsymbol{n}_\Gamma^- = -\boldsymbol{n}_\Gamma^+$, i.e., $\boldsymbol{n}_\Gamma^-$ is oriented from Ω_e^- to Ω_e^+ as well as outwards with respect to Ω_e^- on exterior boundaries, accordingly. In the element-wise notation of the weak form presented in the following subsections, we refer to the current element by the superscript $(\cdot)^-$ and to the neighboring element by $(\cdot)^+$, i.e., $\boldsymbol{n}_\Gamma = \boldsymbol{n}_\Gamma^-$.

The discontinuity of the primary variables across element interfaces may be expressed in terms of jump operators $[\cdot]$ and $[\![\cdot]\!]$, which are defined as $[\phi] = \phi^- - \phi^+$ and $[\![\phi]\!] = \phi^- \otimes \boldsymbol{n}_\Gamma^- + \phi^+ \otimes \boldsymbol{n}_\Gamma^+$, respectively, where the latter is given for the multiplication operator $\otimes$ applicable to scalars, vectors as well as tensors and which increases the tensor rank by one. Similarly, an averaging operator is defined as $\{\{\phi\}\} = w^-\phi^- + w^+\phi^+$ with the weights $w^- = w^+ = 1/2$ if not specified otherwise. This averaging operator will be used in Chapters 5 and 10 to consider spatially dependent material parameters in the viscous term. We also use extensions of these operator definitions to the boundaries, which are $\{\{\phi\}\}^{ND} = \{\{\phi\}\}$ and $[\phi]^{ND} = [\phi]$ on $\partial\Omega_e^\Gamma$ and $\{\{\phi\}\}^{ND} = \phi^-$ as well as $[\phi]^{ND} = 0$ on $\partial\Omega_e^N \cup \partial\Omega_e^D$.

Dirichlet boundary conditions are enforced using the mirror principle according to [71, 112] for all terms, defining $\phi^+ = -\phi^- + 2g$ and $\nabla\phi^+ = \nabla\phi^-$ on $\partial\Omega_e^D$. Neumann boundary conditions are applied by setting the external value to $\phi^+ = \phi^-$ and prescribing a gradient as $\nabla\phi^+ \cdot \boldsymbol{n} = -\nabla\phi^- \cdot \boldsymbol{n} + 2h_\phi$ on $\partial\Omega_e^N$, where h_ϕ is the Neumann boundary condition [71, 112].

Further notation used in the weak formulations below includes L^2-inner products abbreviated with $(a,b)_{\Omega_e} = \int_{\Omega_e} ab \, \mathrm{d}\Omega$ for scalars, $(\boldsymbol{a},\boldsymbol{b})_{\Omega_e} = \int_{\Omega_e} \boldsymbol{a}\cdot\boldsymbol{b} \, \mathrm{d}\Omega$ for vectors and $(\boldsymbol{a},\boldsymbol{b})_{\Omega_e} = \int_{\Omega_e} \boldsymbol{a} : \boldsymbol{b} \, \mathrm{d}\Omega$ for tensors of rank 2 (the double dot product is evaluated as $\boldsymbol{a} : \boldsymbol{b} = a_{ij}b_{ij}$ in index notation). Boundary integrals are defined accordingly.

Approximate solutions are to be found within spaces of the form

$$\mathcal{V}_h^p = \{p_h \in L^2 : p_h|_{\Omega_e} \in P_k(\Omega_e), \forall e \in \Omega_h\}$$

for the pressure and the equivalent vector-valued version $\mathcal{V}_h^{\boldsymbol{u}} = (\mathcal{V}_h^p)^d$ for the velocity. Herein, $P_k(\Omega_e)$ denotes the space of polynomials of tensor degree up to k where the polynomial order is restricted by $k \geq 1$ while the high-order pressure boundary conditions may first be represented sufficiently well with $k \geq 2$. The polynomials are given by a tensor product of one-dimensional Lagrange polynomials based on Legendre–Gauss–Lobatto nodes for good conditioning at arbitrary polynomial degrees [136]. Further details on aspects related to efficient implementation of these elements are discussed in Section 3.5.

3.3.2. Variational Formulation

The variational formulation is derived for each substep of the time integration scheme by multiplying Equations (3.2) to (3.7) with appropriate weighting functions $\boldsymbol{v}_h \in \mathcal{V}_h^{\boldsymbol{u}}$ and $q_h \in \mathcal{V}_h^p$, respectively, and integrating over one element volume Ω_e^-. If possible without ambiguity, the superscript $(\cdot)^-$ is dropped in the following for simplicity. The variational forms are based on the work by Fehn [71] and include the modifications suggested by Krank et al. [147] regarding the viscous term and the pressure boundary condition.

i. *Explicit Convective Step*

The flux formulation of the convective step is derived by integration by parts and subsequent application of the divergence theorem, yielding

$$\left(\boldsymbol{v}_h, \frac{\gamma_0 \hat{\boldsymbol{u}}_h - \sum_{i=0}^{J-1}\left(\alpha_i \boldsymbol{u}_h^{n-i}\right)}{\Delta t}\right)_{\Omega_e} = \sum_{i=0}^{J-1} \beta_i \Big(\big(\nabla \boldsymbol{v}_h, \boldsymbol{\mathcal{F}}^c(\boldsymbol{u}_h^{n-i})\big)_{\Omega_e} - \big(\boldsymbol{v}_h, \boldsymbol{\mathcal{F}}^{c*}(\boldsymbol{u}_h^{n-i}) \cdot \boldsymbol{n}_\Gamma\big)_{\partial\Omega_e} \Big) + \big(\boldsymbol{v}_h, \boldsymbol{f}_h^{n+1}\big)_{\Omega_e} \tag{3.8}$$

where $\hat{\boldsymbol{u}}_h \in \mathcal{V}_h^{\boldsymbol{u}}$. The local Lax–Friedrichs numerical flux is applied as it provides a stable formulation of the convective term. We have

$$\boldsymbol{\mathcal{F}}^{c*}(\boldsymbol{u}_h^{n-i}) = \begin{cases} \{\{\boldsymbol{\mathcal{F}}^c(\boldsymbol{u}_h^{n-i})\}\} + \Lambda/2[\![\boldsymbol{u}_h^{n-i}]\!] & \text{on } \partial\Omega_e^\Gamma, \\ \boldsymbol{\mathcal{F}}^c(\boldsymbol{u}_h^{n-i}) & \text{on } \partial\Omega_e^N, \\ 1/2\big(\boldsymbol{\mathcal{F}}^c(2\boldsymbol{g}_{\boldsymbol{u}}(t^{n-i}) - \boldsymbol{u}_h^{n-i}) + \boldsymbol{\mathcal{F}}^c(\boldsymbol{u}_h^{n-i})\big) & \\ \qquad\qquad +\Lambda\big(\boldsymbol{u}_h^{n-i} - \boldsymbol{g}_{\boldsymbol{u}}(t^{n-i})\big) \otimes \boldsymbol{n}_\Gamma & \text{on } \partial\Omega_e^D \end{cases} \tag{3.9}$$

using $\Lambda = \max(\lambda^-, \lambda^+)$ according to, e.g., [234] and the maximum eigenvalue of the respective flux Jacobian [147],

$$\begin{aligned} \lambda^- &= \max\nolimits_j \left| \lambda_j \left(\tfrac{\partial \boldsymbol{\mathcal{F}}(\boldsymbol{u}) \cdot \boldsymbol{n}_\Gamma}{\partial \boldsymbol{u}} \Big|_{\boldsymbol{u}_h^{-,n-i}} \right) \right| = 2|\boldsymbol{u}_h^{-,n-i} \cdot \boldsymbol{n}_\Gamma| \text{ and} \\ \lambda^+ &= \max\nolimits_j \left| \lambda_j \left(\tfrac{\partial \boldsymbol{\mathcal{F}}(\boldsymbol{u}) \cdot \boldsymbol{n}_\Gamma}{\partial \boldsymbol{u}} \Big|_{\boldsymbol{u}_h^{+,n-i}} \right) \right| = 2|\boldsymbol{u}_h^{+,n-i} \cdot \boldsymbol{n}_\Gamma|. \end{aligned} \tag{3.10}$$

The latter is defined as $\lambda^+ = 2|(2\boldsymbol{g}_{\boldsymbol{u}}(t^{n-i}) - \boldsymbol{u}_h^{-,n-i}) \cdot \boldsymbol{n}_\Gamma|$ on $\partial\Omega_e^D$. Note that Λ is evaluated on each quadrature point according to [71, 147], which differs from several other studies, where it is suggested to evaluate Λ based on mean values of $\boldsymbol{u}^+$ and $\boldsymbol{u}^-$ across the entire respective element (see, e.g., [142, 234]). Further, the factor 2 present in the eigenvalues (3.10) has been omitted in some publications; see, e.g., [112].

ii. *Pressure Poisson Equation and Projection*

For discretization of the pressure term, the SIPG method by Arnold [10] is considered. The weak form of the right-hand side of Equation (3.4), denoted by $a(q_h, \hat{\boldsymbol{u}}_h)$ in the following, is one of the primary objects of study in this chapter and will be discussed in detail in Section 3.4. In the simplest variant according to [112], we have

$$(\nabla q_h, \nabla p_h^{n+1})_{\Omega_e} - \frac{1}{2}(\nabla q_h, \boldsymbol{\Pi}^*)_{\partial\Omega_e} - (q_h, \boldsymbol{\mathcal{P}}^* \cdot \boldsymbol{n}_\Gamma)_{\partial\Omega_e} = \underbrace{-(q_h, \frac{\gamma_0}{\Delta t}\nabla \cdot \hat{\boldsymbol{u}}_h)_{\Omega_e}}_{=:a(q_h, \hat{\boldsymbol{u}}_h)}, \tag{3.11}$$

where the pressure jump function $\boldsymbol{\Pi}^*$ is given as

$$\boldsymbol{\Pi}^* = \begin{cases} [\![p_h^{n+1}]\!] & \text{on } \partial\Omega_e^\Gamma, \\ (2p_h^{n+1} - 2g_p(t^{n+1})) \otimes \boldsymbol{n}_\Gamma & \text{on } \partial\Omega_e^N \text{ and} \\ \boldsymbol{0} & \text{on } \partial\Omega_e^D. \end{cases} \tag{3.12}$$

The interior penalty flux $\boldsymbol{\mathcal{P}}^*$ includes a stabilization term according to

$$\boldsymbol{\mathcal{P}}^* = \begin{cases} \{\{\nabla p_h^{n+1}\}\} - \tau_{\text{IP}}[\![p_h^{n+1}]\!] & \text{on } \partial\Omega_e^\Gamma, \\ \nabla p_h^{n+1} - 2\tau_{\text{IP}}(p_h^{n+1} - g_p(t^{n+1})) \otimes \boldsymbol{n}_\Gamma & \text{on } \partial\Omega_e^N, \\ -\left(\frac{\partial \boldsymbol{g}_{\boldsymbol{u}}(t^{n+1})}{\partial t} + \sum_{i=0}^{J-1} \beta_i\big(\nabla \cdot \boldsymbol{\mathcal{F}}^c(\boldsymbol{u}_h^{n-i}) + \nu\nabla \times \boldsymbol{\omega}_h^{n-i}\big) - \boldsymbol{f}^{n+1}\right) & \text{on } \partial\Omega_e^D, \end{cases} \tag{3.13}$$

with the interior penalty parameter for hexahedra [113]

$$\tau_{\text{IP},e} = (k+1)^2 \frac{A(\partial\Omega_e^\Gamma)/2 + A(\partial\Omega_e^N \cup \partial\Omega_e^D)}{V(\Omega_e)} \tag{3.14}$$

including surface area A and element volume V. We apply the maximum penalty parameter across element boundaries

$$\tau_{\text{IP}} = \begin{cases} \max(\tau_{\text{IP},e}^-, \tau_{\text{IP},e}^+) & \text{on } \partial\Omega_e^\Gamma \text{ and} \\ \tau_{\text{IP},e}^- & \text{on } \partial\Omega_e^N \cup \partial\Omega_e^D. \end{cases} \tag{3.15}$$

The vorticity $\boldsymbol{\omega}_h \in \mathcal{V}_h^{\boldsymbol{u}}$ present in the boundary condition (3.13) is evaluated employing a local L^2-projection as suggested in Fehn [71] by

$$(\boldsymbol{v}_h, \boldsymbol{\omega}_h)_{\Omega_e} = (\boldsymbol{v}_h, \nabla \times \boldsymbol{u}_h)_{\Omega_e} \tag{3.16}$$

in order to avoid the necessity of computing the second derivatives directly.

Using the newly computed pressure p_h^{n+1}, the velocity is projected onto the solenoidal space by an element-wise operation according to

$$(\boldsymbol{v}_h, \hat{\boldsymbol{u}}_h)_{\Omega_e} = (\boldsymbol{v}_h, \hat{\boldsymbol{u}}_h)_{\Omega_e} \underbrace{-(\boldsymbol{v}_h, \frac{\Delta t}{\gamma_0}\nabla p_h^{n+1})_{\Omega_e}}_{=:b(\boldsymbol{v}_h, p_h^{n+1})}, \tag{3.17}$$

with $\hat{\boldsymbol{u}}_h \in \mathcal{V}_h^{\boldsymbol{u}}$. This equation is studied in detail in this work and we are going to modify the right-hand-side term $b(\boldsymbol{v}_h, p_h^{n+1})$ as well as add supplementary terms to ensure stability in the small-time-step limit and underresolved case in Section 3.4.

iii. *Implicit Viscous Step*

The weak form of the viscous step (3.6) reads

$$\begin{aligned}\left(\boldsymbol{v}_h, \frac{\gamma_0}{\Delta t}\boldsymbol{u}_h^{n+1}\right)_{\Omega_e} + \left(\boldsymbol{\epsilon}\left(\boldsymbol{v}_h\right), \boldsymbol{\mathcal{F}}^{\nu}\left(\boldsymbol{u}_h^{n+1}\right)\right)_{\Omega_e} - s\left(w\boldsymbol{\mathcal{F}}^{\nu}\left(\boldsymbol{v}_h\right), \boldsymbol{\mathcal{U}}^*\right)_{\partial\Omega_e} \\ - \left(\boldsymbol{v}_h, \boldsymbol{\mathcal{F}}^{\nu*}\left(\boldsymbol{u}_h^{n+1}\right)\cdot\boldsymbol{n}_\Gamma\right)_{\partial\Omega_e} = \left(\boldsymbol{v}_h, \frac{\gamma_0}{\Delta t}\hat{\boldsymbol{u}}_h\right)_{\Omega_e}.\end{aligned} \tag{3.18}$$

The parameter s is chosen either as 1 or -1 corresponding to the symmetric (SIPG) or nonsymmetric interior penalty (NIPG) methods (see, e.g., [220]). In the remainder of this chapter, we will solely consider the symmetric version $s = 1$ since it qualifies for efficient solution procedures of the linear system via a conjugate gradient solver and enables optimal convergence rates of orders $k + 1$ in the L^2 norm [109]. The nonsymmetric variant may nevertheless be useful since it provides a stable method with relaxed requirements on the penalty parameter τ_{IP}. For example, the extension of the present scheme towards wall modeling via function enrichment presented in Chapters 8 to 10 uses $s = -1$, ensuring coercivity of the bilinear form and thus stability of the numerical method in the context of nonpolynomial shape functions. The velocity jump function $\boldsymbol{\mathcal{U}}^*$ is defined as

$$\boldsymbol{\mathcal{U}}^* = \begin{cases} [\![\boldsymbol{u}_h^{n+1}]\!] & \text{on } \partial\Omega_e^{\Gamma}, \\ (2\boldsymbol{u}_h^{n+1} - 2\boldsymbol{g}_{\boldsymbol{u}}(t^{n+1})) \otimes \boldsymbol{n}_\Gamma & \text{on } \partial\Omega_e^{D} \text{ and} \\ \boldsymbol{0} & \text{on } \partial\Omega_e^{N}. \end{cases} \tag{3.19}$$

The numerical flux $\boldsymbol{\mathcal{F}}^{\nu*}(\boldsymbol{u}_h^{n+1})$ includes a penalty term as already used for the pressure Poisson equation (3.13)

$$\boldsymbol{\mathcal{F}}^{\nu*}(\boldsymbol{u}_h^{n+1}) = \begin{cases} \{\{\boldsymbol{\mathcal{F}}^{\nu}(\boldsymbol{u}_h^{n+1})\}\} - \tau_{\mathrm{IP}}\nu[\![\boldsymbol{u}_h^{n+1}]\!] & \text{on } \partial\Omega_e^{\Gamma}, \\ \boldsymbol{\mathcal{F}}^{\nu}(\boldsymbol{u}_h^{n+1}) - 2\tau_{\mathrm{IP}}\nu(\boldsymbol{u}_h^{n+1} - \boldsymbol{g}_{\boldsymbol{u}}(t^{n+1})) \otimes \boldsymbol{n}_\Gamma & \text{on } \partial\Omega_e^{D} \text{ and} \\ (\boldsymbol{h}(t^{n+1}) + g_p(t^{n+1})\boldsymbol{n}_\Gamma) \otimes \boldsymbol{n}_\Gamma & \text{on } \partial\Omega_e^{N}, \end{cases} \tag{3.20}$$

with τ_{IP} as defined in Equations (3.14) and (3.15). As opposed to the Laplace formulation of the viscous term ($\boldsymbol{\mathcal{F}}^{\nu}(\boldsymbol{u}) = \nu\nabla\boldsymbol{u}$), the present formulation ($\boldsymbol{\mathcal{F}}^{\nu}(\boldsymbol{u}) = \nu(\nabla\boldsymbol{u}+(\nabla\boldsymbol{u})^T)$) has been used less frequently in conjunction with the SIPG method. The stability of the present formulation has yet been proven in the context of cut-FEM in [226] and references therein.

3.3.3. The CFL Condition and Adaptive Time Stepping

The explicit treatment of the convective step restricts the time step size according to the Courant-Friedrichs-Lewy (CFL) condition to

$$\frac{\mathrm{Cr}}{k^2} = \frac{U\Delta t}{h} \tag{3.21}$$

with the Courant number $\mathrm{Cr} = \mathcal{O}(1)$, a characteristic element length h and velocity U, a relation that has been obtained via analysis of linear convection [112]. The CFL restriction scales by theory with the square of the polynomial degree and behaves similarly as for continuous spectral elements [136]. In the DG context, the relevant length is the minimal distance between nodal points inside an element [112].

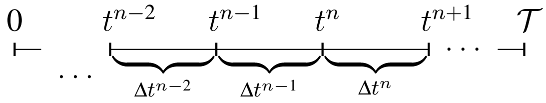
The time step size restriction coming along with this CFL condition may be quite painful, so the condition deserves some more attention. In particular, the most severe problem with Equation (3.21) is that the velocity field is not known *a priori*, such that the time step size has to be chosen based on a rough estimation of the velocity field with a large safety factor, in order to allow a computation to complete. The result is a waste of computational efficiency. Therefore, an adaptive algorithm has been developed, which chooses the largest allowable time step size in each step. In this way, the computational efficiency of the flow solver is considerably increased, no knowledge of the flow is required before the simulation run, and the CFL condition does not cause a simulation to fail.

As a result of a detailed analysis carried out by Legat [173] as part of the present work, the CFL condition is reformulated, reading

$$\frac{\mathrm{Cr}}{k^{1.5}} = \max_j \left|(\boldsymbol{J}^{-T}\boldsymbol{u}_h^n)_j\right| \Delta t, \tag{3.22}$$

where the transposed inverse of the Jacobian is employed to transform $\boldsymbol{u}_h^n$ into the parameter space of each element. Here, the largest absolute vector component of $\boldsymbol{J}^{-T}\boldsymbol{u}_h^n$ as velocity-to-length ratio has performed best in the investigations regarding curved boundaries and anisotropic meshes. The exponent of the polynomial degree of 1.5 has been determined experimentally as giving the tightest fit for $k \leq 8$. A similar trend can be observed in the data provided by [136], and this exponent is lower than the theoretical value of 2 in Equation (3.21).

The time step size Δt resulting from a constant Courant number is computed in each step and applied for the time-advancement from t^n to t^{n+1}. The variable time step increments require the recalculation of the time integration constants α_i, β_i, and γ_0, which are unique in each time step. In order to describe the relevant relations, the time increments along the temporal axis are denoted as Δt^n between the time instants t^n and t^{n+1} according to



following [173], such that the symbol Δt used in the previous Sections is equal to Δt^n. With this definition available, the time integration constants are given by [257] and are listed in Tables 3.2 and 3.3. It is noted that this adaptation scheme allows large ratios of successive time step sizes, which are derived in [257], such that no further modifications of the method are necessary. Problems with this adaptation scheme as reported in [73] were not observed.

Table 3.2.: Time integration parameters α_i for time stepping with variable time step size [257].

J, J_p	α_0	α_1	α_2
1	1	0	0
2	$\frac{\sum_{i=0}^{1}\Delta t^{n-i}}{\Delta t^{n-1}}$	$-\frac{(\Delta t^n)^2}{\Delta t^{n-1}\sum_{i=0}^{1}\Delta t^{n-i}}$	0
3	$\frac{\left(\sum_{i=0}^{1}\Delta t^{n-i}\right)\left(\sum_{i=0}^{2}\Delta t^{n-i}\right)}{\Delta t^{n-1}\sum_{i=1}^{2}\Delta t^{n-i}}$	$-\frac{(\Delta t^n)^2\sum_{i=0}^{2}\Delta t^{n-i}}{\Delta t^{n-1}\Delta t^{n-2}\sum_{i=0}^{1}\Delta t^{n-i}}$	$\frac{(\Delta t^n)^2\sum_{i=0}^{1}\Delta t^{n-i}}{\Delta t^{n-2}\left(\sum_{i=0}^{2}\Delta t^{n-i}\right)\left(\sum_{i=1}^{2}\Delta t^{n-i}\right)}$

Table 3.3.: Time integration parameters β_i and γ_0 for time stepping with variable time step size [257].

J, J_p	β_0	β_1	β_2	γ_0
1	1	0	0	1
2	$\frac{\sum_{i=0}^{1}\Delta t^{n-i}}{\Delta t^{n-1}}$	$-\frac{\Delta t^n}{\Delta t^{n-1}}$	0	$\frac{2\Delta t^n+\Delta t^{n-1}}{\sum_{i=0}^{1}\Delta t^{n-i}}$
3	$\frac{\left(\sum_{i=0}^{1}\Delta t^{n-i}\right)\left(\sum_{i=0}^{2}\Delta t^{n-i}\right)}{\Delta t^{n-1}\sum_{i=1}^{2}\Delta t^{n-i}}$	$-\frac{\Delta t^n\sum_{i=0}^{2}\Delta t^{n-i}}{\Delta t^{n-1}\Delta t^{n-2}}$	$\frac{\Delta t^n\sum_{i=0}^{1}\Delta t^{n-i}}{\Delta t^{n-2}\sum_{i=1}^{2}\Delta t^{n-i}}$	$1+\frac{\Delta t^n}{\sum_{i=0}^{1}\Delta t^{n-i}}+\frac{\Delta t^n}{\sum_{i=0}^{2}\Delta t^{n-i}}$

The present adaptation scheme exhibits Courant numbers, which are approximately independent of the numerical example, of about $\mathrm{Cr} = 0.14$ for $J = 2$ (BDF2) and $\mathrm{Cr} = 0.09$ for $J = 3$ (BDF3), where the inverse Jacobian represents a transformation from physical space to a parameter space of unit length $([0, 1])$. For a parameter space ranging from $[-1, 1]$, the equivalent numerical values would be $\mathrm{Cr} = 0.28$ and $\mathrm{Cr} = 0.18$.

3.4. Four Variants of the Projection Step and Impact on Stability

We pay special attention to the small-time-step as well as the spatially underresolved limit and associated instabilities of the 'standard' version of the present scheme as discussed in a series of recent papers [66, 67, 72, 75, 76, 129, 147, 202, 246]. The investigations carried out in this thesis have confirmed that these aspects are of high relevance regarding an accurate and robust numerical method, especially considering underresolved turbulent flows at high Reynolds number. In the first Subsection 3.4.1 we give numerical evidence that the instabilities may be traced back to two particular sources within the right-hand side of the Poisson equation (3.3). Remedies presented in the literature as well as several extensions are reviewed and compared in Subsection 3.4.2. The two most promising stabilization techniques are selected in Subsection 3.4.3 which will be validated and compared thoroughly in Section 3.6 and Chapter 4 for laminar and turbulent flow, respectively. The matrix formulation is outlined in Subsection 3.4.4.

The preliminary numerical investigations regarding small time steps shown in this section are performed using the laminar vortex problem according to [112] and described in detail in Section 3.6.1 with a domain size of $[-0.5, 0.5] \times [-0.5, 0.5]$ and discretizations of $N_e = 4^2$ elements as well as the polynomial degrees $k = \{2, 3, 4\}$ and a kinematic viscosity of $\nu = 0.025$. The cases are labeled accordingly specifying the number of elements $N4^2$, the polynomial order $k\{2, 3, 4\}$ and the variant under investigation, i.e., *VHW* for the 'standard' variant of Hesthaven

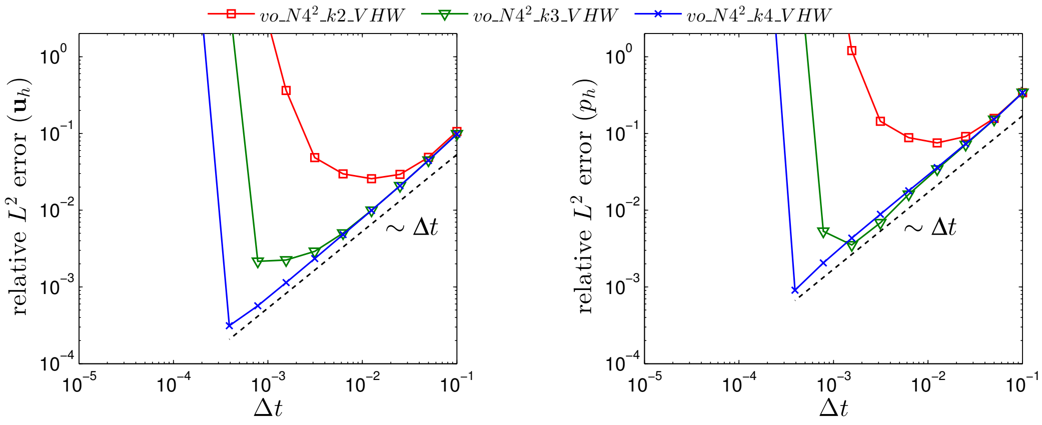


Figure 3.1.: Instability behavior for small time steps using the ‘standard’ variant by Hesthaven and Warburton [112] (*VHW*) for three spatial discretizations. Data taken from [147].

and Warburton [112] given in Section 3.3 or *V1* to *V4* discussed in the following. A first-order time integration scheme is used with $J = J_p = 1$ (BDF1 and EX1) for these developments to show the influence of the temporal discretization error, which would be negligible for $J = \{2, 3\}$, while results presented in the remainder of this thesis will mainly employ the second and third-order accurate scheme. Regarding the CFL condition, this investigation employs the simple CFL criterion based on constant time steps, given in Equation (3.21).

The second numerical example considered in this section is turbulent channel flow at a friction Reynolds number of $\mathrm{Re}_\tau = 180$ and investigates marginal spatial resolution. We employ a discretization of 8^3 elements of degree $k = 3$ and a Courant number of the order of unity for the BDF3 scheme. Further details on the configuration are given in Section 4.2.

3.4.1. Sources of Instabilities

The two modes of instability are introduced and an overview of remedies considered in this work is given.

3.4.1.1. The Small-Time-Step Limit

The occurrence of instabilities for small time step sizes has first been described by Ferrer and Willden [76] and has been investigated further by Ferrer et al. [75]. The source of these instabilities may be identified according to [147] by rewriting the strong form of the pressure Poisson equation (Equation (3.3)) by inserting the first intermediate velocity $\hat{\boldsymbol{u}}$ of the convective step (Equation (3.2)):

$$-\nabla^2 p^{n+1} = -\sum_{i=0}^{J-1}\left(\frac{\alpha_i}{\Delta t}\nabla\cdot\boldsymbol{u}^{n-i}\right) - \nabla\cdot\left(-\sum_{i=0}^{J-1}\beta_i\nabla\cdot\boldsymbol{\mathcal{F}}^c(\boldsymbol{u}^{n-i}) + \boldsymbol{f}^{n+1}\right). \qquad (3.23)$$

It is noted that the first term on the right-hand side includes a scaling of the divergence of $\boldsymbol{u}^{n-i}$ with $1/\Delta t$. So, if there are spurious divergence errors through the finite spatial discretization, these are amplified, which may result in an inaccurate scheme and eventually render the method unstable for very small time steps if no additional measures are taken. The velocity divergence errors $\nabla \cdot \boldsymbol{u}^{n-i}$ may be analyzed in more detail by taking the divergence of the equation for the viscous step (3.6) according to [147], yielding

$$\frac{\gamma_0}{\Delta t} \nabla \cdot \boldsymbol{u}^{n+1} - 2\nu \nabla^2 (\nabla \cdot \boldsymbol{u}^{n+1}) = \frac{\gamma_0}{\Delta t} \nabla \cdot \hat{\hat{\boldsymbol{u}}}. \tag{3.24}$$

Following [147], the second intermediate velocity of the dual-splitting scheme would be exactly divergence-free, $\nabla \cdot \hat{\hat{\boldsymbol{u}}} = 0$, if there was no spatial discretization error. However, the discrete spatial resolution present in the numerical method yields an intermediate velocity, which is not exactly divergence free after the projection step [246]. As it is observed in Equation (3.24), these divergence errors are directly transferred to the final velocity solution $\boldsymbol{u}^{n+1}$. In Equation (3.23), it is observed that the divergence errors are amplified in the first term of the right-hand side by a factor of $1/\Delta t$, which is the source of the instabilities at small time steps [147]. This behavior is observed in numerical experiments by [147], which are shown in Figure 3.1, where the laminar vortex problem has been computed using the ‘standard’ method as presented in Section 3.3.2.

We consider the following remedies to this problem, which will be discussed in more detail in the subsequent Section 3.4.2:

- *V1*: Ferrer et al. [75] propose to increase the penalty parameter τ_{IP} of the discrete Laplace operator of the pressure Poisson equation to circumvent the instabilities. In the same publication, it is stated that this type of instability is related to the inf-sup condition, which we cannot confirm as mixed-order elements of degrees k and $k-1$ for velocity and pressure, respectively, also lead to instabilities in the limit of small time steps. However, the inf-sup problem can indeed yield pressure oscillations at even smaller time steps with this scheme [72].

- *V2*: The problematic term is dropped using the condition $\nabla \cdot \boldsymbol{u}^{n-i} = 0$, which corresponds to the approach proposed by Leriche and Lambrosse [176] as well as Leriche et al. [177] in the context of the unsteady Stokes equations.

- *V3* and *V4*: The divergence error is controlled by an additional and consistent div-div penalty term inspired by works of Steinmoeller et al. [246] as well as Joshi et al. [129]. The term is similar to the popular grad-div term in continuous Galerkin and enables a stable numerical method.

3.4.1.2. Conservation of Mass in the Underresolved Limit

The second aspect of the present DG method requiring special attention is mass conservation in the underresolved limit. The error stemming from the continuity equation in the discontinuous context may for example be described in an element-wise sense as

$$e_e^{\text{continuity}} = \int_{\Omega_e} |\nabla \cdot \hat{\hat{\boldsymbol{u}}}_h| \mathrm{d}\Omega_e + \int_{\partial\Omega_e} \frac{1}{2} |[\hat{\hat{\boldsymbol{u}}}_h] \cdot \boldsymbol{n}_\Gamma| \mathrm{d}\Gamma \tag{3.25}$$

and consists of two contributions: The first term represents the divergence error within elements and the second term accounts for the mass balance across element interfaces. The factor $1/2$ is

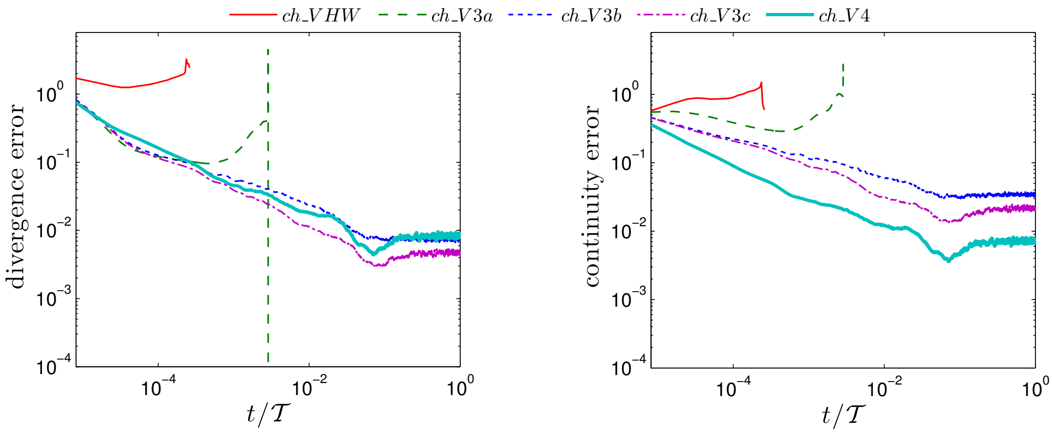


Figure 3.2.: Stability investigation for the spatially underresolved limit for variants *VHW*, *V3a* to *V3c* and *V4* (from left to right) of turbulent channel flow at $\mathrm{Re}_\tau = 180$ and a spatial discretization of 8^3 $k = 3$ elements. The divergence and continuity errors are defined in Equations (3.26) and (3.27), respectively.

included in the latter term since the error appears on two neighboring elements. In the Poisson equation as described in Equation (3.11) no control on the second term is included in the DG context. As a consequence, an instability may arise in underresolved turbulent simulations since marginal resolution generally results in more pronounced velocity discontinuities that increase the impact of the second term in Equation (3.25). This type of instability has recently been examined by Joshi et al. [129] and is investigated numerically in Figure 3.2 with a marginally resolved turbulent channel flow simulation. For these computations, the error in mass conservation is, according to Equation (3.25), defined separately as the divergence error

$$\frac{\delta \int_{\Omega_h} |\nabla \cdot \hat{\boldsymbol{u}}_h| \mathrm{d}\Omega}{\int_{\Omega_h} ||\hat{\boldsymbol{u}}_h|| \mathrm{d}\Omega} \tag{3.26}$$

with the channel-half width δ and the continuity error

$$\frac{\int_{\partial\Omega_h^\Gamma} |[\hat{\boldsymbol{u}}_h] \cdot \boldsymbol{n}_\Gamma| \mathrm{d}\Gamma}{\int_{\partial\Omega_h^\Gamma} |\{\{\hat{\boldsymbol{u}}_h\}\} \cdot \boldsymbol{n}_\Gamma| \mathrm{d}\Gamma} \tag{3.27}$$

measuring the loss of mass in between elements. In Figure 3.2, the 'standard' variant *VHW* exhibits large divergence and continuity errors, which lead to a diverging solution.

In the present contribution, we review and compare two particular remedies to this issue, which will be discussed in more detail in the subsequent Section 3.4.2. Their common purpose of reducing the inter-element continuity error is attained by different approaches:

- *V3b* and *V3c*: We reformulate the right-hand side of the Poisson equation (term $a(q_h, \hat{\boldsymbol{u}}_h)$ in Equation (3.11)) such that a term both for the velocity divergence and the discontinuity at element boundaries is taken into account, in analogy to the error definition listed in Equation (3.25). Steinmoeller et al. [246] use the equivalent strong form of this term. However, that work lacks an explanation as well as investigation of this issue. We also show that it may be beneficial to modify the right-hand side of the projection, given as $b(\boldsymbol{v}_h, p_h^{n+1})$ in Equation (3.17), yielding a complementary improvement of the method.

- *V4*: Joshi et al. [129] propose to penalize velocity jumps at element interfaces within the projection step, which may be seen as a straightforward measure to the underlying problem, however at the cost of an additional global system to solve.

Remark: The two modes of instability have been introduced here separately despite an undoubted mutual dependence. In order to obtain a robust numerical method, it is considered a necessity to include a measure for both limits. For example, the primary instability mode under investigation in Steinmoeller et al. [246] cannot be determined since they operate in the underresolved high-Reynolds-number regime and, in addition, do not specify the time step size used for their computations in terms of the Courant number. Indeed, their proposal is similar to *V3b* discussed in the following including measures for both instabilities.

3.4.2. Four Variants

3.4.2.1. Variant 1 (V1)

Instabilities with the present dual-splitting scheme in the small time step regime have for the first time been reported by Ferrer and Willden [76] considering an unsteady Stokes flow example. This instability analysis is extended in Ferrer et al. [75], where it is suggested that the time step size should not be smaller than a critical value of $\nu \Delta t_{\text{lim}} \sim h^2/k^3$ to guarantee the stability of the method. As a result, a high spatial resolution is required, which is the opposite of what we are aiming for in this thesis, considering implicit LES, i.e., underresolved turbulent flow at high Reynolds number. In order to still allow the use of coarser spatial discretizations, it is in [75] suggested to scale the interior penalty parameter τ_{IP} (3.15) of the Poisson problem (3.11) by a factor of $1/\nu\Delta t$. As this scaling factor is not dimensionless, this approach has been slightly modified by [147] in order to make the approach more general and to assess the stability of the stabilization technique. The interior penalty parameter is scaled by

$$\tau_{\text{IP},V1} = \tau_{\text{IP}} \frac{\Delta t_{\text{ref}}}{\Delta t}, \tag{3.28}$$

where Δt_{ref} is a reference time step, which allows the stable computation of the respective example.

Numerical experiments carried out by [147] are shown in Figure 3.3 and indicate that the small-time-step limit is relaxed in comparison to Figure 3.1. All simulations eventually get unstable, though. Moreover, it is argued in [147] that the most problematic aspect of this stabilization approach lies in the drastic increase of computational cost per time step when small time step sizes are employed. Assuming that the condition number of the discrete Laplace operator is proportional to τ_{IP} [112], the cost per time step increases as $(1/\Delta t)^{1/2}$ when applying an iterative Krylov method to the numerical solution of the pressure Poisson equation (see, e.g., [233]). Finally, this type of stabilization approach does not consider inter-element mass conservation, which would have to be included analogous to *V3b*, *V3c* or *V4*.

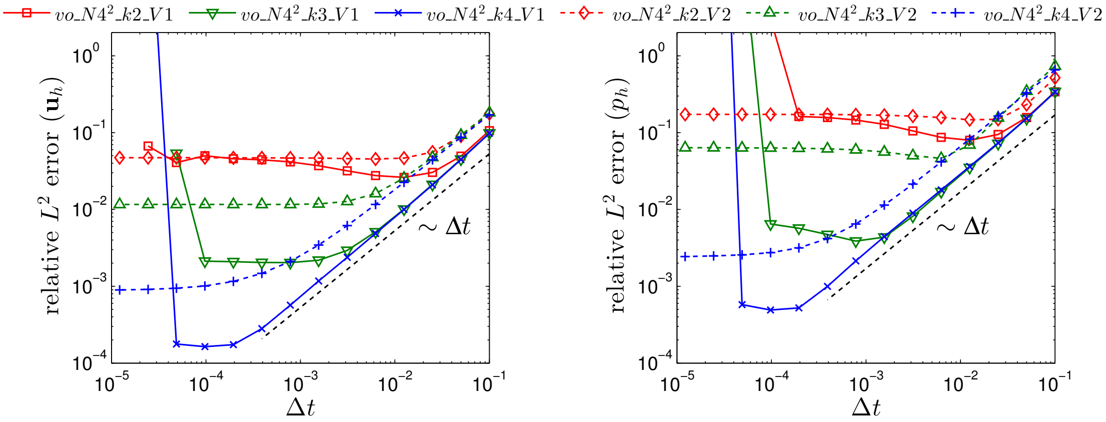


Figure 3.3.: Stability experiments for small time steps using *V1* and *V2* for three spatial discretizations. Data taken from [147].

3.4.2.2. Variant 2 (V2)

The most straightforward approach to tackle the problematic source term on the right-hand-side of the pressure Poisson equation given in Equation (3.23), as well as to verify the hypothesis regarding the small-time-step instability, would be to exploit the condition $\nabla \cdot \boldsymbol{u}^{n-i} = 0$ and to drop the problematic term entirely. This idea leads to a time integration scheme equivalent to the method proposed in Leriche and Lambrosse [176] and Leriche et al. [177] for the unsteady Stokes equations. Regarding the present case of the incompressible Navier–Stokes equations, a modified scheme has been derived in [147] based on this idea.

The numerical results for this scheme according to [147] are depicted in Figure 3.3 including a comparison to *V1*. It may be observed that there is no instability with this scheme for small time steps and the error perfectly converges to a constant level, where the spatial error is dominant. The results therefore support the hypothesis that the divergence term on the right-hand side of the Poisson equation (3.23) causes instabilities in the small-time-step limit [147]. However, Figure 3.3 also reveals a major drawback of the modified time integration scheme: the L^2 error is generally larger compared to *V1* for large time steps, an observation already reported by Leriche et al. [177]. In further numerical experiments carried out in this work, optimal convergence rates of order $k+1$ in space were not obtained for velocity or pressure. The idea of the schemes by Leriche and Lambrosse [176] and Leriche et al. [177] has recently been considered by Emamy et al. [67], who presented similar results. If this scheme would be applied to spatially underresolved high-Reynolds-number flows, an additional stabilization of velocity discontinuities as included in *V3b, V3c*, or *V4* would be necessary nonetheless.

3.4.2.3. Variant 3 (V3)

Variant 3a. Steinmoeller et al. [246] propose to postprocess the second intermediate velocity $\hat{\hat{\boldsymbol{u}}}$ to a point-wise exactly divergence-free velocity field as a means to stabilize the method. Despite the introduction of this idea as an enhancement of the splitting scheme for coarse resolutions and high Reynolds numbers in [246], we demonstrate that this approach also stabilizes in the small-time-step limit.

The postprocessing step applied in [246] appears costly in three space dimensions, however, since a total of nine different shape functions for each polynomial order of the modal space would be necessary. Instead of projecting the velocity field onto an exactly divergence-free basis, we perform this postprocessing in an approximate and very efficient way by including a supplementary div-div penalty term in the projection step, which is similar to the frequently used grad-div stabilization in the context of the continuous FEM (see, e.g., [101, 197]) to enhance mass conservation. A similar term is also included in the weak projection by Joshi [129] which is discussed in *V4* and a coupled DG solver presented in [230]. The local projection step becomes

$$(\boldsymbol{v}_h, \hat{\hat{\boldsymbol{u}}}_h)_{\Omega_e} + \underbrace{(\nabla \cdot \boldsymbol{v}_h, \tau_D \nabla \cdot \hat{\hat{\boldsymbol{u}}}_h)_{\Omega_e}}_{\text{div-div penalty}} = (\boldsymbol{v}_h, \hat{\boldsymbol{u}}_h)_{\Omega_e} \underbrace{-(\boldsymbol{v}_h, \frac{\Delta t}{\gamma_0} \nabla p_h^{n+1})_{\Omega_e}}_{=b(\boldsymbol{v}_h, p_h^{n+1})}, \tag{3.29}$$

where τ_D is a penalty parameter. It is noted that this represents a consistent modification of the projection step since it involves the continuity residual $\nabla \cdot \hat{\hat{\boldsymbol{u}}}_h$. Further, the velocity field approaches the point-wise exactly divergence-free one with increasing τ_D while the simultaneous degradation of the condition of the matrix system plays a minor role due to the locality of the problem.

We exploit the similarity of the present penalty term to the grad-div stabilization and define the penalty parameter according to [197] for equal-order elements as

$$\tau_D = \zeta_D \|\overline{\boldsymbol{u}}_h^n\| h \Delta t, \tag{3.30}$$

where $\|\overline{\boldsymbol{u}}_h^n\|$ is the norm of the element-wise volume-averaged velocity, $h = V(\Omega_e)^{1/3}$ is a characteristic element length defined as the cube root of the respective element volume, and the proportionality parameter ζ_D may be used to control the final divergence error. We further note that the factor Δt is introduced in the parameter, since the projection step is multiplied by the time step size during derivation. In addition, the viscous contribution considered in [197] is omitted in this work, as the focus herein lies on convection-dominated flows.

The impact of the proportionality parameter ζ_D on the small-time-step limit is investigated in Figure 3.4 with data from [147] by comparing $\zeta_D = \{0, 1, 10, 100, 1/\mathrm{Cr}\}$ using the present variant *V3a*. We observe a drastic improvement of the stability behavior of the splitting scheme for increasing penalty parameters and, in particular, stable results are obtained with $\zeta_D = 1/\mathrm{Cr}$ for all time step sizes considered in this investigation. From this result we draw the conclusion that the first term on the right-hand side of Equation (3.23), which appears as a source term in the pressure Poisson equation, is counter-balanced universally by a scaling of ζ_D with $1/\mathrm{Cr}$. In the remainder of this thesis we therefore employ

$$\zeta_D = \zeta_D^*/\mathrm{Cr} \tag{3.31}$$

with $\zeta_D^* = 1$ if not specified otherwise. Further numerical evidence for these arguments is shown in Figure 3.5, again with data from [147], where all cases *V3a* exhibit an ideal behavior for small time steps. With respect to the different choices of the Courant number in Equations (3.21) and (3.22), the user-specified Courant number is used, making the implementation independent of any specific CFL criterion.

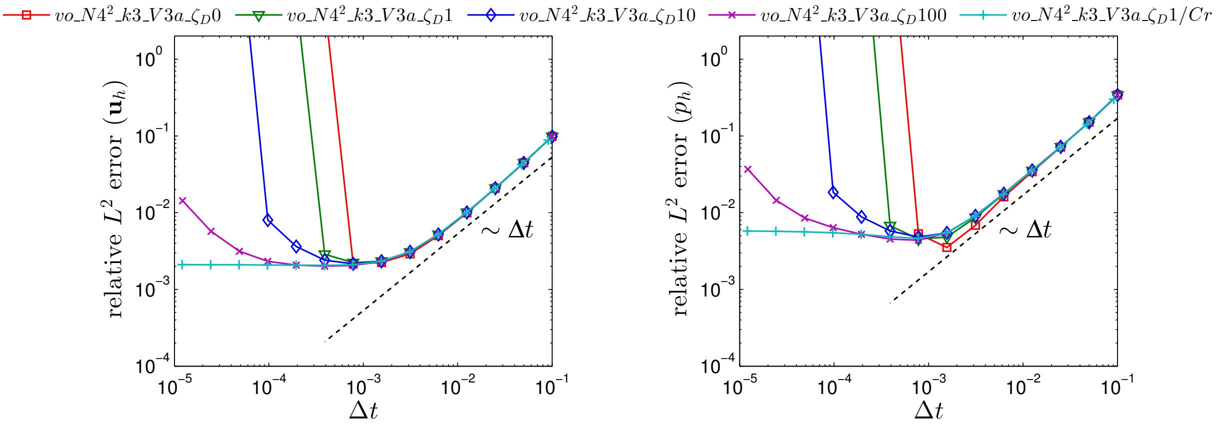


Figure 3.4.: Impact of proportionality parameter ζ_D included in *V3* and *V4* on small-time-step stability. Legend from left to right: $\zeta_D = 0$, $\zeta_D = 1$, $\zeta_D = 10$, $\zeta_D = 100$, $\zeta_D = 1/\mathrm{Cr}$. Data taken from [147].

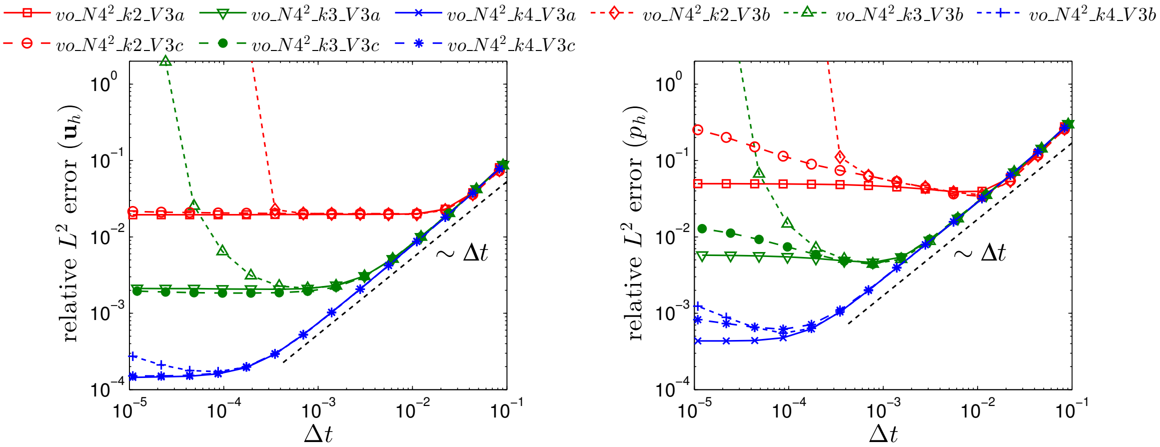


Figure 3.5.: Stability experiments for small time steps using *V3a*, *V3b*, and *V3c* for three spatial discretizations. Data taken from [147].

Remark: An interesting interpretation of the grad-div stabilization again in the context of continuous Galerkin is also given in [197] where the necessity of this term is related to an insufficient resolution of the pressure field. It is further shown that the grad-div term for continuous Galerkin may be seen as the subgrid component for the pressure. Although the transfer of this idea would certainly be illustrative in the context of the current div-div penalty term, we do not elaborate this concept within this work.

Variant 3b. Since *V3a* does not contain a measure for controlling the inter-element mass conservation according to Section 3.4.1.2, we show in *V3b* a modification of *V3a*, which takes this aspect into account. Steinmoeller et al. [246] and Emamy [66] reformulate the right-hand side of the Poisson equation (3.11), $a(q_h, \hat{\boldsymbol{u}}_h)$, by integration by parts and consideration of a central flux formulation, yielding

$$a(q_h, \hat{\boldsymbol{u}}_h) = (\nabla q_h, \frac{\gamma_0}{\Delta t}\hat{\boldsymbol{u}}_h)_{\Omega_e} - (q_h, \frac{\gamma_0}{\Delta t}\{\{\hat{\boldsymbol{u}}_h\}\}^{ND} \cdot \boldsymbol{n}_\Gamma)_{\partial\Omega_e}. \tag{3.32}$$

This expression may be recast into the strong formulation by integrating by parts once again

$$a(q_h, \hat{\boldsymbol{u}}_h) = -(q_h, \frac{\gamma_0}{\Delta t}\nabla \cdot \hat{\boldsymbol{u}}_h)_{\Omega_e} + (q_h, \frac{\gamma_0}{\Delta t}\frac{1}{2}[\hat{\boldsymbol{u}}_h]^{ND} \cdot \boldsymbol{n}_\Gamma)_{\partial\Omega_e}. \tag{3.33}$$

While the strong formulation is mathematically equivalent to Equation (3.32), it highlights that the right-hand side of the Poisson equation now is of the same structure as the continuity error defined in Equation (3.25), considering terms including the velocity divergence and discontinuity. The additional source term in Equation (3.33) thus results in a modified pressure field that takes into account mass conservation in between elements, which comes along with a potentially less smooth pressure field in comparison with the 'standard' variant of $a(q_h, \hat{\boldsymbol{u}}_h)$. In the numerical investigations presented in Figure 3.2 it is found that this definition of $a(q_h, \hat{\boldsymbol{u}}_h)$ indeed improves mass conservation across element boundaries, compared to the standard formulation *V3a*, and results in a constant error level after an initial transient. However, it is observed in data by [147], shown in Figure 3.5, that this modification of the Poisson equation degrades the stability for small time steps. This makes a further modification of the projection step necessary, which is discussed in the following variant *V3c*.

We note that the formulation of the flux in Equation (3.32) has recently been refined by the derivation of a fully consistent boundary condition [72] in replacement of the boundary flux $\{\{\hat{\boldsymbol{u}}_h\}\}^{ND} = \hat{\boldsymbol{u}}_h^-$ employed in the present work. In future work, this new boundary condition should be employed.

Variant 3c. We also consider the partial integration of $b(\boldsymbol{v}_h, p_h^{n+1})$ of the projection step and employ a central flux similar to [66]

$$b(\boldsymbol{v}_h, p_h^{n+1}) = (\nabla \cdot \boldsymbol{v}_h, \frac{\Delta t}{\gamma_0}p_h^{n+1})_{\Omega_e} - (\boldsymbol{v}_h, \frac{\Delta t}{\gamma_0}\{\{p_h^{n+1}\}\}^{ND}\boldsymbol{n}_\Gamma)_{\partial\Omega_e}, \tag{3.34}$$

which increases robustness in our most challenging test cases and yields more accurate results. We choose the partially integrated version of $a(q_h, \hat{\boldsymbol{u}}_h)$ according to Equation (3.32) and include a div-div penalty in the projection (3.29). The results presented in Figure 3.5 [147] show that this definition of $b(\boldsymbol{v}_h, p_h^{n+1})$ cures the deficiencies observed with *V3b* for small time step sizes. Further, mass conservation in Figure 3.2 exhibits even lower error levels compared to *V3b* both for the divergence and continuity error.

It is noted that this combination of $a(q_h, \hat{\boldsymbol{u}}_h)$ and $b(\boldsymbol{v}_h, p_h^{n+1})$ represents a similar formulation as presented by Cockburn et al. [48] in the framework of a coupled mixed-order DG method. A supplementary pressure stabilization as proposed by Cockburn et al. [49] for equal-order coupled DG was not found to be necessary in the context of the present splitting scheme, however.

3.4.2.4. Variant 4 (V4)

A natural approach to handle both instabilities simultaneously is to include a div-div penalty and a supplementary jump-penalty term controlling both divergence and continuity errors within the

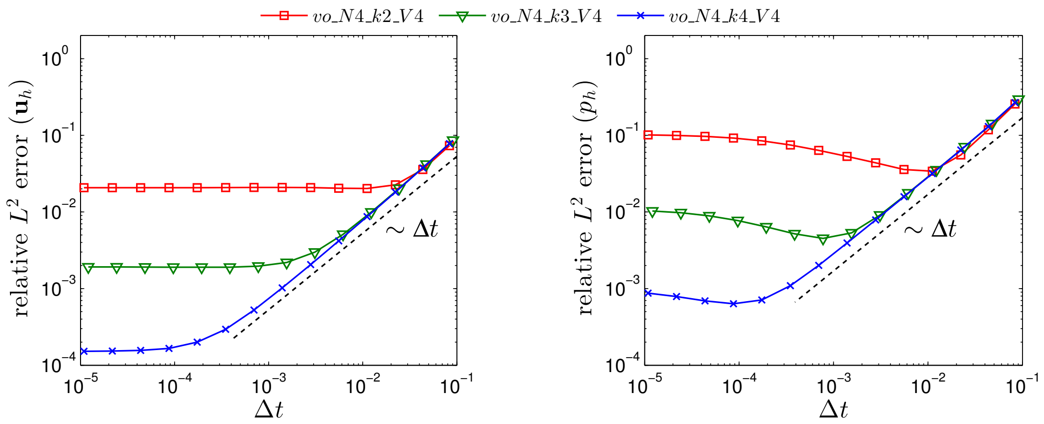


Figure 3.6.: Stability experiments for small time steps using *V4* for three spatial discretizations. Data taken from [147].

projection (3.17). Joshi et al. [129] have recently proposed a similar idea where these penalty terms are contained in a postprocessing step for the intermediate velocity $\hat{\hat{\boldsymbol{u}}}_h$ in order to weakly enforce the incompressibility condition inside cells and the continuity condition across element faces. The projection becomes

$$(\boldsymbol{v}_h, \hat{\hat{\boldsymbol{u}}}_h)_{\Omega_e} + \underbrace{(\nabla \cdot \boldsymbol{v}_h, \tau_D \nabla \cdot \hat{\hat{\boldsymbol{u}}}_h)_{\Omega_e}}_{\text{div-div penalty}} + \underbrace{(\boldsymbol{v}_h, \tau_C [\hat{\hat{\boldsymbol{u}}}_h]^{ND})_{\partial\Omega_e}}_{\text{jump penalty}} = (\boldsymbol{v}_h, \hat{\boldsymbol{u}}_h)_{\Omega_e} + b(\boldsymbol{v}_h, p_h^{n+1}) \tag{3.35}$$

with the continuity-penalty parameter τ_C. We define the latter in analogy to τ_D as [147]

$$\tau_{C,e} = \zeta_C \|\overline{\boldsymbol{u}}_h^n\| \Delta t, \tag{3.36}$$

with $\zeta_C = \zeta_C^*/\mathrm{Cr}$ similar to ζ_D, $\zeta_C^* = 1$ if not specified otherwise. It is noted in [147] that this penalty parameter assures consistent physical units within the projection in contrast to the choice in [129]. On internal faces we use the average according to [147]

$$\tau_C = \begin{cases} \{\{\tau_{C,e}\}\} & \text{on } \partial\Omega_e^\Gamma \text{ and} \\ \tau_{C,e}^- & \text{on } \partial\Omega_e^N \cup \partial\Omega_e^D. \end{cases} \tag{3.37}$$

Regarding the right-hand sides $a(q_h, \hat{\boldsymbol{u}}_h)$ and $b(\boldsymbol{v}_h, p_h^{n+1})$, Preliminary numerical investigations showed that optimal spatial convergence rates may only be obtained in the test case presented in Section 3.6.1.2 using the partially integrated versions according to Equations (3.32) and (3.34). In principle, the approach is also stable with the standard variants given in Equations (3.11) and (3.17), however. The jump-penalty term included in *V4* makes the projection step a more expensive global equation system compared to the purely local projection used in *V3*. The preliminary investigations in Figure 3.6 with data by [147] indicate ideal behavior for small time steps while maintaining low error levels throughout. According to Figure 3.2, conservation of mass gives a similar behavior as variants *V3b* and *V3c* which makes this approach a promising alternative to *V3c* despite the additional computational cost.

3.4.3. Conclusion on Variants 1 to 4

In this section, we have identified small time steps and coarse spatial resolutions as potential sources of instabilities and discussed a number of remedies. On the one hand, spurious divergence errors are amplified for small time steps leading eventually to an unstable scheme. On the other hand, pronounced velocity discontinuities in underresolved simulations give rise to an excessive violation of the continuity equation also leading to instabilities. While the first issue may be stabilized successfully via a div-div penalty term within the projection, two remedies exhibit promising characteristics for stabilization of marginally-resolved simulations: We have obtained a robust and fast computational method by partial integration of the right-hand side of the Poisson equation and projection, detailed in *V3c*, while a supplementary jump-penalty term within the projection step also yields a promising method at slightly higher computational cost according to *V4*.

We anticipate at this point that variant *V3c* is the working-horse for the computation of turbulent flows in this thesis, since it combines all highly desirable aspects of a numerical scheme, consisting of stability regarding small time steps and spatially underresolved simulations, optimal spatial convergence rates as well as low computational cost, thus most of the turbulent flow examples shown below are computed with this variant.

The topic of instabilities regarding the dual-splitting scheme in conjunction with high-order DG discretizations is a topic of active and high-paced research, such that several additional methods have recently been proposed, for example the approaches by Piatkowski et al. [202] and Emamy et al. [67]. These methods should be taken into consideration and a critical assessment has yet to be performed. In addition, the results of the recent analysis of similar continuity and divergence penalty terms as presented herein may be taken into account [5].

3.4.4. Matrix Formulation

The discussion on the spatial discretization in Sections 3.3 and 3.4 is concluded with the matrix formulation, which is the basis for the presentation of the solution procedures employed for the linear systems in the subsequent Section 3.5. The matrix formulation for the convective step (Equation (3.8)) results in

$$\gamma_0 \hat{\boldsymbol{U}} = \sum_{i=0}^{J-1} \alpha_i \boldsymbol{U}^{n-i} - \Delta t \boldsymbol{M}^{-1} \sum_{i=0}^{J-1} \beta_i \boldsymbol{F}^c(\boldsymbol{U}^{n-i})\boldsymbol{U}^{n-i} + \Delta t \boldsymbol{F}(t^{n+1}), \tag{3.38}$$

with the block-diagonal mass matrix $\boldsymbol{M}$, the evaluation of the convective term for the corresponding time step $\boldsymbol{F}^c$, the body-force vector $\boldsymbol{F}$ and the respective velocity vectors $\boldsymbol{U}$. The matrix form of the pressure Poisson equation (3.11) is given by

$$\boldsymbol{L}\boldsymbol{P}^{n+1} = \frac{\gamma_0}{\Delta t}\boldsymbol{A}\hat{\boldsymbol{U}} - \boldsymbol{L}_{\mathrm{BC}}\boldsymbol{G}_p(t^{n+1}), \tag{3.39}$$

with the discrete Laplace operator $\boldsymbol{L}$, the pressure solution vector $\boldsymbol{P}$, the respective variant of the velocity divergence operator $\boldsymbol{A}$ according to $a(q_h, \hat{\boldsymbol{u}}_h)$ in Section 3.4.2, and boundary terms $\boldsymbol{L}_{\mathrm{BC}}\boldsymbol{G}_p$. For the local projection (3.17), we get

$$(\boldsymbol{M} + \tau_D \boldsymbol{D} + \tau_C \boldsymbol{C})\,\hat{\hat{\boldsymbol{U}}} = \left(\boldsymbol{M}\hat{\boldsymbol{U}} + \frac{\Delta t}{\gamma_0}\boldsymbol{B}\boldsymbol{P}^{n+1}\right), \tag{3.40}$$

with the block-diagonal div-div penalty operator $\boldsymbol{D}$ only considered in *V3* and *V4*, the jump-penalty terms $\boldsymbol{C}$ solely included in *V4*, as well as the discrete pressure gradient $\boldsymbol{BP}$ according to $b(\boldsymbol{v}_h, p_h^{n+1})$. Finally, the Helmholtz-like equation of the viscous step reads in matrix form

$$\left(\frac{\gamma_0}{\Delta t}\boldsymbol{M} - \boldsymbol{F}^{\nu}\right)\boldsymbol{U}^{n+1} = \frac{\gamma_0}{\Delta t}\boldsymbol{M}\hat{\hat{\boldsymbol{U}}} + \boldsymbol{F}_{\mathrm{BC}}^{\nu}\boldsymbol{G}_{\boldsymbol{u}}(t^{n+1}), \tag{3.41}$$

with the linearized viscous term $\boldsymbol{F}^{\nu}$ and the right-hand-side boundary terms $\boldsymbol{F}_{\mathrm{BC}}^{\nu}\boldsymbol{G}_{\boldsymbol{u}}$.

3.5. Implementation

The solver presented in the previous sections has been implemented using the open-source deal.II finite element library [9] developed in the C++ programming language. We give an overview of the cell evaluation routines, solution strategies employed to solve the linear systems and present a performance evaluation. The discussion on the implementation and solution strategies of linear systems follows the ideas developed in [71] and [147].

3.5.1. Evaluation and Integration of Cells and Faces via Sum Factorization

The evaluation of cell and face integrals given in the weak forms of the splitting scheme is implemented using the computational kernels by Kronbichler and Kormann [156–158]. These kernels provide an efficient implementation of the evaluation of cell and face shape functions of the form

$$p_h(\boldsymbol{\xi}, t) = \sum_{l,m,n=0}^{k} N_{lmn}^{k}(\boldsymbol{\xi})p_{lmn}(t), \tag{3.42}$$

for the pressure, where the shape functions N_{lmn}^k are given through the tensor product of one-dimensional polynomials $\ell_l^k(\xi_i)$ in the unit-cell direction i, yielding $N_{lmn}^k(\boldsymbol{\xi}) = \ell_l^k(\xi_1)\ell_m^k(\xi_2)\ell_n^k(\xi_3)$, and p_{lmn} is the corresponding DOF. The polynomials considered in this work, $\ell_l^k(\xi_i)$, are nodal Langrange polynomials on $(k+1)$ Gauss–Lobatto points in each cell dimension. If polynomials of this structure are combined with quadrature rules of a similar tensor-product structure, the cells may be evaluated using sum-factorization techniques, which aim at reducing the algorithmic cost of reoccurring terms, an approach that has a long-going tradition in spectral-element methods [61, 78–81, 180, 198, 201, 251].

The computational kernels by Kronbichler and Kormann [156] within the deal.II library use several additional techniques, which aim at fast algorithms problem-tailored for current and future CPU architectures [158, 159]. One of these techniques is the parallel evaluation of cells and faces via vector instructions (AVX on Intel processors) on a single CPU core with individual parameters and geometries each, such that for example four cells or faces are evaluated on current CPU architectures (Intel Sandy Bridge, Haswell, or Broadwell achitecture), which are used for most computations presented in this work.

These computational kernels are used for all evaluations of integrals in weak forms, including matrix vector products. As quadrature formulas, Gaussian quadrature with $n_q = k+1$ quadrature points per dimension is used for all linear terms, yielding an accuracy of $2k+1$. In order to avoid aliasing effects due to underintegration of the nonlinear terms, $n_q = \left\lfloor \frac{3k}{2} \right\rfloor + 1$ quadrature points are employed for the convective terms, which yields exact quadrature on affine cells.

The software toolbox [156, 157] further provides a specialized algorithm for the action of applying the inverse mass matrix. As the mass matrix does not include coupling between cells, the matrix is block-diagonal and may be inverted locally. However, as nodal Langrange polynomials on Gauss–Lobatto points in conjunction with Gaussian quadrature are employed, the mass matrix is nondiagonal. A method has been developed in [161], which applies the inverse mass matrix by a similar sum factorization technique as the evaluation of the shape functions, and this inverse mass operator is employed throughout this Chapter.

3.5.2. Matrix-Free Solution of Linear Systems

The linear systems of the pressure Poisson matrix (Equation (3.39)), projection matrix (Equation (3.40)), and viscous matrix (Equation (3.41)) are all symmetric and solved by preconditioned conjugate gradient methods (CG). To this end, the traditional solution procedure of first computing the system matrix and then passing it to the solver, comes along with very high memory requirements if high polynomial degrees are used. In addition, such codes are usually memory-bound on modern computer architectures, meaning that the computation time required for the memory access limits the speed of the overall solution procedure. The approach chosen in this work is to evaluate all operators in a matrix-free manner by the computational kernels discussed above [156]. This technique both cuts the memory requirements and outperforms the classical solution methods using sparse matrices drastically [156].

For the respective equation systems, problem-tailored preconditioning techniques are described in the remainder of this section, summarizing the main ideas presented in [71] and [147].

Viscous Solver. The Helmholtz problem is solved using a preconditioner based on the inverse mass matrix similar to the method applied in [234] and [71]. This preconditioner is particularly effective in convection-dominated regimes or if the time step is small. Prior to solving the linear system of the viscous step, the solution is extrapolated from the previous time steps using the extrapolation formula already employed for the convective term:

$$\boldsymbol{U}_{\text{ext}}^{n+1} = \sum_{i=0}^{J-1} \beta_i \boldsymbol{U}^{n-i} \tag{3.43}$$

yielding the extrapolated velocity $\boldsymbol{U}_{\text{ext}}^{n+1}$, which is used as an initial guess for the iterative solver. This technique enables low relative solver tolerances while achieving high absolute accuracy. For turbulent eddy-resolving flow, solver iteration counts lie usually in the range of 3 to 5 to guarantee a relative accuracy of 10^{-4}.

In the case of the nonsymmetric face terms ($s = -1$) in Equation (3.18), the matrix is not symmetric, so a GMRES solver is applied using the same preconditioner.

Poisson Solver. The Poisson problem is solved employing the geometric multigrid algorithm by Kronbichler and Wall [162]. Therein, a single V-cycle with a polynomial Chebyshev smoother is used [3]. Besides the matrix diagonal, which is precomputed once during the initial setup, this preconditioner only requires the action of the Poisson operator, which is evaluated in a matrix-free manner using the fast sum factorization kernels. As a coarse-level solver, the Chebyshev iteration is used as well and the number of matrix-vector applications is chosen such that the usual Chebyshev error estimator reaches a tolerance of 10^{-3} [254] in the standard setting. The latter value is increased to 10^{-2} for the periodic hill problem, as this modification does not result in more solver iterations for this example while saving a few percent of computation time. In the geometric multigrid approach, the mesh is coarsened up to one single cell in the case of the turbulent channel flow and up to two cells in the periodic hill flow example in Chapter 4 using the mesh hierarchy management tools provided by the p4est library [37]. Analogous to the viscous solver, the pressure solution is extrapolated to the new time step according to

$$\boldsymbol{P}_{\text{ext}}^{n+1} = \sum_{i=0}^{J-1} \beta_i \boldsymbol{P}^{n-i}, \tag{3.44}$$

which is used as an initial guess for the Poisson solver, in order to yield high absolute accuracy despite of moderate relative solver tolerances of usually 10^{-4}. This setup yields approximately 9 to 12 solver iterations in turbulent eddy resolving flow with moderate mesh stretching, but may increase up to approximately 30 solver iterations in meshes with high aspect ratio. In the case of pure Dirichlet and periodic boundary conditions, pure Neumann boundary conditions are applied on the pressure Poisson problem, which renders the equation system under determined. This issue can be solved by a projection onto the subspace with zero mean [25], see also [147] for details.

Projection Solver. The local projection in the variants *VHW*, *V1*, and *V2* may be solved by simply applying the inverse mass operator. If a div-div or jump penalty term is included according to *V3*, an iterative solution procedure is applied. To this end, the div-div penalty projection is a local problem, as the cells are decoupled, so each cell can be solved independently and a local CG solver is considered. As a preconditioner, the inverse mass matrix is employed. This iterative solution procedure is approximately one to two orders of magnitude faster for computing the local projection with div-div penalty in comparison to a direct solver consisting of a matrix-based LU-factorization approach.

If the jump-penalty term is included in addition in *V4*, the cells are no longer decoupled and the projection becomes a global equation system, which is solved using a CG method preconditioned with an inverse mass operator as well.

3.5.3. Performance Evaluation

In order to benchmark the parallel scaling properties of the code, a strong scaling experiment is performed using the largest turbulent channel flow example considered in Section 4.2. Since *V3c* will be considered almost exclusively in the remainder of this work, the scaling experiments focus on this variant. The spatial discretization consists of a boundary-refined grid of 64^3 elements of fourth degree ($k = 4$), i.e., 33 million node points and 131 million DOFs overall (see Section 4.2 for further details). The highest aspect ratio of $\Delta x^+/\Delta y^+ = 12.3$ is reached at the cells closest to the boundary. Periodic boundary conditions are applied in streamwise and spanwise direction, respectively, and no-slip boundary conditions at the top and bottom walls. As an initial condition, a velocity profile of a polynomial of degree six is prescribed, which fulfills the boundary conditions. It approximately resembles the final mass flux and is heavily disturbed in

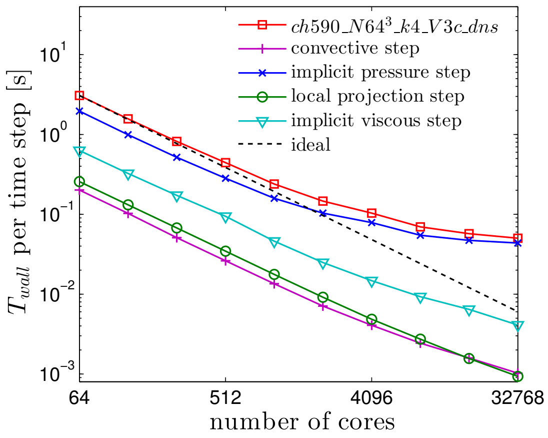


Figure 3.7.: Wall time T_{wall} per time step showing strong scaling of the finest turbulent channel flow DNS presented in Section 4.2 up to 32,768 CPU cores.

streamwise and spanwise direction in order to trigger transition to a fully turbulent flow. The timings are averaged over the first 500 time steps of the simulation since the number of solver iterations is approximately constant over the whole simulation time. The scaling experiments are performed on the SuperMUC Phase 1 system (9,216 nodes of dual-socket, eight-core Intel Sandy Bridge processors at 2.7 GHz each) in Garching, Germany. Figure 3.7 shows the results of the strong scaling tests from 64 and up to 32,768 processor cores for all substeps. Almost ideal scaling is achieved up to 2,048 cores and the scalability saturates at 32,768 cores and 0.05s wall clock time per time step. At 32,768 cores, each processor holds only 8 elements. Due to the vectorized processing of all element evaluations, four elements are evaluated simultaneously in double precision and eight elements in single precision within the pressure Poisson multigrid algorithm. Therefore, no substantial further speed-up of this example is expected beyond 32,768 cores, as all pressure multigrid levels are already latency-bound, except for the finest level.

In order to investigate the algorithm with regard to larger simulation scales and processor counts, the test case is simplified to laminar channel flow using the same boundary conditions at $\mathrm{Re}_\tau = 80$. A uniform mesh in a box of $2 \times 2 \times 2$ units and $\mathrm{Cr} = 1$ is employed. Figure 3.8 shows the results of weak and strong scaling experiments for variant *V3c* using the spatial polynomial degrees $k = 3$ and $k = 5$ up to 147,456 CPU cores again on Phase 1 of SuperMUC. The scaling of all solver components observed in Figure 3.8 is excellent. Further, a comparison of the weak scaling plots in Figure 3.8 reveals that the computation time per DOF and time step is essentially independent of the polynomial degree, making the use of high orders very attractive for example for laminar flows and the DNS of turbulent flows.

Regarding the weak scaling graphs in Figure 3.8, it is observed that the time spent in the local projection solver even decreases with increasing problem size. This behavior is due to the fact that the same flow is used in all computations and that the resolution of this flow increases substantially when going to large processor counts in the weak scaling. As a consequence of this high resolution, the velocity field is highly resolved and divergence-free, so a penalization of the divergence error is not necessary.

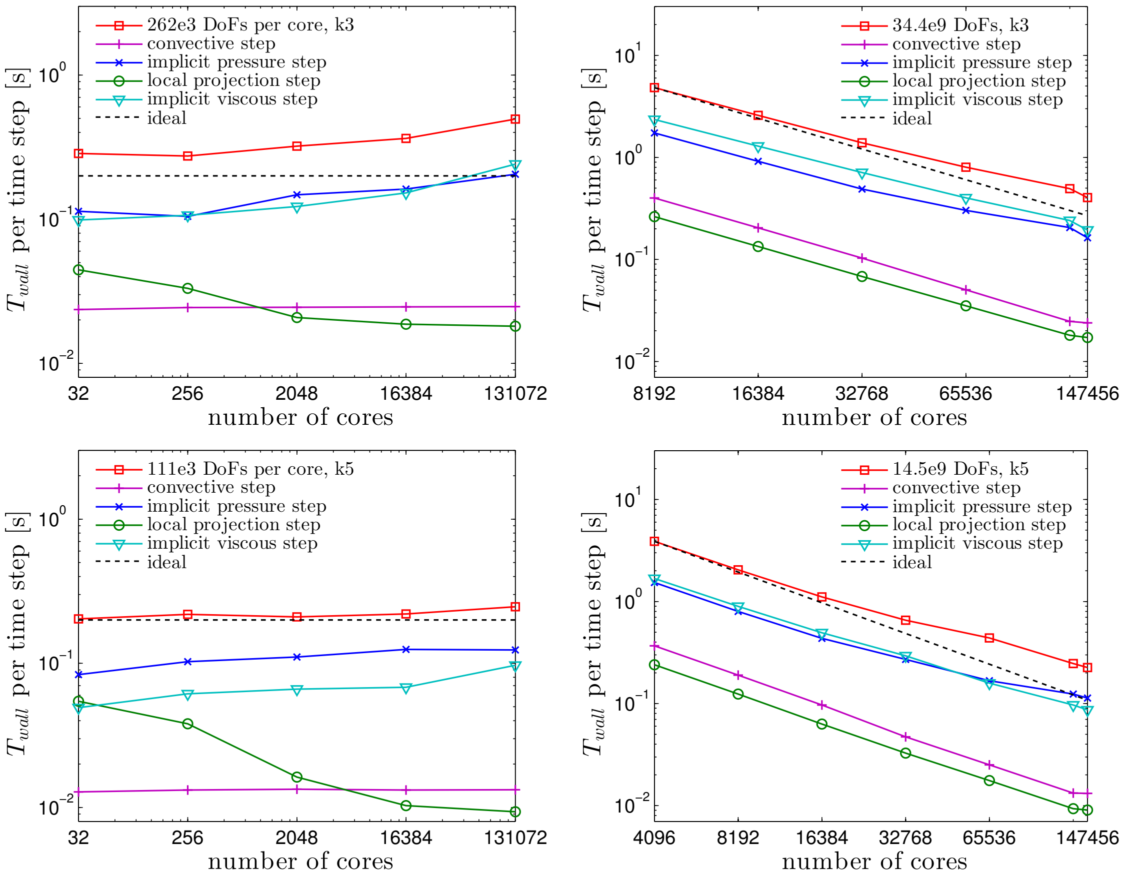


Figure 3.8.: Wall time T_{wall} per time step showing weak (left) and strong (right) scaling up to 147,456 CPU cores for third (top) and fifth (bottom) polynomial degree.

3.6. Verification

We verify the code described above and compare the two variants *V3c* and *V4* which emanated as the most promising ones from the discussion in Section 3.4 in the following. We commence in Section 3.6.1 by proving optimal convergence rates in time (Subsection 3.6.1.1) and space (Subsection 3.6.1.2) for velocity and pressure using the vortex problem already investigated in Section 3.4. We also discuss the efficiency of our code for laminar flows regarding the optimal use of high polynomial orders in Subsection 3.6.1.3 in the context of this example. A further test case is presented in Section 3.6.2 consisting of an unsteady laminar flow past a cylinder that demonstrates the geometrical flexibility of the present approach. Several additional verification examples are shown in [71]. While the examples in this section employ the 2D implementation of the code, we discuss application to large-scale simulations of 3D turbulent channel flow in Chapter 4.

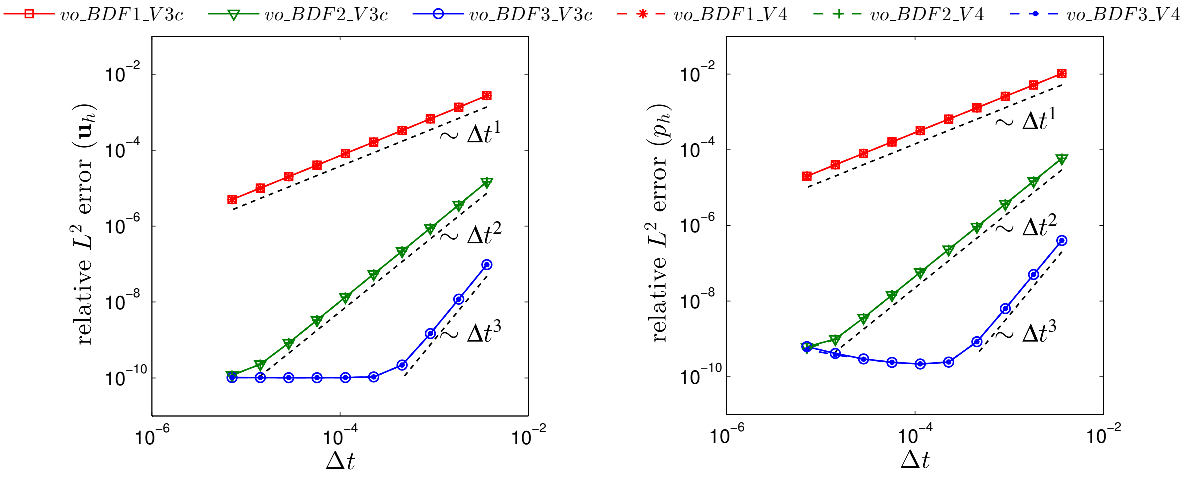


Figure 3.9.: Temporal convergence of *V3c* and *V4* using $J = J_p = 1$ ($BDF1$), $J = J_p = 2$ ($BDF2$) and $J = J_p = 3$ ($BDF3$).

3.6.1. Vortex Problem

Let us consider a laminar vortex problem according to [112] as implemented in [71] with the analytical solution for velocity and pressure given as

$$\boldsymbol{u}(\boldsymbol{x},t) = \begin{pmatrix} -\sin(2\pi x_2) \\ +\sin(2\pi x_1) \end{pmatrix} \exp\left(-4\nu\pi^2 t\right) \text{ and} \tag{3.45}$$

$$p(\boldsymbol{x},t) = -\cos(2\pi x_1)\cos(2\pi x_2)\exp\left(-8\nu\pi^2 t\right), \tag{3.46}$$

defined in the domain $[-0.5, 0.5] \times [-0.5, 0.5]$ with respective Dirichlet boundary conditions on the inflow and exact Neumann boundary conditions given as g_p and $\boldsymbol{h}$ on the outflow boundaries (see [112] for details). We choose a viscosity of $\nu = 0.025$, the simulation time as $\mathcal{T} = 1$ and define the Courant number according to Equation (3.21) with the maximum velocity $U = 1.4$. The relative L^2 error is computed at $t = \mathcal{T}$ and is defined as

$$\frac{\| \boldsymbol{u}(\boldsymbol{x}, t = \mathcal{T}) - \boldsymbol{u}_h(\boldsymbol{x}, t = \mathcal{T})\|_{\Omega_h}}{\|\boldsymbol{u}(\boldsymbol{x}, t = \mathcal{T})\|_{\Omega_h}} \tag{3.47}$$

for the velocity and

$$\frac{\| p(\boldsymbol{x}, t = \mathcal{T}) - p_h(\boldsymbol{x}, t = \mathcal{T})\|_{\Omega_h}}{\|p(\boldsymbol{x}, t = \mathcal{T})\|_{\Omega_h}} \tag{3.48}$$

for the pressure.

3.6.1.1. Temporal Convergence

Simulations are performed for this example in order to investigate the convergence for the temporal orders of accuracy $J = J_p = \{1, 2, 3\}$ in addition to the computations shown in Section 3.4

using $J = J_p = 1$. The time step is chosen using criterion (3.21) starting from Cr $= 2$ by successive bisection down to Cr $= 0.0039$ for all cases. We have observed that the present scheme allows computations with Courant numbers beyond the common limit of Cr $= 1$ for this flow and that there is no major difference in stability limits comparing the second and third order method, which is in contrast to the conditional stability for $J = 3$ reported in [177], since the flow operates near the limit of viscous dominance. The spatial discretization uses $N_e = 8^2$ elements and a polynomial degree of $k = 7$. The results presented in Figure 3.9 are labeled accordingly through the temporal scheme BDF1 for $J = J_p = 1$, BDF2 for $J = J_p = 2$, BDF3 for $J = J_p = 3$ as well as the respective variant *V3c* and *V4*.

The results in Figure 3.9 show optimal convergence rates for the respective order until the spatial error becomes predominant. There is essentially no difference between the variants *V3c* and *V4*. Please note that earlier studies frequently present the current splitting scheme with mixed temporal orders, e.g., $J = 2$ and $J_p = 1$, for which the temporal accuracy according to [106, 177] is at most second order for the velocity and order $3/2$ for the pressure. By choosing $J = J_p$ we get optimal convergence rates for both velocity and pressure at virtually no additional computational cost.

3.6.1.2. Spatial Convergence

We proceed with an investigation of the spatial convergence using the same example. Grid refinement studies are performed for the polynomial degrees $k = \{1, 2, 3, 4, 5, 6, 7\}$ considering both *V3c* and *V4* with the time step chosen according to the CFL condition (3.21) as Cr $= 0.0625$ and $J = J_p = 3$. This way, the spatial error is dominant.

It may be observed in Figure 3.10 that optimal convergence rates of order $k+1$ are obtained for the velocity and pressure in all cases. The error is not distinguishable between the two variants *V3c* and *V4*.

3.6.1.3. Performance Evaluation

Which polynomial degree yields the most efficient algorithm for laminar flows? In order to provide a first answer to this question, Figure 3.11 shows the error of the vortex problem according to Figure 3.10 for $k \geq 1$ over the wall time of the simulations T_{wall}. All simulations have been conducted in serial on the same computational setup, meaning that the wall time may be interpreted as the computational cost. The graph illustrates that increasing polynomial orders result in a steeper slope, i.e., the error decreases more rapidly if larger computational effort is invested. The expected slope of these curves of order $\mathcal{O}(T_{wall}^{-(k+1)/(d+1)})$ (with $d = 2$ for this example) is obtained by considering a decrease of the error with h^{k+1} and an increase in computation time with $T_{wall} \sim h^{-(d+1)}$ due to the number of elements to be evaluated on the one hand as well as the CFL condition on the other hand. The curves depicted in Figure 3.11 show very good agreement with this slope starting from the fourth refinement level, which confirms the optimality of the code for high refinement levels. The cause of the discrepancy regarding the coarser meshes is the efficiency of the particular multigrid algorithm, which results in decreased performance for a small number of refinement levels compared to finer meshes [147].

We conclude from this investigation that high-order methods are very efficient if high precision is required while lower polynomial degrees may be advantageous if a fast time to solution

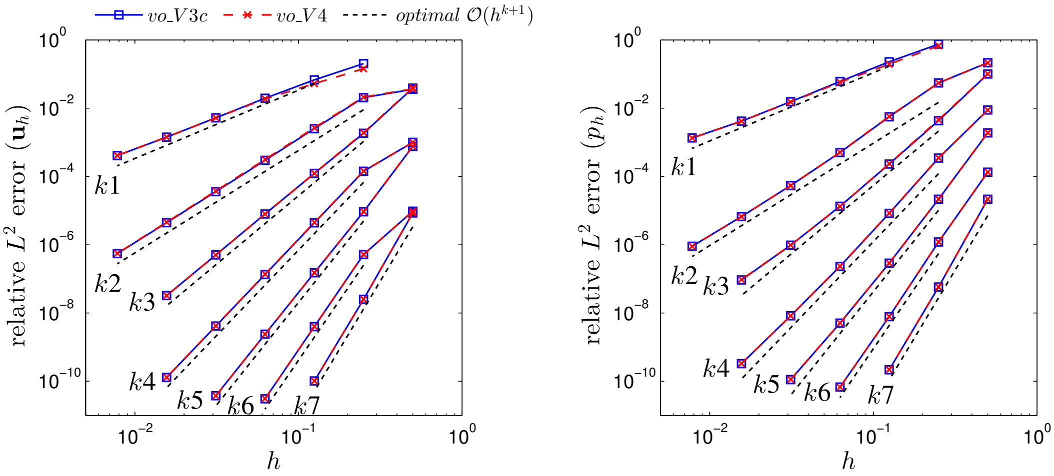


Figure 3.10.: Spatial convergence for several polynomial degrees $k\{1,2,3,4,5,6,7\}$.

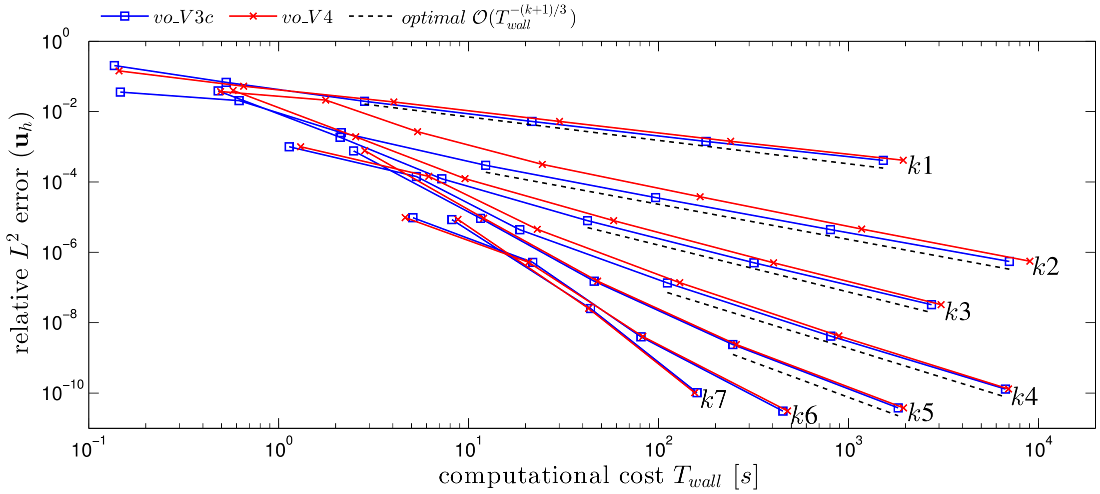


Figure 3.11.: Computational cost in terms of the wall time of a serial computation T_{wall} for the vortex problem using several polynomial orders $k\{2,3,4,5,6,7\}$.

at reduced accuracy is desired. Please note that this discussion only presents a rough estimation since we do not tune the Courant number or the relative error tolerances of the linear solvers for these cases, which would result in lower computation times for moderate precision and low polynomial degrees. Moreover, these conclusions might change in the advection-dominated case, which is dominated by cumulative dispersion errors. Considering high accuracies it is yet the slope that is more relevant than the absolute wall time. Further, computation times of variant *V4* are slightly elevated in comparison to *V3c* due to the global character of the projection equation system.

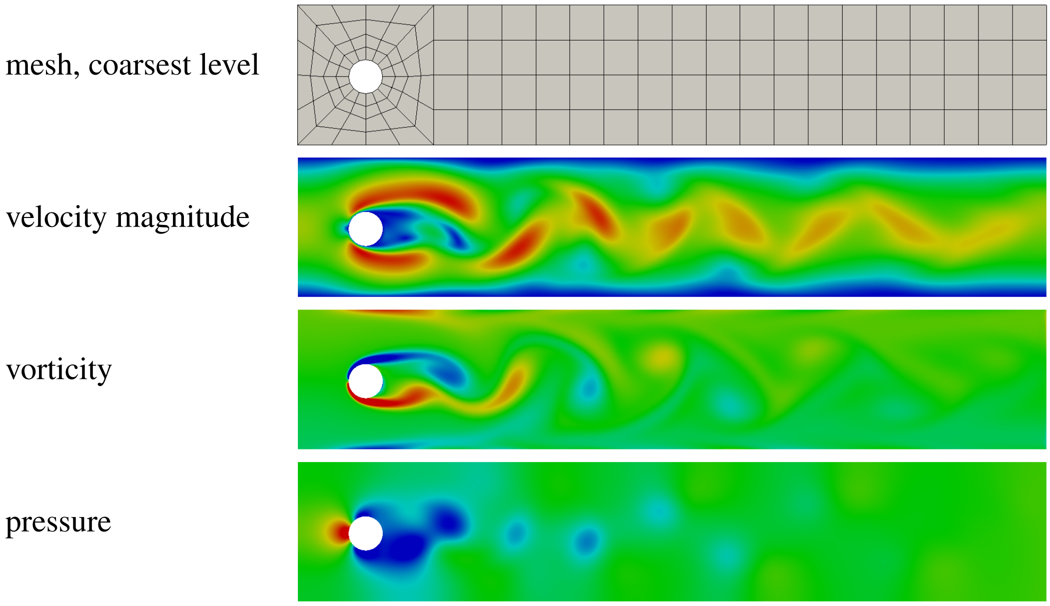


Figure 3.12.: Flow past cylinder from top to bottom: coarsest mesh level, velocity magnitude, vorticity and pressure snapshot at time $t = 5.5$. The latter three have been computed with the finest mesh including $k = 6$ and *V3c*. Red indicates high and blue low values.

3.6.2. Laminar Flow past Cylinder

As a second laminar benchmark we investigate unsteady vortex shedding in the wake of a cylinder presented as test case 2D-3 in [225] with accurate reference data provided in [128] considering the implementation of this flow in INDEXA according to [71]. The domain is of dimensions $W \times H$ in streamwise and vertical direction, respectively, with $W = 2.2$ and $H = 0.41$. The cylinder is of diameter 0.1 and its center point is located at 0.2 units from the inflow as well as the bottom walls. At the inflow boundary, the velocity is prescribed by

$$g_{u1}(x_2, t) = U \frac{4x_2(H - x_2)}{H^2} \sin(\pi t / \mathcal{T}) \tag{3.49}$$

with $U = 1.5$ and the simulation time $\mathcal{T} = 8$. At the top and bottom wall as well as the cylinder surface, no-slip boundary conditions are applied. At the outlet, zero pressure boundary conditions are applied with $g_p = 0$ and $\boldsymbol{h} = \boldsymbol{0}$. The resulting flow exhibits unsteady vortex shedding behind the cylinder as illustrated in Figure 3.12 at the time instant $t = 5.5$.

We perform simulations for the spatial polynomial degrees $k = \{4, 5, 6, 7\}$ and present three levels of refinement for each polynomial degree. The approximation of the cylindrical geometry is enhanced by mapping the boundary nodes onto the cylinder surface using an iso-parametric mapping through facilities provided in the deal.II library [9], which yields an accurate representation of the geometry for high-order polynomials. A similar mapping technique based on commercial preprocessor tools has been presented in [114]. The resulting mesh of the coarsest level is displayed in Figure 3.12. The third-order accurate time integration scheme is chosen with $J = J_p = 3$ and a Courant number of $\mathrm{Cr} = 0.25$ based on U and the minimum edge length of

Table 3.4.: Flow past cylinder cases, resolutions and results: Number of grid points N, maximum drag coefficient $c_{D\,\mathrm{max}}$, maximum lift coefficient $c_{L\,\mathrm{max}}$, and pressure difference between windward and lee side of cylinder at end of simulation Δp_{end}.

		V3c			*V4*		
k	N	$c_{D\,\mathrm{max}}$	$c_{L\,\mathrm{max}}$	Δp_{end}	$c_{D\,\mathrm{max}}$	$c_{L\,\mathrm{max}}$	Δp_{end}
4	3,400	2.767868	0.437474	-0.108995	2.767668	0.414535	-0.108205
	13,600	2.963629	0.487026	-0.111627	2.969142	0.488884	-0.111537
	54,400	2.950792	0.478365	-0.111654	2.950868	0.478380	-0.111638
5	4,896	2.958448	0.512599	-0.109378	2.965825	0.512767	-0.109407
	19,584	2.951065	0.480343	-0.111616	2.951639	0.480529	-0.111599
	78,336	2.950454	0.477967	-0.111618	2.950385	0.477948	-0.111618
6	6,664	2.964686	0.504783	-0.111059	2.968793	0.507933	-0.110952
	26,656	2.949137	0.478317	-0.111614	2.948920	0.478338	-0.111617
	106,624	2.950829	0.477940	-0.111615	2.950826	0.477921	-0.111615
7	8,704	2.949849	0.487586	-0.111414	2.949785	0.487317	-0.111382
	34,816	2.950198	0.477948	-0.111615	2.950064	0.477944	-0.111614
	139,264	2.950927	0.477941	-0.111615	2.950922	0.477920	-0.111616
2	∼200,000 [128]	2.95092	0.47795	-0.1116	2.95092	0.47795	-0.1116
lower bound [225]		2.9300	0.4700	-0.1150	2.9300	0.4700	-0.1150
upper bound [225]		2.9700	0.4900	-0.1050	2.9700	0.4900	-0.1050

the respective mesh h_{min}, and we use the CFL condition with constant time stepping according to Equation (3.21). The relative solver tolerances are chosen as 10^{-6} for the Poisson and Helmholtz solver and 10^{-9} for the projection solver. The solution quality is evaluated according to [225] by the maximum value of the drag and lift coefficients $c_{D\,\mathrm{max}}$ and $c_{L\,\mathrm{max}}$ over time as well as the pressure difference between the windward and the lee side of the cylinder at the end of the simulation, denoted $\Delta p_{\mathrm{end}} = \Delta p(t = \mathcal{T})$. A detailed description of how these quantities are computed is given in [128, 225].

The results of the present simulations are displayed in Table 3.4 along with reference data by [128], where $c_{D\,\mathrm{max}}$ is of absolute accuracy $5 \cdot 10^{-7}$ and $c_{L\,\mathrm{max}}$ as well as Δp_{end} are of accuracy 10^{-4}. In addition, upper and lower bounds for all three quantities as presented in [225] are also included.

The results exhibit excellent agreement with the reference data. Especially the pressure difference is already predicted with the same accuracy as the reference data for 13,600 grid points with *V3c* and reaches two additional digits in precision during refinement. The drag coefficient also converges to the reference data yielding an accuracy of at least five digits and an excellent agreement between *V3c* and *V4*. The lift coefficient converges to 0.4779 using *V3c* and *V4* which is in excellent agreement with reference value given as 0.47795, where the error was specified to be no larger than 10^{-4}.

3.7. Summary

In this chapter, we have developed a stable, accurate and efficient numerical scheme for the simulation of the incompressible Navier–Stokes equations by reviewing, comparing and extending stabilization techniques proposed in the literature. The best stabilization for small time steps is based on a div-div penalty approach that enhances the point-wise divergence-free condition within elements. Underresolved flows have been stabilized by the partial integration of the right-hand side of the Poisson equation and optionally a supplementary jump-penalty term within the projection. The resulting algorithm exhibits convergence orders equal to polynomial degree plus one in space and three in time both in velocity and pressure and is embedded in a matrix-free implementation.

The high efficiency of this implementation for high polynomial degrees was demonstrated by two laminar flow examples present in a vortex problem and flow past a cylinder. This characteristic makes the present methodology also very attractive for an application to DNS and LES of turbulent flows, which is the topic of the next chapter.

Finally, the solver presented in this chapter paves the way for several extensions as part of this thesis: In Chapter 5, an extension of the scheme towards the RANS equations using the SA model is presented. This method is also applied to DES. The primary topic of this work, wall modeling via function enrichment, is also implemented as an extension of this solver in Chapters 8 to 10.

4

Application to DNS and LES of Turbulent Flow

We demonstrate the applicability of the code presented in the previous chapter to DNS and LES of turbulent flow. DNS is the most accurate approach to computing turbulent flows, since all turbulent scales are resolved by the numerical method, see Section 2.3.1. As discussed in the first subsection, the current DG method is very attractive for computing DNS due to its high-order capability, in addition to its speed and high scalability on massively parallel computers. DNS is yet out of reach in the foreseeable future regarding most industrial applications in the moderate to large Reynolds number regime, which is why we show that the present approach is also very well suitable for implicit LES.

Two benchmark flows are considered, turbulent channel flow (Section 4.2) and flow over periodic hills (Section 4.3), which cover all relevant flow phenomena. Turbulent channel flow provides insight into the performance of the numerical scheme in attached boundary layers, while the periodic hill flow shows a number of complex flow conditions such as flow separation and high pressure gradients. Both configurations will be extensively used in the remainder of this thesis to assess CFD models and discretization schemes. Due to the lack of accurate reference data in the literature for the periodic hill flow, we make use of the DNS capabilities of the method and compute this flow with very high resolution in order to provide a new set of reference data at two Reynolds numbers. The research presented in this chapter was previously published in Krank et al. [147, 149].

4.1. DNS and Implicit LES using High-Order DG Methods

We comment on the characteristics of the present DG approach regarding DNS and the potential of ILES for underresolved computations of turbulent flow.

4.1.1. Resolution Power for DNS

DG codes have been used for DNS of several flows, for example in [8, 15, 115, 147, 149, 178, 263] and of a weakly turbulent flow in [54]. It is frequently argued that high-order (spectral) DG methods are particularly suited for DNS, and these arguments are detailed in the following. The basic characteristics of high-order DG methods have recently been investigated via dispersion-diffusion analysis, for example by Gassner and Kopriva [96] and Moura et al. [188, 189], with regard to their applicability to turbulent flows. Although the latter study investigates full upwinding DG methods and the local Lax–Friedrichs flux may be less dissipative, the key findings are still relevant for the present numerical method:

- The smallest resolvable scales, i.e., the grid filter size, may be approximated as $h_e/(k+1)$ with a characteristic cell size h_e.

- Higher order DG schemes exhibit much lower dispersion and dissipation errors than lower order methods and maintain this characteristic over a broader wave number range. Thus, higher order methods can resolve turbulent scales until close to the filter size.
- Scales just marginally larger than the filter size are damped more significantly with increasing degree.

Based on the dispersion-diffusion analysis, [188, 189] introduced a measure for the efficiency of a numerical scheme, the 1% rule, which specifies the minimum number of grid points per wave length in which the amplitude of a scale is dissipated by less than 1% while being convected by a distance of $h_e/(k+1)$. As an example, a second-order DG scheme ($k=1$) would require 8.2 nodes per wave length (in 3D [188]), whereas 3.4 nodes would suffice for a scheme of 6th order accuracy, 3.2 nodes for a scheme of 7th order accuracy, and 3.1 nodes for a scheme of 8th order accuracy. This analysis allows the conclusion that a DG scheme introduces almost zero numerical dissipation if all scales are larger than the limit defined by the 1% rule, and that high polynomial orders are much more efficient in order to achieve this goal. In practice, there are several other techniques required, which make the use of such high degrees possible. In particular, a matrix-free solution procedure of linear systems is necessary, as the computation of system matrices would be very expensive and sparse matrix-based linear solvers are slow for high polynomial degrees (see Section 3.5). In addition, the number of solver iterations of the iterative solver should be constant with respect to the polynomial degree, which is the case for the solvers used herein. The limiting factors regarding high polynomials are the decreasing time step size due to the CFL condition as $\sim k^{-1.5}$ (see Section 3.3.3), the increase in solution time of the linear system as a result of an increased communication cost in parallel, and a saturation of the resolution power for even higher polynomial degrees. We argue therefore, that the polynomial degrees $k=5$ to 7 are particularly efficient for DNS using the present method. As an alternative, fully explicit codes may be used for the compressible Navier–Stokes equations [115], which do not require the solution of linear systems, but their time step is additionally restricted by the viscous term and the speed of sound in nearly incompressible schemes, see Section 3.1.1.

4.1.2. Application to LES

The high-order accuracy present in DG methods is also efficient for LES. The low dissipation and dispersion errors allow a resolution of turbulent scales with relatively few DOFs, such that coarser meshes may be used in comparison to low-order methods. The low dispersion errors are particularly relevant for LES, since eddies close to the resolution limit can be convected over large distances with high accuracy. A typical application for such a requirement would be a formula one car, where vortical structures are generated by the front wing and the front wheels, which interact with other parts of the car at a downstream location. High-order methods would provide a higher accuracy for the intermediate distance between these two areas of interest. Furthermore, a faster damping of the poorly-resolved scales quickly removes them from the velocity field, preventing “them from polluting the numerical solution” [189].

This numerical dissipation is also the basis for the implicit LES (ILES)-paradigm. In ILES, the idea is that the numerical dissipation necessary to stabilize the scheme in underresolved flows is in fact appropriate to model the unresolved subgrid scales. This method will be assessed with the present numerical method in the remainder of this chapter. In addition, high-order DG

Table 4.1.: Comparison of y^+ criteria for the assessment of near-wall resolution.

degree k	$1/k$	$1/(k+1)$	$\Delta y_{1,\mathrm{GL}}/\Delta y_e$
1	1	0.5	1
2	0.5	0.3333	0.5
3	0.3333	0.25	0.2764
4	0.25	0.2	0.1727
5	0.2	0.1667	0.1175
6	0.1667	0.1429	0.0849
7	0.1429	0.125	0.0641

schemes have been extensively used for ILES, see for example [15, 19, 20, 28, 62, 147, 149, 178, 262, 264], and [117] for a rigorous derivation in the variational context. While ILES is used for most LES-type simulations in this thesis, an eddy-viscosity subgrid model will be considered in the context of DES in Chapters 5 and 9 and other eddy-viscosity subgrid models have been investigated in conjunction with high-order DG in the literature, e.g., the WALE model in [178].

4.1.3. Criteria for Assessing the Near-Wall Resolution

With regard to the assessment of resolution, we discuss one particular aspect in detail, which is relevant for all turbulent wall-bounded simulations presented herein. Near no-slip boundaries, a requirement is in all numerical schemes that the first grid point should be located near $y_1^+ \sim 1$ in order to capture the velocity gradient in the laminar sublayer. Regarding higher order elements, several interpretations of this requirement are possible. The above discussion on the resolution power suggested that the smallest resolvable scales may be approximated by $\Delta y_e/(k+1)$ with the wall-normal element size Δy_e. As the boundary nodes of two neighboring cells lie at the same location, one may also consider the factor of $\Delta y_e/k$ appropriate; see, e.g., [267]. Going back to the original requirement that the first off-wall node should be considered, the best consistent way of applying the y^+ criterion would be to determine the location of the first off-wall node inside the cells, $\Delta y_{1,\mathrm{GL}}$, which is given through the Gauss–Lobatto nodes in this case. These criteria are compared in Table 4.1 for the polynomial degrees one to seven. It is noted that the criterion using the Gauss–Lobatto points yields a strong variation with the polynomial degree, which mirrors the higher resolution power of higher order polynomials. This criterion is considered as best comparable to the classical requirement of $y_1^+ \sim 1$ and is therefore primarily used in the remainder of this work. As an alternative, one may also determine the width of an entire cell required to capture the near-wall gradient and the turbulent motions through numerical experiments, and apply that criterion. Regarding the assessment of the wall-parallel solution, the factor $1/(k+1)$ seems to be most relevant, as it is in line with the findings within the dispersion-diffusion analysis cited above.

4.2. Turbulent Channel Flow

We begin with the presentation of the DNS results in Section 4.2.1 and discuss the application to ILES in Section 4.2.2.

Table 4.2.: Channel flow cases and resolutions. Specification of the first off-wall node through $y^+_{1,\mathrm{GL}} = y_{1,\mathrm{GL}} u_\tau/\nu$ and all other mesh quantities $\Delta(\cdot)^+ = \Delta(\cdot)_e u_\tau/\nu(k+1)$ is given as the respective element length $\Delta(\cdot)_e$ divided by the number of nodes in each spatial direction per element $k+1$ where k is the polynomial degree. Δx^+: resolution in x_1-direction; $\Delta y^+_{1,\mathrm{GL}}$: first off-wall point in x_2-direction; Δy^+_c: resolution at center in x_2-direction; Δz^+: resolution in x_3-direction.

Case	N_e^3	k	N	Cr	Re_τ	γ	Δx^+	$\Delta y^+_{1,\mathrm{GL}}$	Δy^+_c	Δz^+
$ch180\_N32^3\_k5\_V3c\_dns$	32^3	5	7.1M	1	180	1.4	11.8	0.76	3.0	3.9
$ch590\_N64^3\_k4\_V3c\_dns$	64^3	4	32.8M	0.8	590	1.65	11.6	0.81	6.5	5.8
$ch180\_N8^3\_k3\_V3c\_les$	8^3	3	0.03M	1	180	1.8	35.3	3.9	20.1	17.7
$ch180\_N16^3\_k3\_V\{3c,4\}\_les$	16^3	3	0.26M	1	180	1.8	17.7	1.5	10.5	8.8
$ch180\_N32^3\_k3\_V3c\_les$	32^3	3	2.1M	1	180	1.8	8.8	0.69	5.3	4.4
$ch590\_N16^3\_k4\_V\{3c,4\}\_les$	16^3	4	0.51M	1	590	2.25	46.3	1.7	33.1	23.2
$ch590\_N32^3\_k4\_V3c\_les$	32^3	4	4.1M	1	590	2.25	23.2	0.73	16.9	11.6

4.2.1. Direct Numerical Simulation

There is a vast number of publications in the field of DNS of turbulent channel flow providing accurate reference data for turbulence modeling research as well as the validation of numerical schemes, see for example Moser et al. [187] for friction Reynolds numbers $\mathrm{Re}_\tau = 180$, 395, and 590 and an overview of publications on DNS of the case $\mathrm{Re}_\tau = 180$ by Vreman [256]. Herein, Re_τ is defined as $\mathrm{Re}_\tau = u_\tau\delta/\nu$ with the given channel-half height δ and friction velocity $u_\tau = \sqrt{\tau_w/\rho}$, where τ_w is the wall shear stress. In the present work, we perform DNS of turbulent channel flow at $\mathrm{Re}_\tau = 180$ and 590. Computational domain sizes for these flows are specified in [187] as $4\pi\delta\times 2\delta\times\frac{4}{3}\pi\delta$ in streamwise, wall-normal, and spanwise direction, respectively, for the case $\mathrm{Re}_\tau = 180$ and $2\pi\delta\times 2\delta\times\pi\delta$ for the case $\mathrm{Re}_\tau = 590$ accordingly. Periodic boundary conditions are considered in the streamwise and spanwise directions and no-slip boundary conditions are imposed at the walls. The mesh is graded towards the no-slip boundaries to improve the resolution of near-wall turbulent structures according to the hyperbolic mesh mapping given as $f\colon [0,1] \to [-\delta,\delta]$:

$$x_2 \mapsto f(x_2) = \delta\frac{\tanh(\gamma(2x_2-1))}{\tanh(\gamma)} \tag{4.1}$$

using the mesh-stretching parameter γ according to Table 4.2. The spatial discretizations employed are similar in resolution as the simulations presented in [187] and are listed in Table 4.2. For the case $\mathrm{Re}_\tau = 180$, 32×32×32 elements of degree 5 are used, resulting in 7.1 million nodes and 28.3 million DOFs overall, which is slightly finer than in [187] with 128×129×128 Fourier modes in the periodic directions as well as Chebyshev nodes in the wall-normal direction and 2.1 million nodes. Regarding $\mathrm{Re}_\tau = 590$, a mesh of 64×64×64 elements of degree 4 with 32.8 million nodes (131 million DOFs) is used again compared to a mesh of 384×257×384 (37.9 million nodes) in [187]. The time step of the BDF3 scheme ($J = J_p = 3$) is chosen based on the CFL condition using constant time stepping (Equation (3.21)) according to $\Delta t = \mathrm{Cr}\; h_{\min}/Uk^2$ where we take $\mathrm{Cr} = 1$ for $\mathrm{Re}_\tau = 180$ and $\mathrm{Cr} = 0.8$ for $\mathrm{Re}_\tau = 590$, $h_{\min}$ as the minimum edge length and $U = 15u_\tau$ representing the estimated maximum velocity occurring in the cells

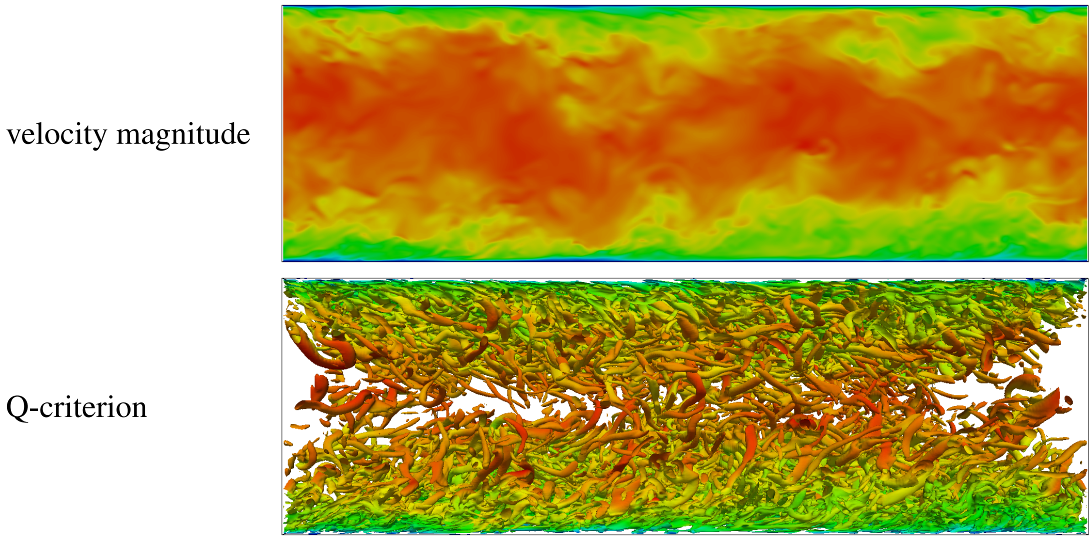


Figure 4.1.: DNS of turbulent channel flow at $\mathrm{Re}_\tau = 590$: Contour of velocity magnitude (top) and eddies visualized via the Q-criterion, colored by velocity magnitude (bottom). High velocity is colored red and low velocity blue.

with the shortest edge length, which are located at the no-slip boundaries. Statistics are sampled spatially over homogeneous planes and temporally over approximately 30 flow-through times based on the mean center-line velocity for the case $\mathrm{Re}_\tau = 180$ and 68 flow-through times for $\mathrm{Re}_\tau = 590$. The nominal simulation parameters are chosen as $u_\tau = \tau_w = \rho = \delta = 1$, resulting in $\nu = 1/\mathrm{Re}_\tau$ for the viscosity, and the results are normalized with the numerical value of τ_w. The flow is driven by a body force, which may be determined through equilibrium of forces to $f_1 = 1$. Our numerical investigations have shown that relative solver tolerances of 10^{-4} for the Poisson solver, 10^{-6} for the local projection solver, and 10^{-4} for the Helmholtz solver are sufficient and may even be relaxed by one magnitude regarding the Poisson solver. For the DNS computations, we solely consider the variant *V3c*.

The resulting turbulent flow is depicted in Figure 4.1 for the case $\mathrm{Re}_\tau = 590$ via velocity contours as well as the Q-criterion for eddy visualization of one snapshot. The cases are labeled according to Table 4.2 and the results are plotted over the wall coordinate $y^+ = y u_\tau/\nu$ and x_2/δ, respectively, in Figure 4.2. Herein, the normalized mean velocity is defined as $u^+ = u_1/u_\tau$ and fluctuations in form of the root-mean-square (RMS) velocities as $u'^+ = \mathrm{RMS}(u_1)/u_\tau$, $v'^+ = \mathrm{RMS}(u_2)/u_\tau$, and $w'^+ = \mathrm{RMS}(u_3)/u_\tau$ as well as the Reynolds shear stresses (RSS) as $(u'v')^+ = (u_1 u_2)/u_\tau^2$. The curves exhibit excellent agreement with the reference data from [187] labeled DNS_MKM99. The marginal underprediction of the mean velocity in the center of the channel may possibly be due to a too coarse mesh in that region.

4.2.2. Implicit Large-Eddy Simulation

We consider the same benchmark example for the assessment of the scheme regarding ILES. We employ a setup similar to the previous subsection with a domain size of $2\pi\delta \times 2\delta \times \pi\delta$ for all computations. All cases considered are listed in Table 4.2 and include three refinement levels for

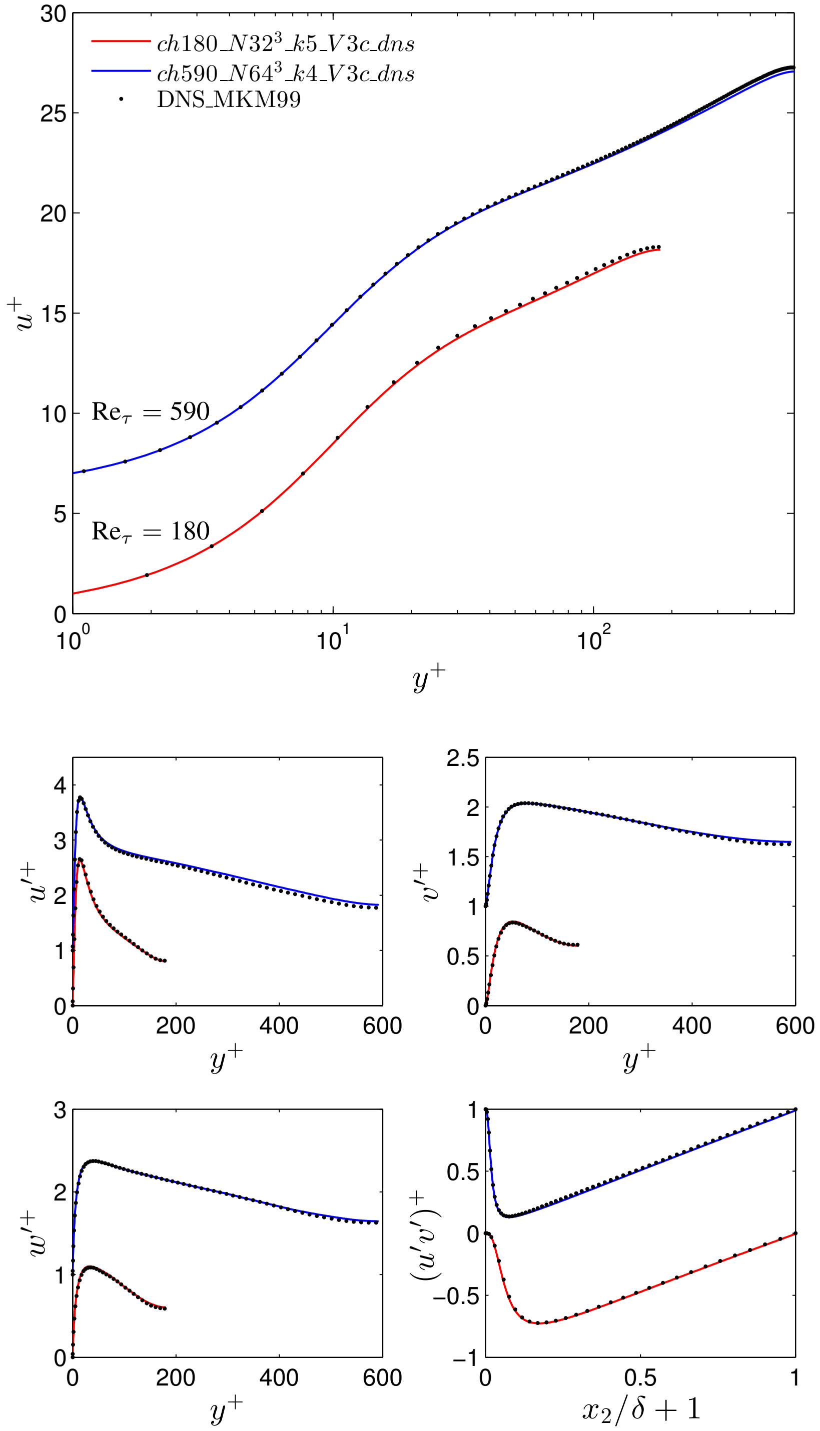


Figure 4.2.: DNS of turbulent channel flow at $\mathrm{Re}_\tau = 180$ and 590: Mean velocity $u^+ = u_1/u_\tau$ (top) and RMS velocities $u'^+ = \mathrm{RMS}(u_1)/u_\tau$, $v'^+ = \mathrm{RMS}(u_2)/u_\tau$, and $w'^+ = \mathrm{RMS}(u_3)/u_\tau$ as well as RSS $(u'v')^+ = (u_1u_2)/u_\tau^2$ (bottom). For the case $\mathrm{Re}_\tau = 590$, the mean velocity is shifted upwards by six units and all other quantities by one unit for clarity.

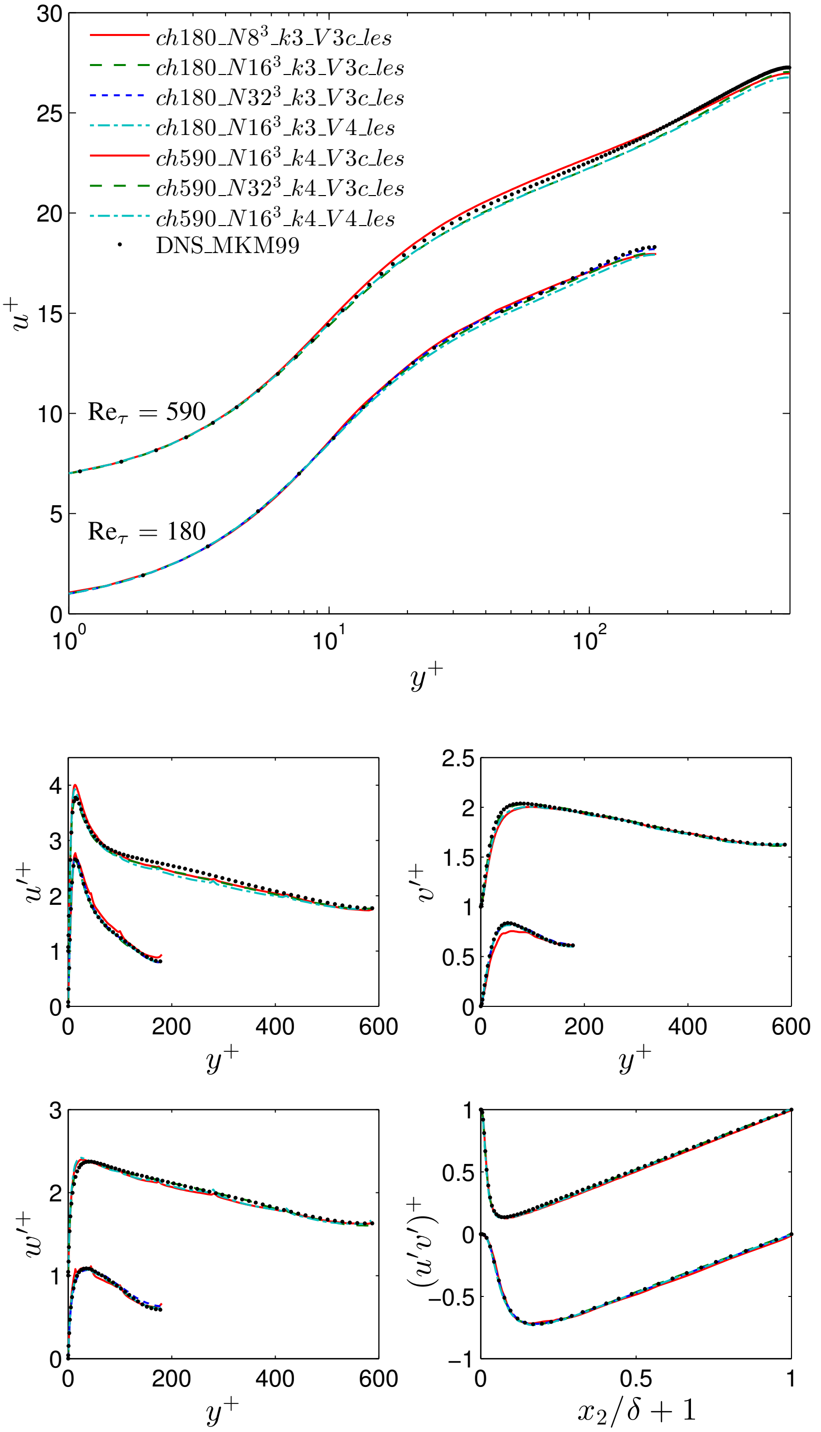


Figure 4.3.: ILES of turbulent channel flow at $\mathrm{Re}_\tau = 180$ and 590: Mean velocity $u^+ = u_1/u_\tau$ (top) and RMS velocities $u'^+ = \mathrm{RMS}(u_1)/u_\tau$, $v'^+ = \mathrm{RMS}(u_2)/u_\tau$, and $w'^+ = \mathrm{RMS}(u_3)/u_\tau$ as well as RSS $(u'v')^+ = (u_1u_2)/u_\tau^2$ (bottom). For the case $\mathrm{Re}_\tau = 590$, the mean velocity is shifted upwards by six units and all other quantities by one unit for clarity.

the case $\mathrm{Re}_\tau = 180$ and two refinement levels for $\mathrm{Re}_\tau = 590$. All cases are computed with the variant *V3c* and two representative cases are shown employing variant *V4* (see Section 3.4.2), which are labeled accordingly.

The results of all ILES cases listed in Table 4.2 are depicted in Figure 4.3 along with the DNS data from [187]. Regarding variant *V3c*, an excellent agreement with the reference solutions is observed and the coarsening of the discretizations has very little effect on the solution quality. The mean velocity of the cases *V4* is very similar to variant *V3c*.

A comparison of computation times yields an elevated computational cost of the case $ch180\_N16^3\_k3\_V4\_les$ by a multiplicative factor of 2.2 in comparison to $ch180\_N16^3\_k3\_V3c\_les$ using the same computational setup. Analogously, the case $ch590\_N16^3\_k4\_V4\_les$ also completes in 2.2 times the computational cost of $ch590\_N16^3\_k4\_V3c\_les$. These differences are largely due to the global equation system introduced in the projection step in *V4*. It is therefore concluded from these investigations that *V3c* is the most efficient variant discussed in the present work for simulation of turbulent flows, both for DNS and ILES, as it combines high accuracy with short computation times. Therefore, this variant is considered in the remainder of this thesis.

4.3. Flow over Periodic Hills

We proceed with an application of the present solver to DNS and ILES of flow over periodic hills according to, e.g., Temmerman et al. [249] and Fröhlich et al. [93]. This example has in recent years become one of the most widely used test cases for the validation and assessment of CFD codes and turbulence models. Its popularity lies in its simplicity regarding simulation setup and boundary conditions on the one hand and complexity with respect to flow phenomena and turbulence modeling on the other hand. The flow configuration consists of streamwise periodically arranged smoothly curved hills at the lower wall, a plane boundary at the upper wall, and periodic boundary conditions in spanwise direction, so the setup is quite similar to the turbulent channel flow presented in the previous section. A major advantage of this setup is the periodicity in streamwise direction, which does not require the definition of synthetic inflow conditions, and therefore simplifies the reproducibility of the results. The developed turbulent flow is depicted in Figure 4.4 and shows several challenging and highly relevant flow conditions, which make this example particularly interesting as a benchmark case, such as: separation from a curved surface, recirculation in the wake of the hill, a sharp shear layer, strong pressure gradients, and flow reattachment. The flow was used as a test case, e.g., for wall-resolved LES [13, 19, 28, 62, 159, 178] and wall models [13]. In the context of RANS and hybrid RANS/LES, this flow was analyzed rigorously within the European initiative ‘Advanced Turbulence Simulation for Aerodynamic Application Challenges’ (ATAAC), of which the final report by Jakirlić is available at [210]. In the subsequent chapters, this flow is used to assess several modeling approaches, such as RANS and DES (Chapter 5), and wall models (Chapters 7 to 10).

The first results of ILES for the periodic hill flow at a Reynolds number of $\mathrm{Re}_H = 10{,}595$ were presented in [160] using this scheme and the computations presented in the following sections were published in [149].

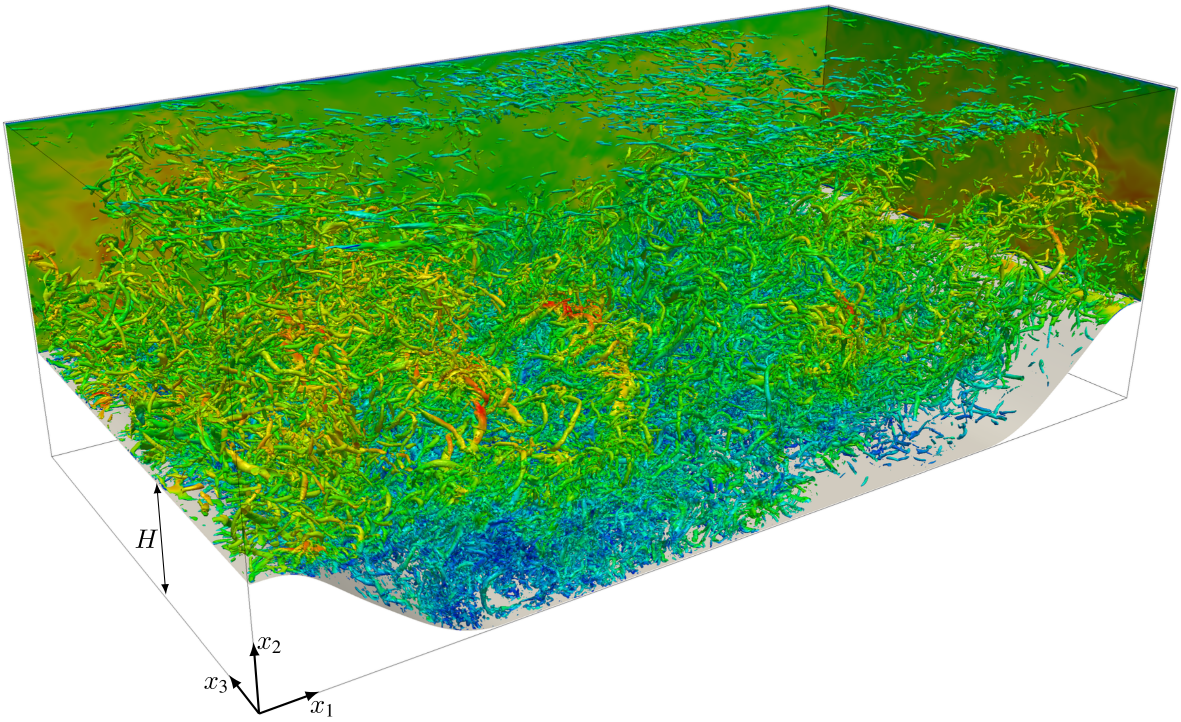


Figure 4.4.: Visualization of the turbulent flow structures at $\mathrm{Re}_H = 10{,}595$ via iso-surfaces of the Q-criterion colored by velocity magnitude. Red indicates high and blue low velocity.

4.3.1. Previous Reference Data

Several reference databases exist for this flow and an overview is given in Tables 4.3 and 4.4 for two Reynolds numbers. The flow has been investigated numerically via LES by Fröhlich et al. [93] at a Reynolds number $\mathrm{Re}_H = Hu_b/\nu = 10{,}595$, based on the hill height H and bulk velocity above the hill u_b. The width of the domain in spanwise direction was studied and a value of $4.5H$ was found sufficient, which has since been used in most studies. As a numerical scheme, two independent second order accurate finite volume methods were used and the upper wall was modeled by a wall function. In Table 4.4, these cases are labeled as FMRTL R1 and R2, respectively, corresponding to RUN1 and RUN2 in [93]. The reference database has been complemented by Breuer et al. [32], who performed simulations over a wide range of Reynolds numbers, starting from $\mathrm{Re}_H = 700$ and up to $\mathrm{Re}_H = 10{,}595$. Again, second order accurate finite volume schemes were used and the simulations are fully resolved (DNS) up to $\mathrm{Re}_H = 5{,}600$ and well-resolved (LES) at $\mathrm{Re}_H = 10{,}595$. These cases are referred to as BPRM (DNS/LES) in Tables 4.3 and 4.4. Finally, Rapp and Manhart [211] conducted water-channel experiments of the Reynolds numbers $\mathrm{Re}_H = 5{,}600$, $\mathrm{Re}_H = 10{,}600$, $\mathrm{Re}_H = 19{,}000$, and $\mathrm{Re}_H = 37{,}000$, labeled as RM in Tables 4.3 and 4.4. The data of Breuer et al. [32] and Rapp and Manhart [211] is available in the ERCOFTAC QNET-CFD Wiki contributed by Rapp et al. [210] alongside a detailed test case description. Further experimental data was recently acquired at the Reynolds numbers $\mathrm{Re}_H = 8{,}000$ and $\mathrm{Re}_H = 33{,}000$ [131].

Table 4.3.: $\mathrm{Re}_H = 5{,}600$: DNS and highly resolved LES (HRLES) cases in comparison to reference data, including polynomial degree k, order of accuracy $(k+1)$, total number of grid points N, sampling time Δt_{aver} in number of flow-through times T^*, separation and reattachment length $x_{1,\text{sep}}$ and $x_{1,\text{reatt}}$. The number of nodes of the present DG solver is obtained by the number of cells in each spatial direction times the number of nodes per cell $(k+1)$ in each direction. All cases are incompressible.

label	grid points	k	order	N	type	$\frac{\Delta t_{\text{aver}}}{T^*}$	$\frac{x_{1,\text{sep}}}{H}$	$\frac{x_{1,\text{reatt}}}{H}$
HRLES 5600	448×224×224	6	7	22.5M	HRLES	61	0.16	4.82
DNS 5600	512×256×256	7	8	33.6M	DNS	61	0.17	5.04
BPRM [32]	280×220×200	-	2	13.1M	LES	140	0.18	5.09
BPRM [32]	765×750×404	-	2	231M	DNS	38	0.18	5.14
RM [211]	-	-	-	-	Exp.	-	-	4.83

The available reference data shows significant differences, in particular between the experimental and the numerical DNS/LES data; for example, the reattachment lengths range from $4.83H$ (experiments) up to $5.14H$ (DNS) in the case $\mathrm{Re}_H = 5{,}600$ and from $4.21H$ (experiments) to $4.72H$ (LES) in the case $\mathrm{Re}_H = 10{,}595$.

Further simulations with higher resolution and high-order accurate schemes have been presented in [13] (labeled BPP in Table 4.4) and [62] (labeled DM in Table 4.4), however, these studies employed compressible solvers at low Mach numbers and comparably short averaging times of 15 flow-through times[1] and 25 flow-through times, respectively, which renders them unsuitable as reference data [62].

4.3.2. Extension of the Reference Database

As a primary concern of this section, we shed light on the differences visible in the previous reference data sets, discuss sources of error, and provide a new set of reference data for the Reynolds numbers $\mathrm{Re}_H = 5{,}600$ and $\mathrm{Re}_H = 10{,}595$, which will be used in the remainder of this thesis as reference data. Possible weaknesses of the previous numerical reference data may be:

- All reference computations were carried out using similar numerical schemes of second order spatial accuracy, which consequently also have similar dissipation and dispersion properties. In order to provide new insights into the flow cases, we perform simulations using the present DG scheme at high-order spatial accuracy with very low dissipation and dispersion characteristics (see Section 4.1).
- The averaging times of the numerical reference data lie in a broad range of approximately $\Delta t_{\text{aver}}/T^* = 38$ to $\Delta t_{\text{aver}}/T^* = 140$ flow-through times, based on the sampling time Δt_{aver} and a single flow-through time $T^* = 9H/u_b$. In this work, we investigate the issue of the sampling time and estimate the sampling error in order to assess the relevance of the averaging error in the context of the most widely used computational setup.

---

[1] Private communication with Ponnampalam Balakumar

Table 4.4.: $\mathrm{Re}_H = 10{,}595$: DNS and highly resolved LES (HRLES) cases in comparison to reference data, including polynomial degree k, order of accuracy $(k+1)$, total number of grid points N, Mach number Ma/incompressible (inc.), sampling time Δt_{aver} in number of flow-through times T^*, separation and reattachment length $x_{1,\text{sep}}$ and $x_{1,\text{reatt}}$. The number of nodes of the present DG solver is obtained by the number of cells in each spatial direction times the number of nodes per cell $(k+1)$ in each direction.

label	grid points	k	order	N	type	Ma	$\frac{\Delta t_{\text{aver}}}{T^*}$	$\frac{x_{1,\text{sep}}}{H}$	$\frac{x_{1,\text{reatt}}}{H}$
HRLES 10595	768×384×384	5	6	113M	HRLES	inc.	61	0.19	4.57
DNS 10595	896×448×448	6	7	180M	DNS	inc.	61	0.20	4.51
FMRTL R1 [93]	196×128×186	-	2	4.7M	LES	inc.	55	0.20	4.56
FMRTL R2 [93]	196×128×186	-	2	4.7M	LES	inc.	55	0.22	4.72
BPRM [32]	280×220×200	-	2	13.1M	LES	inc.	140	0.19	4.69
DM [62]	384×192×192	7	8	14M	ILES	0.1	25	0.20	4.37
BPP [13]	800×500×500	-	7	200M	DNS	0.2	15[1]	-	4.5
RM [211]	-	-	-	-	Exp.	inc.	-	-	4.21

- The available skin friction and pressure coefficient curves show numerical oscillations due to the spatial discretization. It is the ambition of this work to provide these quantities with a higher quality.

In addition to the highly resolved simulations, the present DG solver is assessed regarding ILES with this benchmark example in a detailed h/p-refinement analysis using the new reference data. In the following section, we discuss the simulation setup, present the meshes and discuss the resolution of the DNS. The DNS results are shown in Section 4.3.4 and the assessment of ILES is presented in Section 4.3.5.

4.3.3. Simulation Setup, Resolution Requirements, and Mesh

We give an overview of the simulation setup, boundary conditions, and the mesh used for the DNS simulations.

4.3.3.1. Simulation Setup

We consider a domain of the dimensions $9H\times 3.036H\times 4.5H$ in streamwise, wall-normal, and spanwise direction, respectively. Periodic boundary conditions are applied in streamwise and spanwise direction and no-slip conditions at the lower and upper wall. The flow is driven as in [93] by a pressure gradient, represented by a body force $\boldsymbol{f} = (f_1, 0, 0)^T$, where f_1 is constant in space. The volume flux $Q = \int_\Omega u_1 \, \mathrm{d}\Omega / 9H$, and thus the Reynolds number, is kept constant in time by a control algorithm, which is similar to the one proposed in Benocci and Pinelli [23] and detailed in the following. We use a simple proportional control algorithm with an additional damping term to ensure a constant mass flux. We define

$$f_1^{n+1} = f_1^n + b_1(Q^0 - Q^n) - b_2(Q^n - Q^{n-1}), \tag{4.2}$$

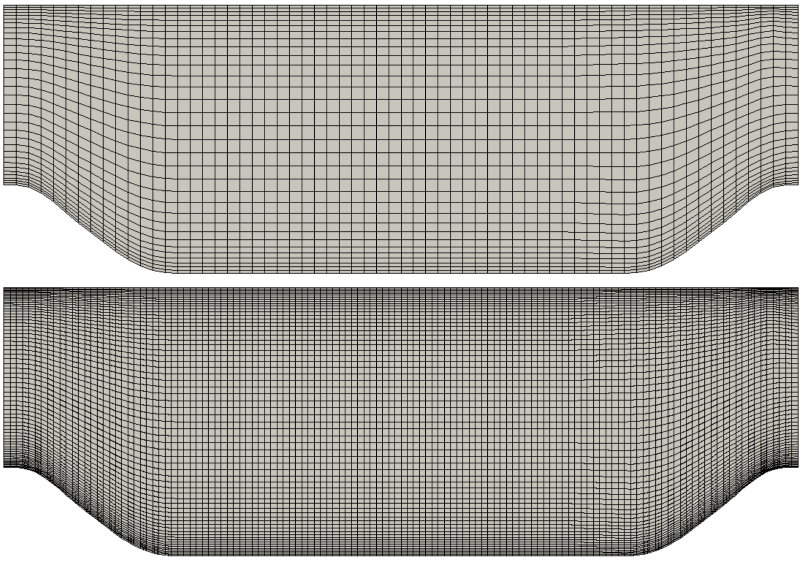


Figure 4.5.: Grid for DNS. Top: DNS 5600. Bottom: DNS 10595. In each grid cell, the solution is represented by polynomials of 7th ($\mathrm{Re}_H = 5{,}600$) and 6th ($\mathrm{Re}_H = 10{,}595$) degree and the scheme exhibits an order of accuracy of polynomial degree plus one.

where b_1 is the gain of the proportional controller and b_2 of the damper. If we would choose $b_1 = b_2 = 1/\Delta t$, the controller would be identical to the frequently used algorithm in [23]. In practice, we have found that another empirical choice of b_1 and b_2 increases the robustness of the approach. For the present computations, we choose $b_1 = 500$ and $b_2 = 30{,}000$, corresponding to $H = 0.028$, $\rho = 1$, and $u_b = 5.621$, yielding $Q^0 = 0.040376$. With this choice, the mass flux is constant to five digits of accuracy after the start-up.

The simulations are run for about 84 flow-through times T^* of which statistics are sampled during $\Delta t_{\text{aver}} = 61T^*$ after the initial transient. The statistics are also averaged over the homogeneous spanwise direction and more than 100,000 samples are used in the averaging process for the highly resolved simulation cases in order to minimize the error stemming from the statistical postprocessing.

4.3.3.2. Meshes

For the DNS cases, we use a spatial discretization with 64×32×32 cells with a polynomial degree of $k = 7$, yielding 512×256×256 grid points and a spatial accuracy of 8th order, for the case $\mathrm{Re}_H = 5{,}600$, and a grid of 128×64×64 cells with a polynomial degree of $k = 6$, yielding 896×448×448 grid points and a spatial accuracy of 7th order, for the case $\mathrm{Re}_H = 10{,}595$. These meshes are displayed in Figure 4.5. The grid is equally spaced in streamwise and spanwise direction and mildly stretched towards the walls in order to improve the near-wall resolution. We refrain from refining the grid locally, for example near the hill crest or in the shear layer. In particular, the locations of the separation point on the hill crest and the shear layer are highly unsteady [93], thus a locally refined grid design based on averaged flow quantities may not be sufficient to fully resolve the unsteady flow. In addition, the propagation of small scales through

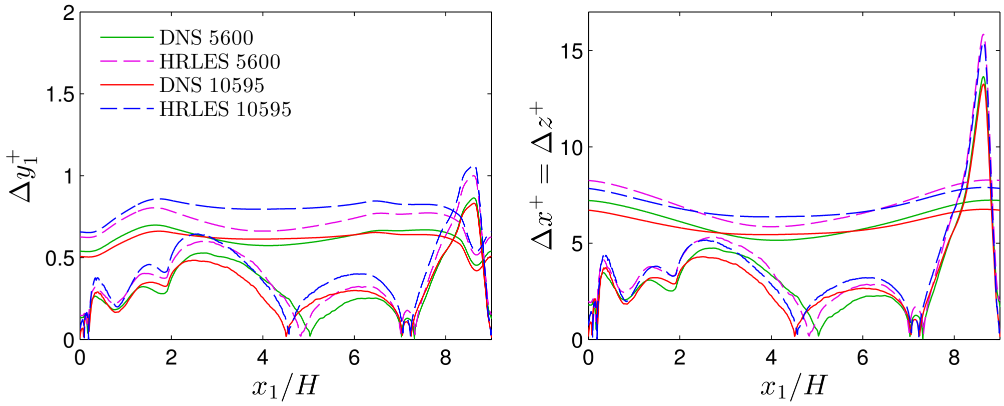


Figure 4.6.: Location of first off-wall Gauss–Lobatto point $\Delta y^+_{1,\mathrm{GL}}$ (left) as well as normalized grid spacings in streamwise and spanwise direction $\Delta x^+ = \Delta z^+$ (right). The shallower curves correspond to the upper wall.

streamwise stretched grids has a dissipating effect and through streamwise refined grids an anti-dissipation-effect, which has for example been investigated for standard second order central differences in [155], and the implications of such behavior on the turbulent flow are unclear. In order to guarantee a highly accurate representation of the boundary, the curved hill geometry is represented by the same polynomial degree as the solution with mapping facilities provided by the deal.II library [9]. This mapping and a preliminary version of the problem setup was implemented in INDEXA by Legat [173]. The vertical grid lines are aligned parallel to the vertical axis and it is noted that orthogonality of the grid is not relevant in the context of DG as the stencils used herein are truly three-dimensional. A second grid is considered for each Reynolds number with a very similar layout, but with one degree lower (p-coarsening). The DNS cases are in the following labeled DNS 5600 and DNS 10595, respectively, and the p-coarsened cases HRLES 5600 and HRLES 10595, HRLES denoting highly resolved LES, according to Tables 4.3 and 4.4.

4.3.3.3. Assessment of Resolution

The mesh resolution is evaluated using criteria for the near-wall region and the bulk flow as in [32, 93]. Figure 4.6 shows the location of the first off-wall Gauss–Lobatto point, $\Delta y^+_{1,\mathrm{GL}}$, as well as the dimensionless streamwise and spanwise grid-spacings $\Delta x^+ = \Delta z^+$. The quantity Δy_1^+ is below 0.70 at the upper wall for both DNS cases. At the lower wall, Δy_1^+ exhibits a characteristic peak on the windward side of the hill crest with a maximum of $\max(\Delta y_1^+) = 0.86$. In the streamwise and spanwise directions, the distribution is similar with maximum values of $\max(\Delta x^+) = 7.2$ at the upper wall and a more pronounced peak at the lower wall of up to $\max(\Delta x^+) = 13.7$. This resolution is well sufficient to fully resolve the velocity gradient and the turbulent scales at the wall.

The resolution of the bulk flow is assessed by comparing the grid filter width $h = h_e/(k+1)$ (see Section 4.1) with the resolved Kolmogorov length $\eta = (\nu^3/\tilde{\varepsilon})^{1/4}$. The grid size is defined based on a cell volume $V(\Omega_e)$ as $h_e = V(\Omega_e)^{1/3}$ neglecting direction-dependence, however, the cells in the bulk flow are almost cubic, since $\Delta x = \Delta z$ throughout and Δy is in a similar range.

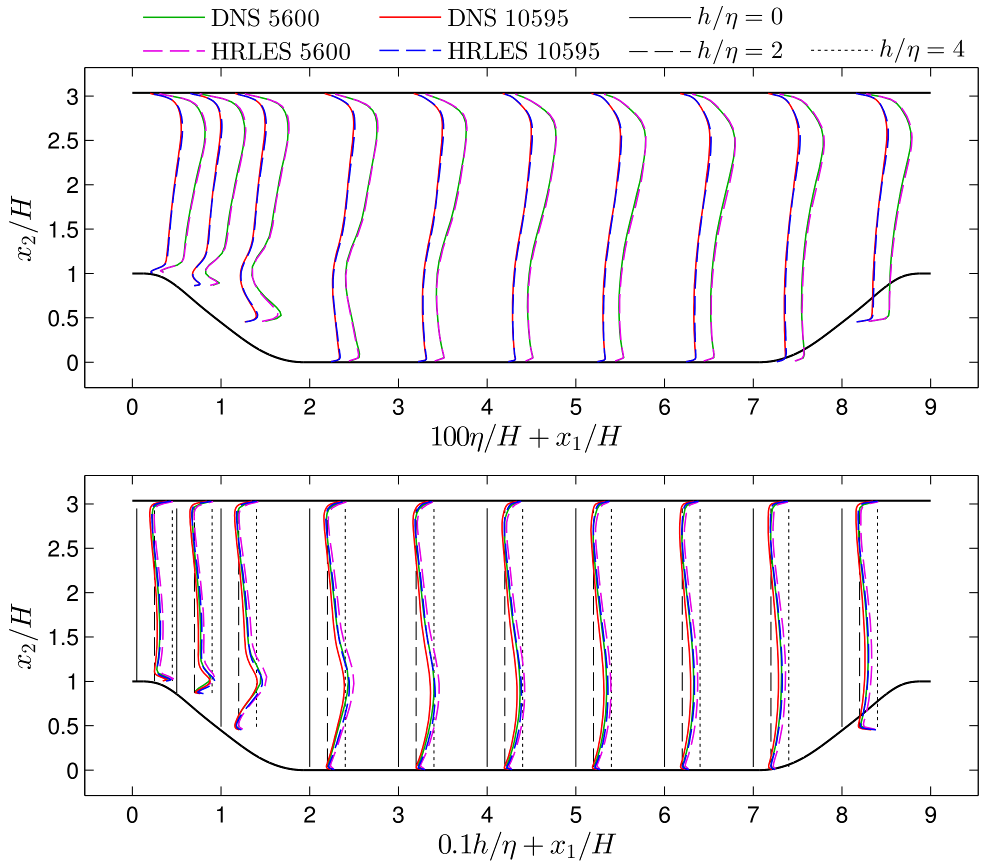


Figure 4.7.: Assessment of resolution: Kolmogorov length through resolved dissipation (top) and ratio of grid size to Kolmogorov length (bottom).

The dissipation rate is computed via the resolved velocity field vector $\boldsymbol{u}$ as $\tilde{\varepsilon} = \nu((\nabla\boldsymbol{u})_{ij}(\nabla\boldsymbol{u})_{ij})$ in index notation (see [206] for further details) and averaged over time and the spanwise direction. In Figure 4.7 (top), we show averaged profiles of η at ten streamwise stations, located at $x_1/H = \{0.05, 0.5, 1, 2, 3, 4, 5, 6, 7, 8\}$, and compare the respective DNS and HRLES cases. As essentially no difference is visible despite the varying resolution, it is assumed that the subgrid scale contribution to the dissipation rate is negligible. In Figure 4.7 (bottom), the ratio h/η is plotted over all ten stations and iso-lines at values of 0, 2, and 4 are included as a reference. The value of h/η is located between 2 and 4 with peaks in the shear layer of 4.48 in the case DNS 5600 and 4.07 in the case DNS 10595. As already described in Section 2.3.1, Pope [206] specifies the 24η-range as the length scale where most dissipation occurs and Moin and Mahesh [184] state that most dissipation is resolved above 15η, and the latter publication lists DNS examples with a resolution around $h/\eta \sim 4$. Considering the spectral characteristics of the present scheme (see Section 4.1), we regard the resolution of the DNS cases sufficient to be denoted *direct numerical simulation*. The HRLES cases exhibit $\max(h/\eta) = 5.15$ and also use a scheme with one degree lower, which exhibits larger dissipation and dispersion errors, and are therefore denoted *highly resolved LES*.

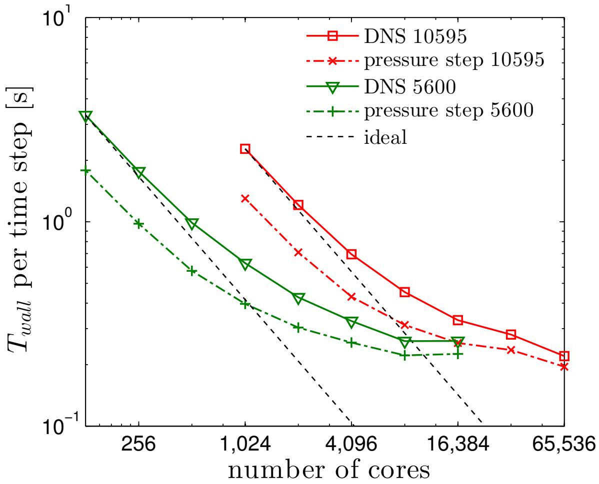


Figure 4.8.: Parallel scaling of the DNS setups at both Reynolds numbers up to 65,536 CPU cores.

4.3.3.4. Computational Cost

The DNS are performed on Phase 1 of SuperMUC in Garching, Germany, on 2×8 core Intel Sandy Bridge CPUs at 2.7GHz. Scaling experiments for both cases are shown in Figure 4.8. At $\mathrm{Re}_H = 5{,}600$, the solver exhibits very good strong scaling up to 2,048 CPU cores and the curve flattens up to 16,384 CPU cores. The case $\mathrm{Re}_H = 10{,}595$ yields excellent strong scaling up to 8,192 CPU cores and shows a speed-up until 65,536 CPU cores. All steps of the time integration scheme show close-to-optimal scaling, except for the pressure Poisson algorithm, which involves a significant amount of communication; the scaling of the pressure Poisson solver is also included in Figure 4.8. Regarding the simulation at $\mathrm{Re}_H = 10{,}595$, the shortest absolute wall time achieved in this scaling test is approximately 0.22s per time step at 65,536 compute cores. As a good compromise between resource efficiency and run time, the final computations were performed on 8,192 cores, yielding a wall time per time step T_{wall} of approximately 0.45s and an overall run time of approximately six weeks for 8.5 million time steps.

4.3.4. Direct Numerical Simulation

We present the results for the skin friction and pressure coefficients as well as velocity statistics separately for each Reynolds number and compare the DNS with the available reference data. While it is the aim of this study to provide new reference data for both Reynolds numbers, the results of the present work are first compared to highly resolved reference data at the lower Reynolds number to validate the simulation setup. Subsequently, some more controversial issues are discussed regarding the higher Reynolds number. The last part of this section presents an analysis of the statistical sampling error including the confidence interval of the reattachment length.

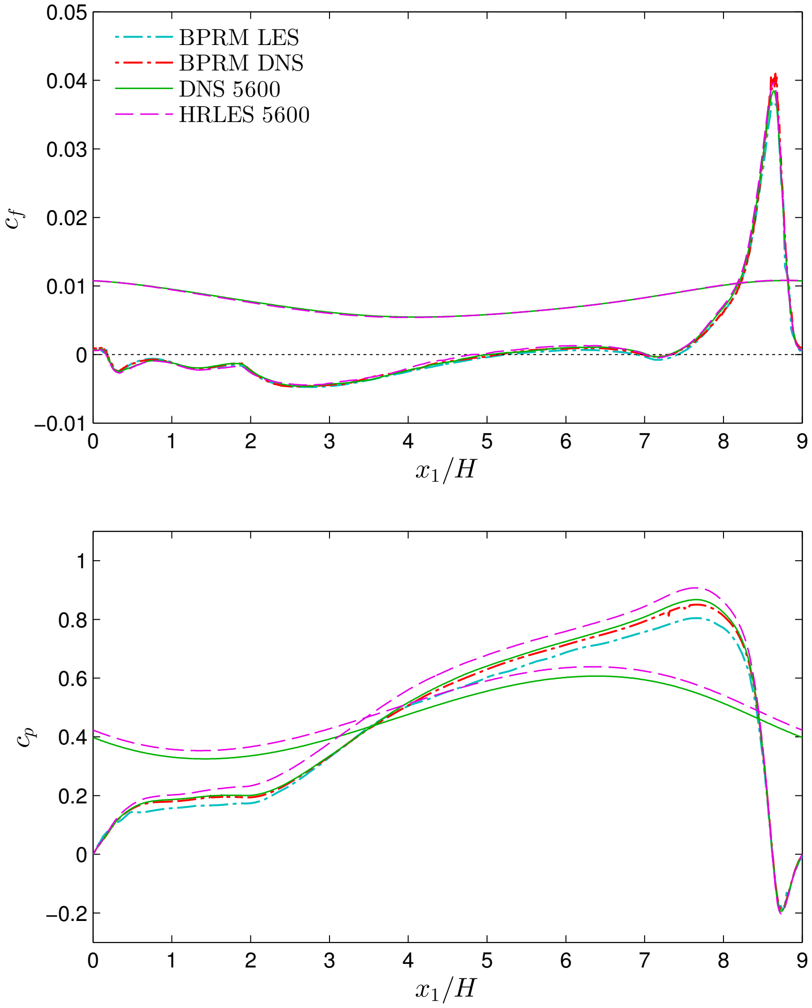


Figure 4.9.: $\mathrm{Re}_H = 5{,}600$: Skin friction (top) and pressure (bottom) coefficients. The shallower curves correspond to the upper wall. Results for the cases BPRM LES and BPRM DNS are only available on the lower wall.

4.3.4.1. $\mathrm{Re}_H = 5{,}600$: Skin Friction and Pressure Coefficients

We commence the discussion of the simulation results by considering the skin friction and pressure coefficients

$$c_f = \frac{\tau_w}{\frac{1}{2}\rho u_b^2}, \tag{4.3}$$

$$c_p = \frac{\rho(p - p_{\mathrm{ref}})}{\frac{1}{2}\rho u_b^2}, \tag{4.4}$$

with the wall shear stress τ_w, the kinematic pressure p and a reference pressure p_{ref} at the probe location at $x_1 = 0$ on the bottom wall. The pressure coefficient is multiplied by the density since the kinematic pressure is used throughout this work. The wall-parallel component of the stress is computed on the hill shape at the lower boundary by

$$\tau_w = \operatorname{sgn}\left(\frac{\partial u_1}{\partial x_2}\right)\rho\nu\sqrt{\left(\frac{\partial u_1}{\partial x_1}n_1 + \frac{\partial u_1}{\partial x_2}n_2\right)^2 + \left(\frac{\partial u_2}{\partial x_1}n_1 + \frac{\partial u_2}{\partial x_2}n_2\right)^2} \tag{4.5}$$

and using the components of a wall-normal vector $\boldsymbol{n} = (n_1, n_2, n_3)^T$ of unit length, and the sign function sgn. Note that this formulation is equivalent to the alternative $\tau_w = \rho\nu\boldsymbol{t}^T \cdot (\nabla\boldsymbol{u} \cdot \boldsymbol{n})$ (evaluated in the x_1x_2 plane, with the tangential vector $\boldsymbol{t}$, in the continuous sense. The results for both coefficients at the upper and lower wall are shown in Figure 4.9 and the curves are compared to the DNS and LES results of [32], labeled BPRM DNS and BPRM LES, respectively, according to Table 4.3. Regarding the skin friction, the cases DNS 5600 and HRLES 5600 are in very good agreement overall. Between $x_1/H = 4$ and $x_1/H = 7$, the case HRLES 5600 overpredicts the skin friction to a minor extent. In comparison to the reference data, the agreement between the two DNS is excellent. However, the case BPRM DNS exhibits strong oscillations near the peak of the skin friction at the hill crest, as well as minor oscillations at the lower wall around $x_1/H = 0$ and $x_1/H = 7$ where the boundary is curved, while the present DNS case yields smooth data. The simulation BPRM LES predicts the skin friction in the region $x_1/H = 4$ to $x_1/H = 7$ at the bottom wall slightly lower, but the agreement is on a very high level overall. The profiles of the skin friction at the lower wall predict the presence of two recirculation bubbles in accordance with the reference simulations [32]. The primary recirculation zone separates around $x_{1,\text{sep}}/H = 0.17$ and the reattachment point lies near $x_{1,\text{reatt}}/H = 5.0$, determined via the zero-crossings of the skin friction. These locations are in good agreement with the reference data and the exact values are compared in Table 4.3. Furthermore, a separate section is devoted to a discussion of the quality of the predicted reattachment point, see Section 4.3.4.5. That discussion concludes that the reattachment lengths of the cases BPRM LES, BPRM DNS, and DNS 5600 are in agreement with each other within the available accuracy, while the case HRLES may predict the reattachment point slightly shorter. The additional data provided by experiments [211] also predicts the reattachment length marginally shorter at $x_{1,\text{reatt}}/H = 4.83$. The second recirculation bubble is located just in front of the hill at the windward side, between $x_1/H = 7.04$ and $x_1/H = 7.31$ considering the results of the present DNS.

There are more significant differences in the pressure coefficient. The largest deviations between the simulation cases are visible in the range $x_1/H = 0$ to $x_1/H = 1$ at the lower wall, where the pressure sees a strong positive pressure gradient. The length of this zone with positive pressure gradient varies significantly for the different flow cases. Since the pressure at $x_1/H = 0$ at the lower wall is used as the reference pressure, this discrepancy shifts the pressure curves considerably. We argue therefore that the pressure at $x_1/H = 0$ at the upper wall should be used as the reference pressure in future work, which has shown to be much more robust with respect to the resolution near the hill crest. This pressure value is unfortunately not available for all reference data sets considered herein. While the cases HRLES 5600 and BPRM LES are affected by this shift, the two DNS cases yield very good agreement overall. The simulations BPRM LES and HRLES 5600 show noticeable differences to the two DNS.

The results discussed up to this point show particularly good accordance with the highly resolved reference simulation BPRM DNS, and the new reference results are of even higher quality due to the absence of oscillations in the skin friction and the longer averaging time (see Section 4.3.4.5).

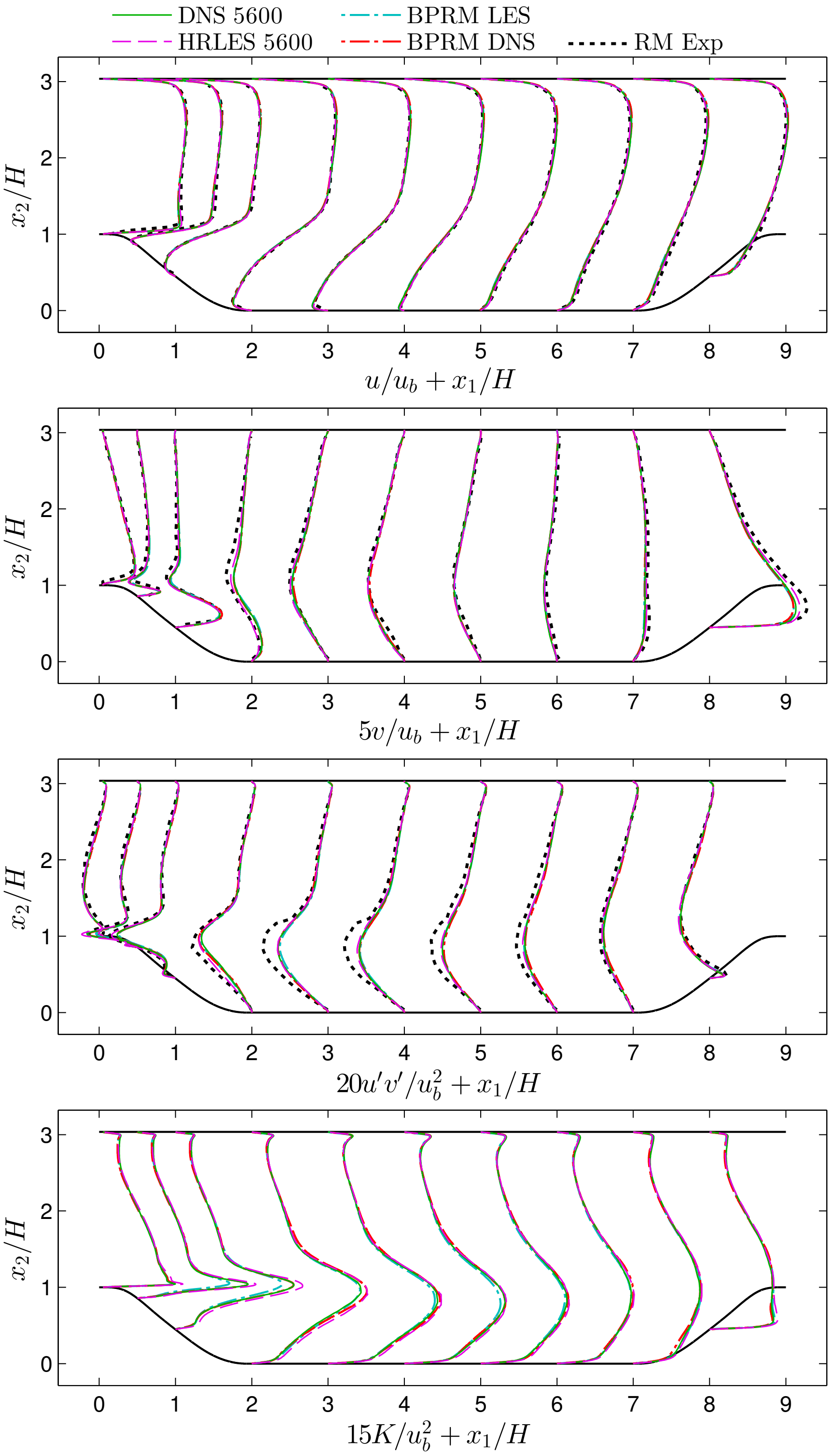


Figure 4.10.: Re$_H$ = 5,600: Profiles of mean velocity in streamwise and vertical direction u and v, RSS $u'v'$, as well as TKE $K = 1/2(u'u' + v'v' + w'w')$ (from top to bottom).

4.3.4.2. $\mathrm{Re}_H = 5{,}600$: Velocity Statistics

The same cases are further compared via profiles of the streamwise and vertical mean velocity, RSS, and turbulence kinetic energy (TKE) in Figure 4.10. The data is plotted at ten streamwise stations, which are located at $x_1/H = \{0.05, 0.5, 1, 2, 3, 4, 5, 6, 7, 8\}$. Additionally, the simulation results are compared to the experimental data available [211], including the mean velocity profiles and the RSS. The results of the mean velocity profiles of all four simulation cases are in excellent agreement, and the accordance of the streamwise velocity with the experimental data is good. Regarding the vertical velocity, the experimental data exhibits the largest deviations from the other data sets. At the fourth ($x_1/H = 2$) and last station ($x_1/H = 8$), the case HRLES 5600 also shows differences to the three other simulation cases.

With respect to the RSS, the picture gets more diverse. Firstly, the experimental data shows values of much higher magnitude than the simulation cases. This effect may be due to the different spanwise extension of the experimental and numerical domains [131]. At the three stations up to $x_1/H = 1$, the two DNS simulations exhibit an excellent agreement, while the other cases show a tendency of overpredicting (HRLES 5600) and underpredicting (BPRM LES) the magnitude in the shear layer above the recirculation zone. The agreement of the numerical data at $x_1/H \geq 3$ is generally good. At $x_1/H = 2$, the new DNS (DNS 5600) shows values of smaller magnitude in the shear layer than in the DNS reference. This deviation may either be due to a statistical sampling error, or due to the significant mesh coarsening in streamwise direction taking place in the case of BPRM DNS near $x_1/H = 2$, which can have a dissipative effect caused by the behavior of second-order central difference schemes (see Section 4.3.3.2).

Finally, profiles of the TKE show even more pronounced differences in the shear layer region above the recirculation zone. The two DNS exhibit excellent accordance up to $x_1/H = 1$. The case BPRM LES shows an underprediction of the profile in the shear layer. This behavior may indicate a minor underresolution in that region. The case HRLES 5600 may also be slightly underresolved in that region, as the peak in the shear layer is higher than the DNS, a characteristic behavior indicating underresolution in DG (cf. results presented in Section 4.2). At $x_1/H = 2$, the cases BPRM DNS and DNS 5600 deviate significantly, for which again the change in resolution of the case BPRM DNS may be held responsible. The differences of the numerical data are less significant at the downstream stations.

From this discussion, we conclude that

- the two DNS cases are in excellent agreement overall,
- the new DNS data does not show oscillations in the skin friction, whereas the case BPRM DNS does,
- the new DNS data has a longer sampling time than the case BPRM DNS, which reduces the uncertainty of the quantities (cf. Section 4.3.4.5),
- the present simulation cases employ equidistant grids in streamwise direction, which excludes artificial numerical dissipation errors due to such a grid stretching.

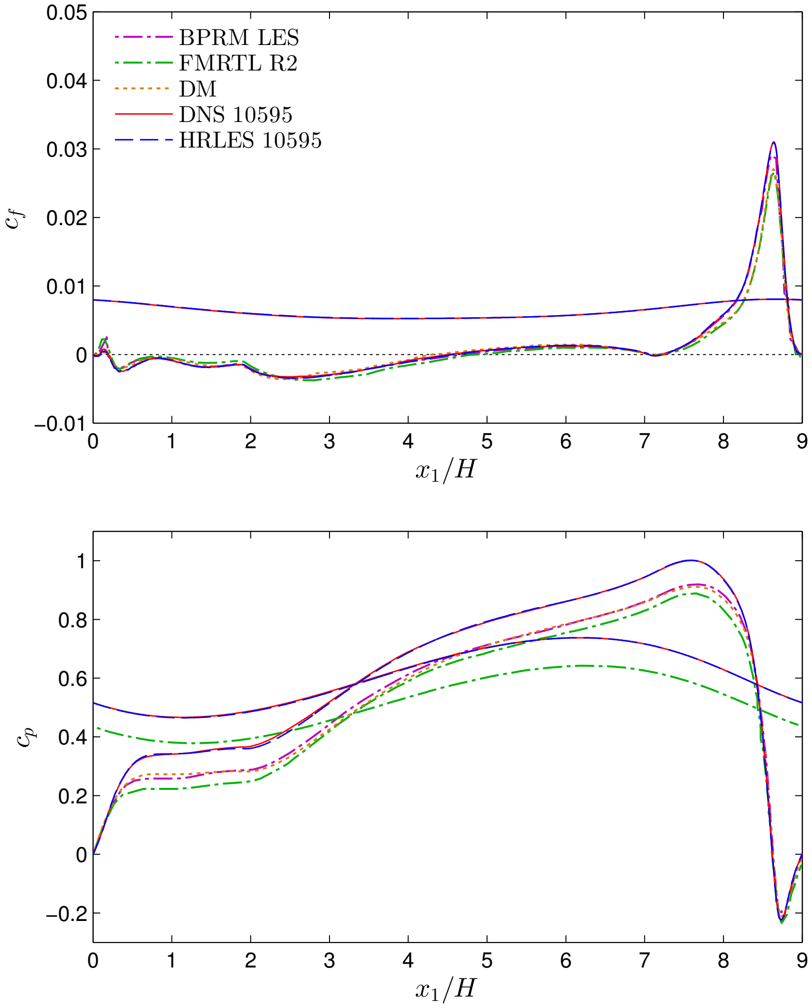


Figure 4.11.: $\text{Re}_H = 10{,}595$: Skin friction (top) and pressure (bottom) coefficients. The shallower curves correspond to the upper wall. Reference results at the upper wall are only available for c_p through the case FMRTL R2.

4.3.4.3. $\text{Re}_H = 10{,}595$: Skin Friction and Pressure Coefficients

In Figure 4.11, we compare the profiles of the skin friction and pressure coefficients to the available reference data of the cases BPRM LES [32], FMRTL R2 [93], and DM [62], according to Table 4.4. Regarding the skin friction, we notice significant disagreement between $x_1/H = 7$ and $x_1/H = 9$: in the acceleration zone at the windward side of the hill, the present simulation cases HRLES 10595 and DNS 10595 show very good agreement with BPRM LES in that region, whereas the cases FMRTL R2 and DM predict a significantly smaller skin friction, but agree well with each other. This difference may be due to a different postprocessing of the wall shear stress. In this work, we consider the wall-parallel wall shear stress τ_w, which takes into account the slope of the hill shape and considers the contributions from both velocity components, u_1 and u_2, according to Equation (4.5). We have found that the DNS results agree well with the cases FMRTL R2 and DM in that region if only the u_1-component of the wall shear stress, $\tau_{w,1}$, is taken into account according to

$$\tau_{w,1} = \rho\nu \left(\frac{\partial u_1}{\partial x_1} n_1 + \frac{\partial u_1}{\partial x_2} n_2 \right). \tag{4.6}$$

This component can be computed from τ_w by

$$\tau_{w,1} = \frac{\tau_w}{\sqrt{1 + \left(\frac{\mathrm{d}g}{\mathrm{d}x_1}\right)^2}}, \tag{4.7}$$

where $\frac{\mathrm{d}g}{\mathrm{d}x_1}$ is the slope of the function g that defines the hill shape. Unfortunately, none of the references gives details on how the wall shear stress was computed on the hill. We consider the wall-parallel definition of τ_w as more consistent, in particular also for the assessment of the near-wall resolution through the y_1^+ criterion, but $\tau_{w,1}$ may also be a possible choice if indicated clearly. Note that this issue is also present on the lee side of the hill, although not as clearly visible.

However, in the region between the hills, $2 \leq x_1/H \leq 7$, which is most interesting, the results are not affected by the difference in the prostprocessing as the wall is horizontal, allowing a comparison of the curves. The data sets HRLES 10595, DNS 10595, DM and BPRM LES show good accordance in the region from $x_1/H = 2$ to $x_1/H = 4$, but the data of DM exhibits minor oscillations, which may be due to the comparably short sampling time used in that work or due to a remaining discretization error (see Section 4.3.5). The case FMRTL R2 underpredicts the wall shear stress in that region. Near the reattachment zone in the interval $x_1/H = 4$ to $x_1/H = 6$, the accordance of DM and the present simulation cases is still good, whereas both FMRTL R2 and BPRM LES yield an underprediction. Accordingly, the primary recirculation bubble separates at around $x_{1,\text{sep}}/H = 0.20$ and reattaches near $x_{1,\text{reatt}}/H = 4.5$, where the two cases FMRTL R2 and BPRM LES predict larger values and the case DM a shorter reattachment length as compared to the other three cases. The measurement data specifies the reattachment length significantly shorter as $x_1/H = 4.21$. All available data is compared in Table 4.4. In Section 4.3.4.5, we discuss these quantities from the perspective of the statistical sampling error. The secondary recirculation at the windward side of the hill separates at $x_1/H = 7.05$ and reattaches at $x_1/H = 7.24$ according to the case DNS 10595, which is in agreement with the reference data [32]. In addition to these two recirculations, a third recirculation may be observed at the hill crest according to the references [32, 93]. The separation point is computed to $x_1/H = 0.00$ and the reattachment to $x_1/H = 0.08$ based on the case DNS 10595. This recirculation is so thin that it is not visible in the mean velocity profiles presented in the subsequent section, where the first data point at $x_1/H = 0.05$ is located $0.0015/H$ above the wall. This may explain why this recirculation bubble was not found in the experiments in [131] at a similar Reynolds number.

Finally, it is remarkable how well the two simulation cases presented in this work agree with each other in the whole range and including the upper wall, indicating that the spatial discretization error, as well as other uncertainties, indeed are negligible in these cases.

Next, we discuss the pressure coefficient also included in Figure 4.11. The effect of the different length of the pressure ramp near $x_1/H = 0$ discussed in conjunction with the lower Reynolds number is even more severe in the present case. Again, this disagreement shifts the remaining part of the plot in vertical direction, as the reference pressure is considered at the lower wall at $x_1/H = 0$. The two curves computed in this study show an outstanding agreement throughout, which confirms that the discretization error in these cases is very small. The other simulation cases follow the trend well, despite the significant offset. Also, the pressure coefficient curves on the upper wall exhibit the same trend as the reference FMRTL R2 but with a shift of approximately 0.1. This aspect highlights once again that the value at $x_1/H = 0$ at the upper wall should be used as the reference pressure in future work, which would allow a better comparison of the curves.

4.3.4.4. $\mathrm{Re}_H = 10{,}595$: Velocity Statistics

The profiles of the averaged velocity, RSS, and TKE are plotted in Figure 4.12 for the Reynolds number $\mathrm{Re}_H = 10{,}595$ at ten stations. The present data is compared to the two LES reference cases, BPRM LES and FMRTL R2, as well as experimental results, where available. Regarding the mean streamwise velocity, the two data sets DNS 10595 and HRLES 10595 exhibit an excellent agreement. These curves lie mostly in between the experimental and LES references, for example above the hill crest or in the reattachment zone, but the differences are very small overall. With respect to the vertical velocity, there are more noticeable aspects. On the lee side of the hill, there is a distinct gap visible between the LES reference cases and the present data, and the new results are closer to the experimental reference. The last station at $x_1/H = 8$ shows a similar picture, as the present simulation results lie in the gap between the LES and experimental references.

Regarding the profiles of the RSS, the cases BPRM LES and FMRTL R2 underpredict the magnitude of the stress in the thin shear layer in the range $0 \leq x_1/H \leq 1$. The experimental data is in much better accordance to the numerical data than for the lower Reynolds number. The agreement of all data sets beyond $x_1/H = 3$ is good.

The profiles of the TKE of the cases DNS 10595 and HRLES 10595 are in excellent agreement, which allows the conclusion that the spatial discretization of the DNS is capable of resolving all relevant scales. In the sharp shear layer above the recirculation bubble, both LES reference cases predict considerably lower magnitudes. A possible explanation for these differences may be that the turbulent motions in this area are not sufficiently resolved. We note that this underresolution also may be the cause for the overestimation of the reattachment lengths by the two LES reference cases in comparison to the present DNS. The deviations of the profiles beyond the reattachment point are minor.

The following major conclusions may be drawn from the discussion of the Reynolds number $\mathrm{Re}_H = 10{,}595$:

- the DNS 10595 and HRLES 10595 cases show excellent agreement in all figures, which indicates that the spatial discretization error as well as other sources of error are minor,

- the widely used reference data BPRM LES and FMRTL R2 may not capture all relevant turbulent motions in the thin shear layer above the recirculation bubble sufficiently well, which may come along with an underestimation of the momentum exchange in that region, resulting in a longer recirculation,

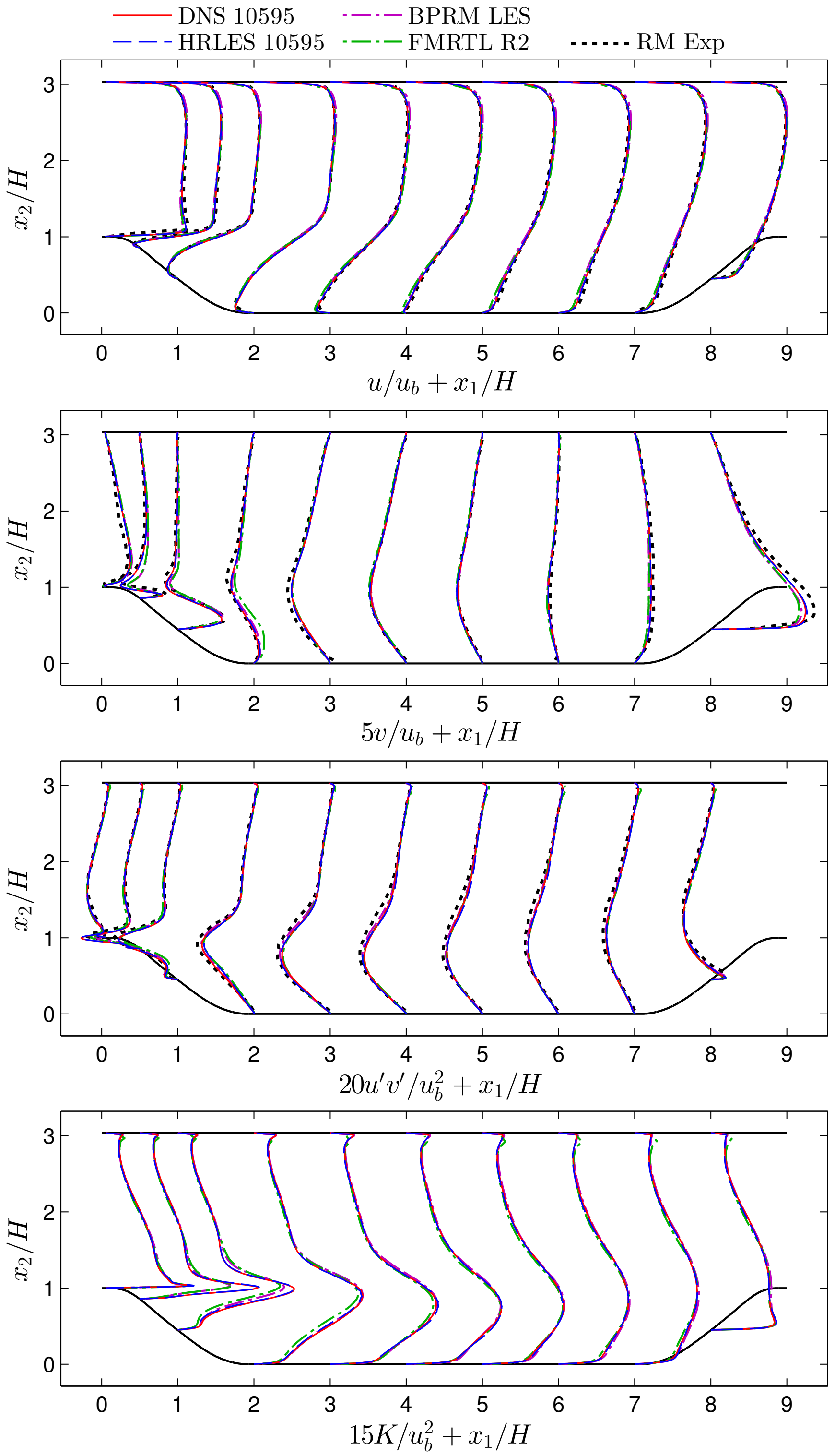


Figure 4.12.: $\mathrm{Re}_H = 10{,}595$: Profiles of mean velocity in streamwise and vertical direction u and v, RSS $u'v'$, as well as TKE $K = 1/2(u'u' + v'v' + w'w')$ (from top to bottom).

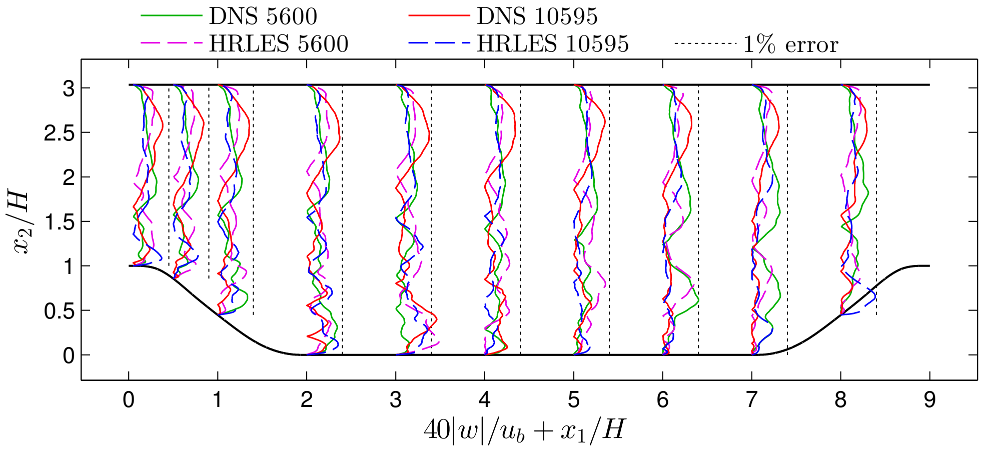


Figure 4.13.: Averaging error of the spanwise velocity component w (absolute value) and comparison to iso-lines of 1%.

- the DNS yields a shorter reattachment length than the previously available LES reference data, but longer than in the experiments.

4.3.4.5. Analysis of the Sampling Error

The averaging times considered in the reported simulation cases differ, according to Tables 4.3 and 4.4, by approximately one order of magnitude, which we see as an indication that this parameter should be investigated in more detail. Beginning with the mean velocity components, a frequently used measure for the averaging error in these quantities is the deviation of the spanwise velocity, which should vanish in the mean. The absolute value of the spanwise velocity is plotted in Figure 4.13 and iso-lines indicate the 1% error level. It may be observed that the error of the spanwise velocity is consistently below 1%, most of the time even significantly below 0.5%; an error that seems to be acceptable.

However, it was already reported by Rapp [209] in the context of the measurements that it was difficult to determine the reattachment length of the primary recirculation bubble due to low-frequency motions of the reattachment point in streamwise direction. The recent experimental study [131] further elaborates on the issue of rare events having an influence on the flow statistics. This behavior motivates a separate analysis of the averaging error of the reattachment length. This length is of paramount importance in this benchmark case, as it is frequently used as an 'error norm', summarizing the overall performance of a numerical method or model in one number.

In order to analyze the statistical convergence behavior of this quantity in detail, we plot intermediate convergence results of $x_{1,\text{reatt}}/H$ over the respective sampling time in number of flow-through times $\Delta t_{\text{aver}}/T^*$, see Figure 4.14 for $\text{Re}_H = 5{,}600$ and Figure 4.15 for $\text{Re}_H = 10{,}595$. These data points are subject to an error bandwidth, which converges as $\sim \pm\text{const}/\sqrt{\Delta t_{\text{aver}}/T^*}$ according to the statistical theory of the 'standard error of the mean'. In order to estimate the error at the end of the sampling time, this error bandwidth is also plotted in Figures 4.14 and 4.15 by considering the curves $x_{1,\text{reatt}}/H \pm \alpha/\sqrt{\Delta t_{\text{aver}}/T^*}$ and adjusting the constant parameter α

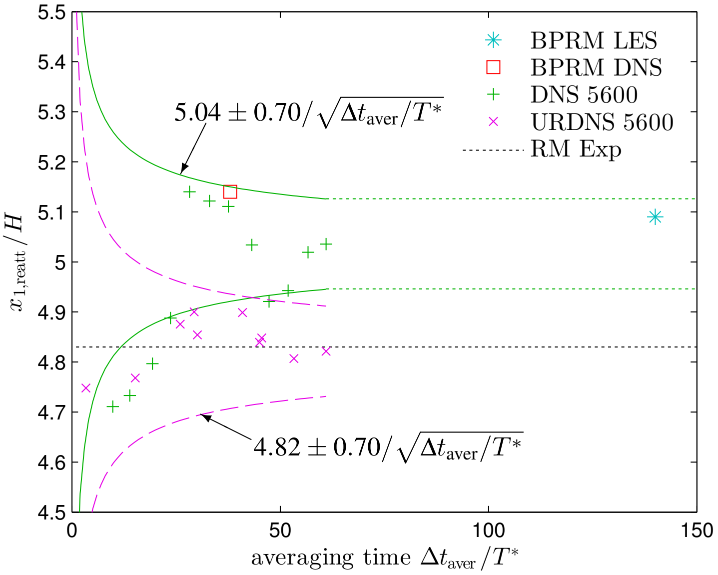


Figure 4.14.: $\text{Re}_H = 5{,}600$: Analysis of the statistical averaging error of the reattachment length $x_{1,\text{reatt}}/H$. The error bandwidth is given as $x_{1,\text{reatt}}/H \pm \text{const}/\sqrt{\Delta t_{\text{aver}}/T^*}$ and yields an estimation of the error after 61 flow-through times of $\pm 0.09H$.

manually such that most points lie within the bandwidth. This procedure yields an estimation of the sampling error for $x_{1,\text{reatt}}/H$ of $\pm 0.09H$ for $\text{Re}_H = 5{,}600$ and $\pm 0.06H$ for $\text{Re}_H = 10{,}595$.

Both plots include the available reference data, which may be compared to the error bandwidth of the DNS cases. Regarding $\text{Re}_H = 5{,}600$, the references BPRM DNS/LES lie within the bandwidth, which means that they are in agreement with the present DNS within the available accuracy. The experimental reattachment length and the case HRLES 5600 lie outside of the bandwidth, rendering other sources of error than the statistical averaging error relevant. Regarding $\text{Re}_H = 10{,}595$, it is apparent that the bandwidths of HRLES 10595 and DNS 10595 overlap significantly, such that the remaining difference in the reattachment length may be a statistical averaging error. This result mirrors the excellent agreement of the data discussed in the previous section. The cases BPRM LES and FMRTL R2 lie well outside of the error bandwidth, indicating that the differences to the DNS are other sources of error, such as underresolution. The experimental reference also lies significantly outside of the uncertainty range. The cases BPP, FMRTL R1, and DM lie inside or near the bandwidth, such that these cases are in agreement with the DNS within the available accuracy, which does not exclude other sources of error, however.

It may be concluded from this discussion that the reattachment length of this flow is very sensitive to insufficient sampling. This fact may be explained by the small slope of the c_f curves in Figures 4.9 and 4.11 near the reattachment points. While the statistical error after 61 flow-through times of $\pm 0.09H$ and $\pm 0.06H$ may seem large as reference data, the reduction by a factor of two for the higher Reynolds number would necessitate an additional sampling time of 183 flow-through times due to the slow convergence with $\sim 1/\sqrt{\Delta t_{\text{aver}}/T^*}$, which would re-

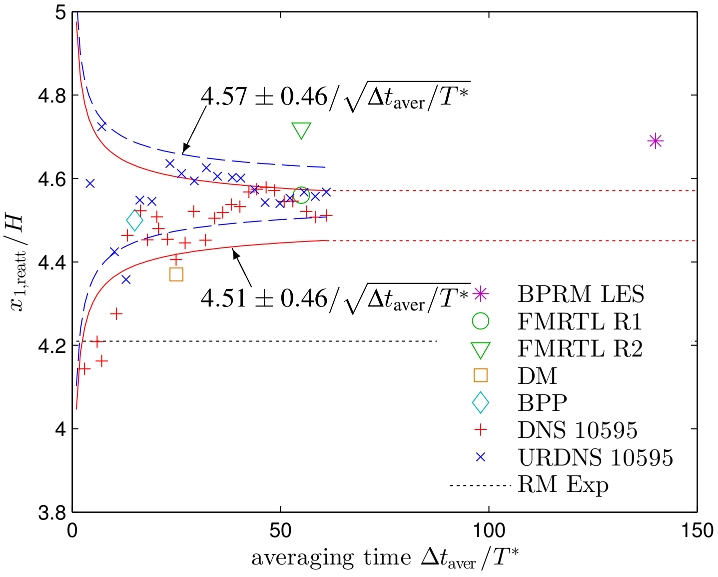


Figure 4.15.: $\text{Re}_H = 10{,}595$: Analysis of the statistical averaging error of the reattachment length $x_{1,\text{reatt}}/H$. The error bandwidth is given as $x_{1,\text{reatt}}/H \pm \text{const}/\sqrt{\Delta t_{\text{aver}}/T^*}$ and yields an estimation of the error after 61 flow-through times of $\pm 0.06H$.

quire additional 19 million CPU core hours and three months of computation time, as well as 400,000kWh of electricity. Furthermore, sampling times as high as 1,000 flow-through times suggested in [131] will not be available in the context of DNS at the present Reynolds numbers for several years, nor are such sampling times desirable in the application of benchmark simulations. As a result, researchers have to be made aware of the presence of statistical errors and they should be taken into account when using the DNS data as a reference.

4.3.5. Implicit Large-Eddy Simulation

The high computational cost of the DNS makes an application of this approach to most industrial flows infeasible. Computing only the larger, inhomogeneous eddies and modeling the smaller, homogeneous turbulent motions promises a similar level of accuracy for the present benchmark example at a drastically reduced numerical effort. In this section, we assess the idea of ILES using the same high-order DG scheme and computational setup as employed for the highly resolved cases. We perform a detailed h/p-refinement study in order to analyze the influence of the polynomial degree and the rate of convergence to the DNS.

Regarding the h-refinement study, we consider meshes of 32×16×16 (coarse mesh), 64×32×32 (medium mesh), and 128×64×64 (fine mesh) cells, all with $k = 4$, since this polynomial degree has given accurate results in the context of turbulent channel flow and can be seen as a compromise between accuracy and time to solution. With respect to p-refinement, we take the medium mesh, 64×32×32, and vary the polynomial degree through $k = \{3, 4, 5, 6\}$. In order to achieve a full h/p analysis, we present further results of the coarse and fine mesh with all four

Table 4.5.: ILES h/p-refinement study at $\mathrm{Re}_H = 10{,}595$: Simulation cases and reference data, including polynomial degree k, order of accuracy $(k+1)$, total number of grid points N, Mach number Ma/incompressible (inc.), maximum value of $\Delta y^+_{1,\mathrm{GL}}$ at the upper wall (near-wall resolution) and h/η (bulk flow resolution), sampling time Δt_{aver} in number of flow-through times T^*, separation and reattachment length $x_{1,\mathrm{sep}}$ and $x_{1,\mathrm{reatt}}$. The number of nodes of the present DG solver is obtained by the number of cells in each spatial direction times the number of nodes per cell $(k+1)$ in each direction.

ref.	grid points	k	order	N	Ma	$\max\left(\Delta y^+_{1,\mathrm{GL}}\right)$	$\max\left(\frac{h}{\eta}\right)$	$\frac{\Delta t_{\mathrm{aver}}}{T^*}$	$\frac{x_{1,\mathrm{sep}}}{H}$	$\frac{x_{1,\mathrm{reatt}}}{H}$
-	128×64×64	3	4	0.52M	inc.	5.69	17.3	61	0.27	3.65
-	160×80×80	4	5	1.0M	inc.	3.28	15.4	61	0.23	3.90
-	192×96×96	5	6	1.8M	inc.	2.21	13.8	61	0.22	4.19
-	224×112×112	6	7	2.8M	inc.	1.60	12.8	61	0.21	4.19
-	256×128×128	3	4	4.2M	inc.	2.62	11.7	61	0.21	3.81
-	320×160×160	4	5	8.2M	inc.	1.66	0.96	61	0.21	4.22
-	384×192×192	5	6	14.2M	inc.	1.30	8.48	61	0.21	4.24
-	448×224×224	6	7	22.5M	inc.	1.09	7.55	61	0.21	4.38
-	512×256×256	3	4	33.6M	inc.	1.75	6.17	61	0.20	4.23
-	640×320×320	4	5	65.5M	inc.	1.10	5.36	61	0.19	4.36
-	768×384×384	5	6	113M	inc.	0.86	4.65	61	0.19	4.57
-	896×448×448	6	7	180M	inc.	0.66	4.07	61	0.20	4.51
[28]	160×80×80[2]	4	-	1.0M	0.1	-	-	56	-	3.62[3]
[28]	224×112×112[2]	6	-	2.8M	0.1	-	-	56	-	4.18[3]
[178]	256×128×128	3	4	4.2M	0.1	-	-	-	-	3.9

polynomial degrees, which are only evaluated by comparing their reattachment length as a global measure of the error. The total number of grid points for these meshes is obtained by taking the number of cells times $k+1$ in each dimension. All simulation cases are listed in Table 4.5. Note that the two finest cases are equal to the DNS 10595 and HRLES 10595 cases presented in the previous sections. The grid stretching is adjusted in order to achieve a compromise between near-wall and bulk resolution. Representative mesh parameters are also included in Table 4.5, given as the maximum value of Δy^+_1 at the upper wall, and the maximum ratio of h/η occurring in the shear layer similar to Figure 4.7. The results are compared to the case DNS 10595 discussed in the previous sections. We additionally compare the results of the reattachment length to previous ILES by [28] and LES (based on the WALE model) by [178] using compressible high-order DG methods with comparable spatial discretizations, which are also included in Table 4.5.

---

[2] Private communication with Thomas Bolemann

[3] Extracted from figure 9 in [28]

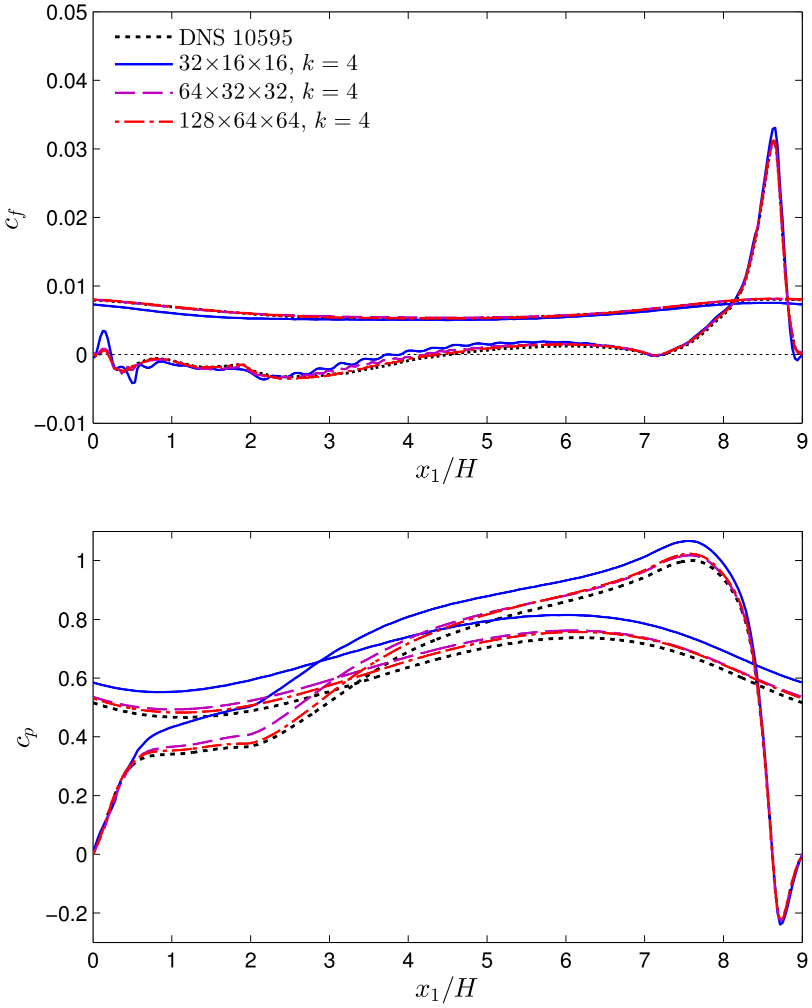


Figure 4.16.: ILES h-refinement study at $\mathrm{Re}_H = 10{,}595$: Skin friction (top) and pressure (bottom) coefficients. The shallower curves correspond to the upper wall.

4.3.5.1. ILES h/p-Refinement Study at $\mathrm{Re}_H = 10{,}595$: Skin Friction and Pressure Coefficients

We present the skin friction and pressure coefficients for the h-refinement study in Figure 4.16 and for the p-refinement study in Figure 4.17. Regarding the h-refinement, the skin friction on the upper wall and on the windward side of the hill exhibits differences to the DNS for the coarse mesh only and is converged for the medium and fine mesh. On the lee side of the hill, the coarse mesh yields significant deviations from the reference, whereas the medium and fine meshes agree well with the DNS. In the recirculation zone between $x_1/H = 3$ and $x_1/H = 5$, the skin friction coefficient is overpredicted by the coarse and medium simulations. In addition, the coarse case exhibits characteristic waves, which are due to the spatial discretization. The waves may be observed with reduced amplitude for the medium mesh and are not visible for the fine mesh. Such behavior was previously described for this flow in [28] within a compressible DG scheme and is also visible in the skin friction curves of the original publication of the case

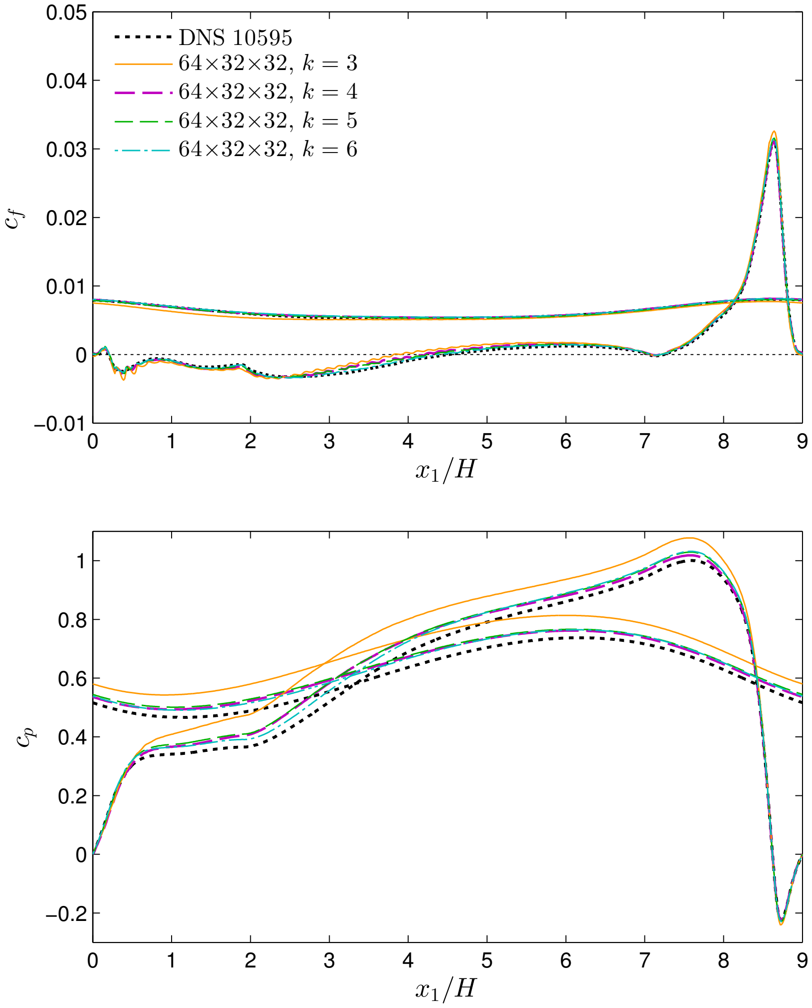


Figure 4.17.: ILES p-refinement study at $\mathrm{Re}_H = 10{,}595$: Skin friction (top) and pressure (bottom) coefficients. The shallower curves correspond to the upper wall.

DM [62], which was discussed in the previous sections, despite the relatively high resolution. It was found in [28] that this effect is not related to insufficient statistical averaging. It is interesting to see that [178] presents smooth curves in the simulations using the WALE subgrid model, and it should be further investigated if this is due to the particular DG formulation used in that study, for example due to differences in the imposition of the Dirichlet boundary conditions, or due to the eddy viscosity subgrid model. The extension of the present numerical method to DES (including an eddy viscosity subgrid model) in Chapter 5 shows these waves as well, but with a smaller magnitude. With respect to the pressure coefficient, the curves converge quickly, with significant deviations for the coarse case, moderate differences for the medium case and minor errors for the fine case.

The p-refinement shows generally a similar trend as the h-refinement, and exhibits a smaller bandwidth regarding the error level. The skin friction of the case $k = 5$ shows minor errors and the case $k = 6$ is almost indistinguishable from the DNS. The waviness of the skin friction near

Table 4.6.: ILES h/p-refinement study at $\mathrm{Re}_H = 10{,}595$: Summary of reattachment lengths. Data is redundant with Table 4.5, and reprinted using number of cells N_{ei} per spatial direction i and polynomial degree k. The statistical averaging uncertainty is approximately $\pm 0.06H$ in all cases.

$N_{e1} \times N_{e2} \times N_{e3} \setminus k$	3	4	5	6
32×16×16	3.65	3.90	4.19	4.19
64×32×32	3.81	4.22	4.24	4.38
128×64×64	4.23	4.36	4.57	4.51

the recirculation zone is also present with a smaller amplitude, but vanishes incrementally for the higher degrees. The pressure curves improve significantly from $k = 3$ to $k = 4$, but the finest case still differs from the DNS. Comparing the case $k = 6$ on the medium mesh across the plots with the fine mesh using $k = 4$, one observes a similar error level. This fact is remarkable, since the fine mesh with $k = 4$ has almost three times the number of DOFs compared to the medium mesh with $k = 6$.

The reattachment lengths of all cases of the h/p-refinement analysis are given in Table 4.6. It is noted that the reattachment length is consistently underpredicted by the coarse meshes, in accordance with the overprediction of the skin friction coefficient in that region, and most cases with a reasonably fine LES grid lie in the area around $x_1/H = 4.2$. While this result is quite good, it also indicates that a further improvement of the ILES approach is possible. The ILES and LES references [28, 178] listed in Table 4.5, which also use high-order DG methods, show almost the same reattachment lengths as the simulation case with the corresponding mesh and polynomial degree.

4.3.5.2. ILES h/p-Refinement Study at $\mathrm{Re}_H = 10{,}595$: Velocity Statistics

The cases considered in the h/p-refinement study are further compared via profiles at ten streamwise stations in Figures 4.18 and 4.19. Regarding the streamwise velocity, the agreement with the DNS is generally very good. Solely near the lower wall in the region of the reattachment point, the velocity is overpredicted by the coarser simulation cases in accordance with the shorter recirculation bubble observed in these cases. Regarding the vertical velocity, the coarser cases exhibit differences to the DNS, but the simulations show a clear trend to convergence with increasing resolution. With respect to the RSS, the errors are on a very low level throughout and deviations from the DNS are only significant for the coarsest case in the h and p-refinement study, respectively. The TKE profiles agree well with the DNS, again except for the coarsest cases, which show some deviations. In addition, the profiles of the coarsest simulations exhibit pronounced ticks at the element boundaries, which originate from the discontinuity present in the solution at the cell boundaries and have been reported for turbulent channel flow as well (see Section 4.2). This issue occurs since the velocity statistics include data points on the cell boundaries where the contributions from two neighboring cells are considered. The study in [178] found that this issue is considerably improved if an eddy viscosity subgrid model is included; this observation is confirmed in Section 5.5 in the context of DES. However, the ticks vanish with increasing resolution and the case DNS 10595 is smooth.

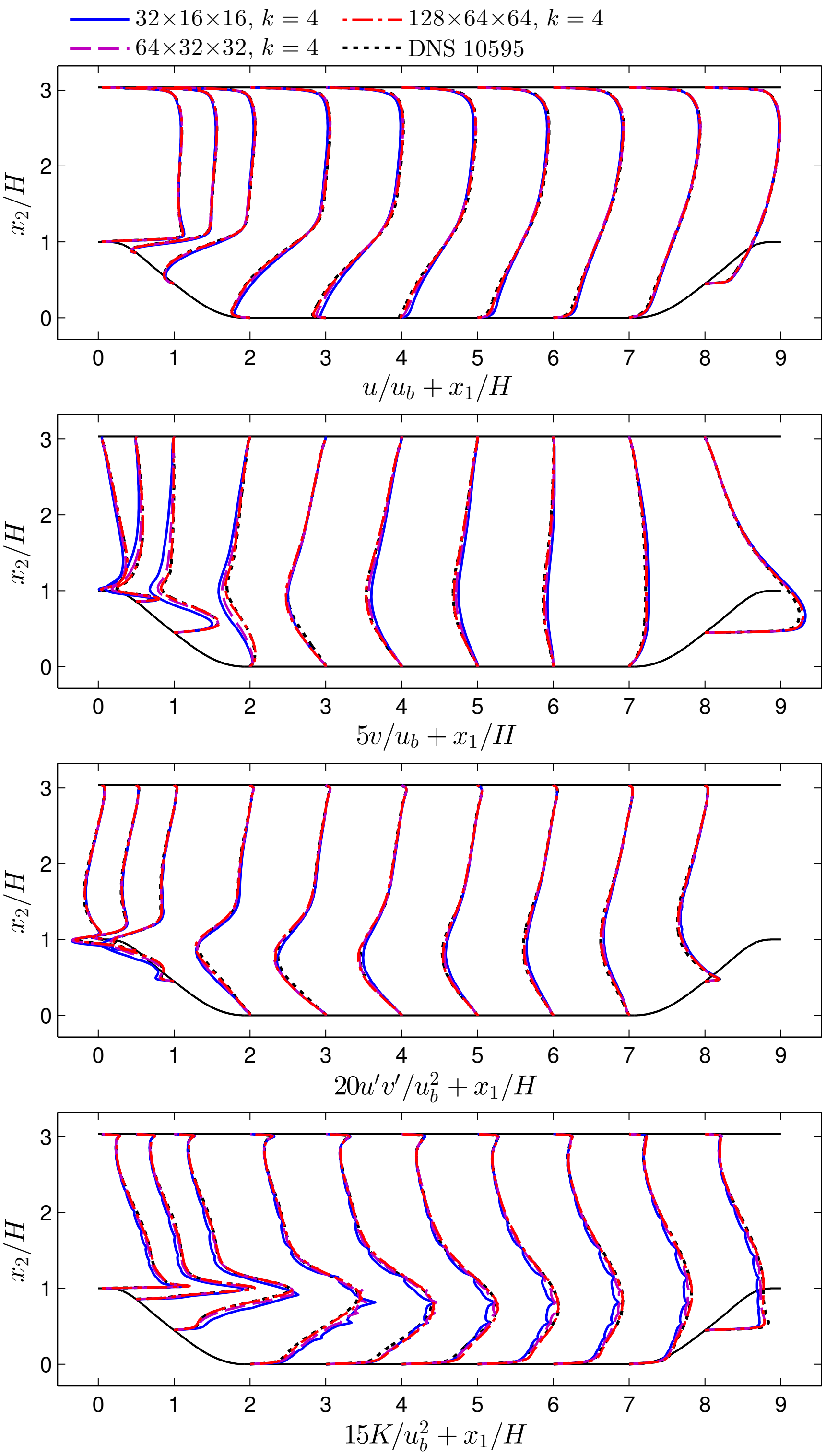


Figure 4.18.: ILES h-refinement study at $\mathrm{Re}_H = 10{,}595$: Profiles of mean velocity in streamwise and vertical direction u and v, RSS $u'v'$, as well as TKE K (from top to bottom).

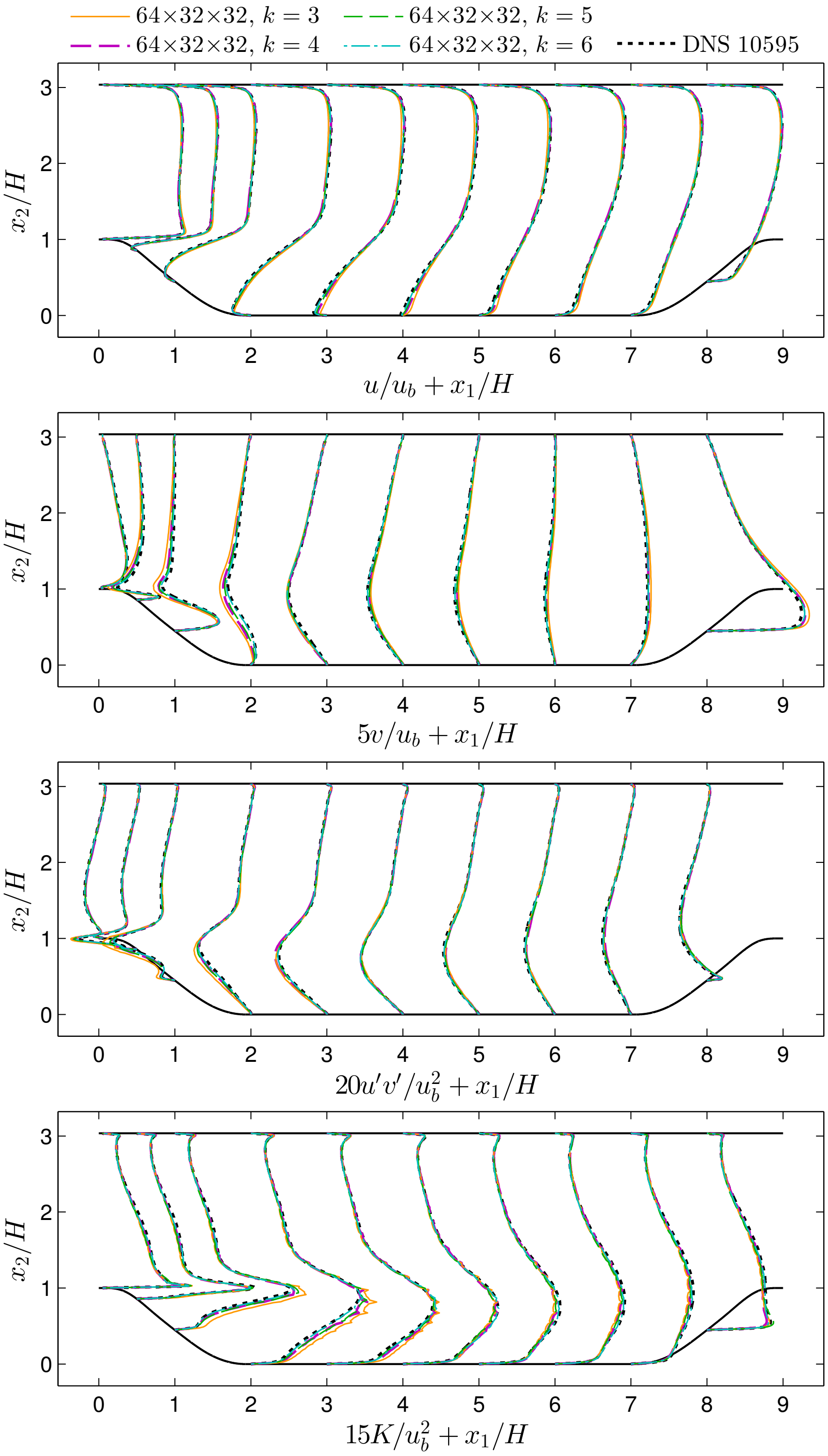


Figure 4.19.: ILES p-refinement study at $\mathrm{Re}_H = 10{,}595$: Profiles of mean velocity in streamwise and vertical direction u and v, RSS $u'v'$, as well as TKE K (from top to bottom).

The present analysis of ILES using the high-order scheme has exhibited a good level of accuracy with the DNS simulations and has shown that the present DG scheme can be used to predict challenging flows such as the periodic hill flow via underresolved simulations. The use of high polynomial degrees yields better results than a refined mesh regarding the required number of DOFs. The scheme may be further improved considering the prediction of the reattachment length, the waviness of the skin friction at the lower boundary, as well as the ticks present in the solution of the TKE.

4.4. Summary

In this chapter, we have applied the present high-order semi-explicit DG code to DNS and ILES of turbulent channel flow and flow over periodic hills. The numerical method is very well suitable for DNS due to its high-order accuracy, which yields a high resolution power with relatively few DOFs. These DNS capabilities were used to compute new reference data for the periodic hill flow at the hill-Reynolds numbers $\mathrm{Re}_H = 5{,}600$ and $\mathrm{Re}_H = 10{,}595$. These reference results have been made available on a public open source repository in [148] and will be extensively used as reference data in the remainder of this thesis. Finally, we have shown that the present numerical method may be employed for ILES as well. We have performed detailed h-refinement for the turbulent channel and h/p-refinement for the periodic hill flow, and have obtained an excellent agreement with the DNS results in a wide range of spatial resolution.

5

Extension to RANS and DES

The aim of this chapter is to extend the high-order DG scheme presented in Chapter 3 to the RANS equations (see Section 2.3.3) and to perform wall-modeling DES (see Section 2.3.4) using this scheme. Both RANS and DES are of high relevance in industrial applications, due to their low cost in comparison to LES. The RANS equations are particularly cheap to compute, as they exhibit very large time scales (see Section 2.3.3), enabling large time steps in implicit schemes, and many applications even allow the use of a stationary solver. DES is increasingly used in research and industry, as it provides an approach with similar accuracy as LES in separated flows, however with reduced computational cost of the near-wall layer, and also allows for the analysis of aeroacoustical noise sources in the flow. In addition, the computational requirements have become available for example in the automotive industry [24, 241].

In order to extend the present velocity-correction scheme to these simulation approaches, the SA turbulence model [242] is considered in an additional step. The SA model is popular due to its simplicity with only one additional transport equation while yielding accurate results in many applications [166]. The present method requires minimal additional implementation effort if it is implemented as an extension of the velocity-correction scheme, and does not require any nonlinear iterations in the algorithm. This explicit approach renders the method particularly efficient for DES. We begin in this chapter with a literature overview of applications of the DG method to the SA model. Subsequently, the governing equations are recited and the velocity-correction approach of Section 3.2 is extended by including the SA equation, see Section 5.2. Details on the spatial discretization are given in Section 5.3. Numerical results are presented for RANS in Section 5.4 and DES in Section 5.5. The work presented in Sections 5.2 to 5.4 is based on Krank et al. [152].

5.1. Literature Review and Motivation

During the past ten years, a large number of implementations of the SA model for the DG and HDG method have been proposed, for example in [35, 53, 55, 166, 185, 270] (DG) and [194] (HDG). All of these methods have been developed within implicit time integration schemes and most approaches consider the compressible Navier–Stokes equations [35, 55, 166, 185, 194, 270] while [53] considers the incompressible Navier–Stokes equations. Solely [166] compares the performance of an implicit and explicit method and comes to the conclusion that the time step restrictions of an explicit method for RANS are significant. Issues with negative values of the working variable of the SA model are the primary object of study in all of these publications. These problems occur primarily at the outer edge of the boundary layer, where the variable sees a steep decrease in magnitude in order to become zero in the laminar freestream. The issues arise as this outer edge of the boundary layer is usually underresolved, which can result in a negative eddy viscosity and thus an unstable method. While these problems have been reported for other discretization approaches as well [7, 35], high-order methods come along with less numerical diffusion and are therefore more exposed. The proposed methods include regularization [185], clipping of the working variable [53, 166], and modification of the definition of the SA model [7]. In this work, we do not examine these issues in detail and a simple clipping of the working variable is employed [53, 166].

The focus of this work lies in the investigation of another algorithmic approach: The SA model is frequently implemented in the DG method with the aim of computing eddy-resolving DES-type simulations [244], for which high-order DG methods are particularly suited; see, e.g., [267]. Considering an eddy-resolving approach, the time scales of the simulation have to be quite small in order to resolve the temporal scales of the turbulent motions sufficiently, however. Therefore, the implicit schemes, which are undoubtedly more efficient for RANS than explicit ones due to time step restrictions [166, 241], are not necessarily more efficient for DES. On the contrary, due to the larger cells in the near-wall region used in DES, one would expect that the allowable time step size of the Navier–Stokes steps is much larger in comparison to LES for an explicit scheme. This becomes in particular relevant if the time step restrictions from an explicit diffusive term in the SA model can be decoupled from the Navier–Stokes terms.

5.2. Incompressible Navier–Stokes Equations and the Spalart–Allmaras Model

5.2.1. The Spalart–Allmaras Model

The incompressible Navier–Stokes equations (2.1) and (2.2) are extended by an additional equation, which provides a closure model via the Boussinesq approximation (see Section 2.3.3), yielding

$$\begin{aligned}
&\nabla \cdot \boldsymbol{u} && = 0 && \text{in } \Omega \times [0, \mathcal{T}], && (5.1)\\
&\frac{\partial \boldsymbol{u}}{\partial t} + \nabla \cdot \left(\boldsymbol{\mathcal{F}}^{c}(\boldsymbol{u}) + p \boldsymbol{\mathcal{I}} - \boldsymbol{\mathcal{F}}^{\nu+\nu_t}(\boldsymbol{u}) \right) && = \boldsymbol{f} && \text{in } \Omega \times [0, \mathcal{T}], && (5.2)\\
&\frac{\partial \widetilde{\nu}}{\partial t} + \nabla \cdot \left(\widetilde{\boldsymbol{\mathcal{F}}}^{c}(\boldsymbol{u}, \widetilde{\nu}) - \widetilde{\boldsymbol{\mathcal{F}}}^{\widetilde{\nu}}(\widetilde{\nu}) \right) && = Q(\boldsymbol{u}, \widetilde{\nu}) && \text{in } \Omega \times [0, \mathcal{T}]. && (5.3)
\end{aligned}$$

In addition to the familiar quantities, $\boldsymbol{u}$, p, $\boldsymbol{f}$, Ω, and $\mathcal{T}$ according to Section 2.1.1, a new variable is introduced, $\widetilde{\nu}$, which is the eddy-viscosity-like working variable of the SA model. The convective flux of the momentum equation is defined as in Section 2.1.1 and the viscous flux takes into account the model term as

$$\boldsymbol{\mathcal{F}}^{c}(\boldsymbol{u}) = \boldsymbol{u} \otimes \boldsymbol{u}, \qquad \boldsymbol{\mathcal{F}}^{\nu+\nu_t}(\boldsymbol{u}) = 2(\nu + \nu_t)\boldsymbol{\epsilon}(\boldsymbol{u})$$

with the eddy viscosity $\nu_t = \widetilde{\nu} f_{v1}$, where f_{v1} is a quantity defined through the SA model. The convective as well as diffusive fluxes of the SA equation are given as

$$\widetilde{\boldsymbol{\mathcal{F}}}^{c}(\boldsymbol{u}, \widetilde{\nu}) = \boldsymbol{u}\widetilde{\nu}, \qquad \widetilde{\boldsymbol{\mathcal{F}}}^{\widetilde{\nu}}(\widetilde{\nu}) = \frac{\nu + \widetilde{\nu}}{c_{b3}} \nabla \widetilde{\nu},$$

where c_{b3} is a model constant. The source term $Q(\boldsymbol{u}, \widetilde{\nu})$ is considered in the following formulation:

$$Q(\boldsymbol{u}, \widetilde{\nu}) = c_{b1}\widetilde{S}\widetilde{\nu} + \frac{c_{b2}}{c_{b3}}\nabla\widetilde{\nu} \cdot \nabla\widetilde{\nu} - c_{w1} f_w \left(\frac{\widetilde{\nu}}{y}\right)^2 \tag{5.4}$$

using the expressions

$$\begin{aligned}
\widetilde{S} &= S + \frac{\widetilde{\nu}}{\kappa^2 y^2} f_{v2}, & \boldsymbol{\Omega} &= \frac{1}{2}(\nabla\boldsymbol{u} - (\nabla\boldsymbol{u})^T), & S &= \sqrt{2\boldsymbol{\Omega} : \boldsymbol{\Omega}}, \\
\chi &= \frac{\widetilde{\nu}}{\nu}, & f_{v1} &= \frac{\chi^3}{\chi^3 + c_{v1}^3}, & f_{v2} &= 1 - \frac{\chi}{1 + \chi f_{v1}}, \\
f_w &= g\left(\frac{1 + c_{w3}^6}{g^6 + c_{w3}^6}\right)^{1/6}, & g &= r + c_{w2}(r^6 - r), & r &= \frac{\widetilde{\nu}}{\widetilde{S}\kappa^2 y^2},
\end{aligned}$$

and constants

$$\begin{aligned}
c_{b1} &= 0.1355, & c_{b2} &= 0.622, & c_{b3} &= 2/3, \\
c_{v1} &= 7.1, & \kappa &= 0.41, & & \\
c_{w1} &= \frac{c_{b1}}{\kappa^2} + \frac{1 + c_{b2}}{c_{b3}}, & c_{w2} &= 0.3, & c_{w3} &= 2.
\end{aligned}$$

In the nonphysical case of $\widetilde{\nu} < 0$, we set $Q(\boldsymbol{u}, \widetilde{\nu}) = 0$ [53] as well as $\nu_t = 0$ [53, 166].

The initial conditions are specified at $t = 0$ as

$$\boldsymbol{u}(t = 0) = \boldsymbol{u}_0 \quad \text{in } \Omega \quad \text{and} \tag{5.5}$$

$$\widetilde{\nu}(t = 0) = \widetilde{\nu}_0 \quad \text{in } \Omega. \tag{5.6}$$

In the context of the SA model, periodic as well as no-slip Dirichlet boundary conditions on solid walls $\partial\Omega^D = \partial\Omega$ are considered that close the problem with

$$\boldsymbol{u} = \boldsymbol{g}_{\boldsymbol{u}} = \boldsymbol{0} \quad \text{on } \partial\Omega^D \quad \text{and} \tag{5.7}$$

$$\widetilde{\nu} = g_{\widetilde{\nu}} = 0 \quad \text{on } \partial\Omega^D. \tag{5.8}$$

5.2.2. Extension of the Velocity-Correction Scheme

The governing equations (5.1) to (5.3) are integrated in time by extending the velocity-correction scheme used in Chapter 3 as follows. The Navier–Stokes equations are treated using the exact same temporal scheme as before and the SA equation is integrated in time explicitly in order to avoid nonlinear iterations.

Explicit Spalart–Allmaras Step. The semi-discrete form of the SA equation becomes

$$\begin{aligned}
&\frac{\gamma_0 \widetilde{\nu}^{n+1} - \sum_{i=0}^{J-1}\left(\alpha_i \widetilde{\nu}^{n-i}\right)}{\Delta t} = \\
&\qquad -\sum_{i=0}^{J-1} \beta_i \left(\nabla \cdot \left(\widetilde{\boldsymbol{\mathcal{F}}}^c\left(\boldsymbol{u}^{n-i}, \widetilde{\nu}^{n-i}\right) - \widetilde{\boldsymbol{\mathcal{F}}}^\nu\left(\widetilde{\nu}^{n-i}\right)\right) - Q\left(\boldsymbol{u}^{n-i}, \widetilde{\nu}^{n-i}\right)\right),
\end{aligned} \tag{5.9}$$

with the BDF and EX schemes already used in the velocity-correction scheme, either using constant time step sizes according to Table 3.1, or employing the coefficients α_i, β_i, and γ_0 allowing adaptive time stepping according to Sections 3.3.3 and 5.3.2. The explicit SA step yields $\widetilde{\nu}^{n+1}$ at the new time step.

Explicit Convective Step. The temporal integration of the Navier–Stokes equations follows the method presented in Section 3.2 and is repeated for completeness. The convective term is integrated explicitly, yielding the first intermediate velocity $\hat{\boldsymbol{u}}$:

$$\frac{\gamma_0 \hat{\boldsymbol{u}} - \sum_{i=0}^{J-1} \left(\alpha_i \boldsymbol{u}^{n-i}\right)}{\Delta t} = -\sum_{i=0}^{J-1} \beta_i \nabla \cdot \boldsymbol{\mathcal{F}}^{c}\left(\boldsymbol{u}^{n-i}\right) + \boldsymbol{f}^{n+1}. \tag{5.10}$$

Pressure Poisson Equation and Projection. The pressure is subsequently computed by solving a Poisson equation, given as

$$-\nabla^2 p^{n+1} = -\frac{\gamma_0}{\Delta t} \nabla \cdot \hat{\boldsymbol{u}}. \tag{5.11}$$

The high-order boundary conditions in Equation (3.4) may be modified by taking into account the boundary conditions $\boldsymbol{g}_{\boldsymbol{u}} = \boldsymbol{0}$ and $g_{\widetilde{\nu}} = 0$ according to:

$$\nabla p^{n+1} \cdot \boldsymbol{n} = -\left(\sum_{i=0}^{J-1} \beta_i \left(\nabla \cdot \boldsymbol{\mathcal{F}}^{c}\left(\boldsymbol{u}_h^{n-i}\right) + \nu \nabla \times \boldsymbol{\omega}^{n-i}\right) - \boldsymbol{f}^{n+1}\right) \cdot \boldsymbol{n} \tag{5.12}$$

on $\partial\Omega^D$. No modifications are made to the projection equation (3.5), which reads

$$\hat{\hat{\boldsymbol{u}}} = \hat{\boldsymbol{u}} - \frac{\Delta t}{\gamma_0} \nabla p^{n+1} \tag{5.13}$$

and results in the second intermediate velocity $\hat{\hat{\boldsymbol{u}}}$.

Implicit Viscous Step. Finally, the viscous term takes into account the eddy viscosity and yields the velocity $\boldsymbol{u}^{n+1}$ at time level t^{n+1} in a Helmholtz-like equation, which is given as

$$\frac{\gamma_0}{\Delta t} \left(\boldsymbol{u}^{n+1} - \hat{\hat{\boldsymbol{u}}}\right) = \nabla \cdot \boldsymbol{\mathcal{F}}^{\nu+\nu_t}\left(\boldsymbol{u}^{n+1}\right). \tag{5.14}$$

The system is closed with boundary conditions on the no-slip walls

$$\begin{align} \boldsymbol{u}^{n+1} &= \boldsymbol{0} \quad \text{on } \partial\Omega^D \quad \text{and} \tag{5.15} \\ \widetilde{\nu}^{n+1} &= 0 \quad \text{on } \partial\Omega^D. \tag{5.16} \end{align}$$

5.3. Spatial Discretization

The spatial discretization of the incompressible Navier–Stokes equations is similar to Chapter 3 and introduces several minor modifications, which guarantee the stability of the approach. The spatial discretization of the SA equation employs similar methods.

5.3.1. Galerkin Formulation

The following variational formulations are derived analogous to Section 3.3.2 by multiplication of the strong forms (5.9) to (5.16) with an appropriate weighting function $\mu_h \in \mathcal{V}_h^{\widetilde{\nu}}$, $\boldsymbol{v}_h \in \mathcal{V}_h^{\boldsymbol{u}}$, or $q_h \in \mathcal{V}_h^{p}$ and integration over one element volume. Regarding the function spaces for the primary variable of the SA model, we take the same space as for the pressure according to $\mathcal{V}_h^{\widetilde{\nu}} = \mathcal{V}_h^{p}$, defined in Section 3.3.1. Upon partial integration, suitable fluxes are specified in order to guarantee a stable numerical method.

Explicit Spalart–Allmaras Step. The variational formulation of the SA step becomes

$$
\begin{aligned}
&\left(\mu_h, \frac{\gamma_0 \widetilde{\nu}_h - \sum_{i=0}^{J-1}\left(\alpha_i \widetilde{\nu}_h^{n-i}\right)}{\Delta t}\right)_{\Omega_e} \\
&= -\sum_{i=0}^{J-1} \beta_i \Bigg(-\left(\nabla \mu_h, \widetilde{\boldsymbol{\mathcal{F}}}^{c}\left(\boldsymbol{u}_h^{n-i}, \widetilde{\nu}_h^{n-i}\right)\right)_{\Omega_e} + \left(\mu_h, \widetilde{\boldsymbol{\mathcal{F}}}^{c*}\left(\boldsymbol{u}_h^{n-i}, \widetilde{\nu}_h^{n-i}\right) \cdot \boldsymbol{n}_\Gamma\right)_{\partial\Omega_e} \\
&+ \left(\nabla \mu_h, \widetilde{\boldsymbol{\mathcal{F}}}^{\widetilde{\nu}}\left(\widetilde{\nu}_h\right)\right)_{\Omega_e} - \left(w \widetilde{\boldsymbol{\mathcal{F}}}^{\widetilde{\nu}}\left(\mu_h\right), \llbracket \widetilde{\nu}_h \rrbracket\right)_{\partial\Omega_e} - \left(\mu_h, \widetilde{\boldsymbol{\mathcal{F}}}^{\widetilde{\nu}*}\left(\widetilde{\nu}_h\right)\right)_{\partial\Omega_e} \\
&- \left(\mu_h, Q\left(\boldsymbol{u}_h^{n-i}, \widetilde{\nu}_h^{n-i}\right)\right)_{\Omega_e} \Bigg).
\end{aligned}
\tag{5.17}
$$

Herein, the local Lax–Friedrichs numerical flux is employed for the convective term as for the Navier–Stokes equations in Equation (3.8), yielding

$$
\widetilde{\boldsymbol{\mathcal{F}}}^{c*}\left(\boldsymbol{u}_h^{n-i}, \widetilde{\nu}_h^{n-i}\right) = \{\{\widetilde{\boldsymbol{\mathcal{F}}}^{c}\left(\boldsymbol{u}_h^{n-i}, \widetilde{\nu}_h^{n-i}\right)\}\} + \widetilde{\Lambda}/2 \llbracket \widetilde{\nu}_h^{n-i} \rrbracket,
\tag{5.18}
$$

where $\widetilde{\Lambda} = \max(\widetilde{\lambda}^-, \widetilde{\lambda}^+)$ represents the largest eigenvalue of the flux Jacobian across the element interface with

$$
\begin{aligned}
\widetilde{\lambda}^- &= \max_j \left| \widetilde{\lambda}_j \left(\frac{\partial \widetilde{\mathcal{F}}\left(\boldsymbol{u}_h^{-,n-i}, \widetilde{\nu}\right) \cdot \boldsymbol{n}_\Gamma}{\partial \widetilde{\nu}} \right) \right| = |\boldsymbol{u}_h^{-,n-i} \cdot \boldsymbol{n}_\Gamma| \text{ and} \\
\widetilde{\lambda}^+ &= \max_j \left| \widetilde{\lambda}_j \left(\frac{\partial \widetilde{\mathcal{F}}\left(\boldsymbol{u}_h^{+,n-i}, \widetilde{\nu}\right) \cdot \boldsymbol{n}_\Gamma}{\partial \widetilde{\nu}} \right) \right| = |\boldsymbol{u}_h^{+,n-i} \cdot \boldsymbol{n}_\Gamma|.
\end{aligned}
\tag{5.19}
$$

The diffusive term is discretized by the SIPG method in the same way as the pressure Laplacian (3.11) and the viscous terms (3.18). However, the term is modified to take into account harmonic weighting [36] of the discontinuous diffusivity by specifying weights for the averaging operator defined in Section 3.3.1 using the material parameter of the diffusive term:

$$
w^- = \frac{\nu + \widetilde{\nu}_h^+}{2\nu + \widetilde{\nu}_h^- + \widetilde{\nu}_h^+}, \qquad w^+ = \frac{\nu + \widetilde{\nu}_h^-}{2\nu + \widetilde{\nu}_h^- + \widetilde{\nu}_h^+},
\tag{5.20}
$$

which renders the formulation stable for large discontinuities in $\widetilde{\nu}_h$ as well. The diffusive flux then becomes

$$
\widetilde{\boldsymbol{\mathcal{F}}}^{\widetilde{\nu}*}\left(\widetilde{\nu}_h^{n-i}\right) = \{\{\widetilde{\boldsymbol{\mathcal{F}}}^{\widetilde{\nu}}\left(\widetilde{\nu}_h^{n-i}\right)\}\} - \tau_{\mathrm{IP}} \frac{2\left(\nu + \widetilde{\nu}_h^-\right)\left(\nu + \widetilde{\nu}_h^+\right)}{c_{b3}\left(2\nu + \widetilde{\nu}_h^- + \widetilde{\nu}_h^+\right)} \llbracket \widetilde{\nu}_h^{n-i} \rrbracket.
\tag{5.21}
$$

The interior penalty parameter τ_{IP} according to Equations (3.14) and (3.15) is used. On external boundaries, we define $\widetilde{\nu}_h^+ = -\widetilde{\nu}_h^-$, $\nabla \widetilde{\nu}_h^+ = \nabla \widetilde{\nu}_h^-$ and $\boldsymbol{u}_h^+ = -\boldsymbol{u}_h^-$, which are in agreement with no-slip Dirichlet boundary conditions.

Explicit Convective Step. The weak form of the convective step is exactly equivalent to Equation (3.8) and the associated local Lax–Friedrichs flux, so it is not repeated here.

Poisson Equation and Projection. The second derivatives present in the pressure Poisson equation are again discretized by use of the SIPG method. The formulation of the pressure and projection step is equivalent to the solver in Equations (3.11) to (3.17) with the modifications proposed in *V3c* in Section 3.4.2. In brief, we get

$$\begin{aligned}\left(\nabla q_h, \nabla p_h^{n+1}\right)_{\Omega_e} - \frac{1}{2}\left(\nabla q_h, \llbracket p_h^{n+1}\rrbracket\right)_{\partial\Omega_e} - \left(q_h, \boldsymbol{\mathcal{P}}^* \cdot \boldsymbol{n}_\Gamma\right)_{\partial\Omega_e} \\ = \left(\nabla q_h, \frac{\gamma_0}{\Delta t}\hat{\boldsymbol{u}}_h\right)_{\Omega_e} - \left(q_h, \frac{\gamma_0}{\Delta t}\{\{\hat{\boldsymbol{u}}_h\}\} \cdot \boldsymbol{n}_\Gamma\right)_{\partial\Omega_e}, \quad (5.22)\end{aligned}$$

with the flux function including the high-order boundary conditions for the RANS case defined in Equation (5.12)

$$\boldsymbol{\mathcal{P}}^* = \begin{cases} \{\{\nabla p_h^{n+1}\}\} - \tau_{\mathrm{IP}}\llbracket p_h^{n+1}\rrbracket & \text{on } \partial\Omega_e^\Gamma \text{ and} \\ -\left(\sum_{i=0}^{J-1}\beta_i\left(\nabla\cdot\boldsymbol{\mathcal{F}}^c\left(\boldsymbol{u}_h^{n-i}\right) + \nu\nabla\times\boldsymbol{\omega}_h^{n-i}\right) - \boldsymbol{f}^{n+1}\right) & \text{on } \partial\Omega_e^D. \end{cases} \quad (5.23)$$

Again, the same interior penalty parameter definition as in Equation (3.15) is adopted for τ_{IP} and the vorticity $\boldsymbol{\omega}_h$ is computed as previously. On $\partial\Omega_e^D$ the exterior pressure is further set to $p_h^+ = p_h^-$. According to *V3c* in Section 3.4.2, the projection includes the div-div penalty term and the partially integrated right-hand side, yielding

$$\begin{aligned}\left(\boldsymbol{v}_h, \hat{\hat{\boldsymbol{u}}}_h\right)_{\Omega_e} + \left(\nabla\cdot\boldsymbol{v}_h, \tau_{\mathrm{D}}\nabla\cdot\hat{\hat{\boldsymbol{u}}}_h\right)_{\Omega_e} \\ = \left(\boldsymbol{v}_h, \hat{\boldsymbol{u}}_h\right)_{\Omega_e} + \left(\nabla\cdot\boldsymbol{v}_h, \frac{\Delta t}{\gamma_0}p_h^{n+1}\right)_{\Omega_e} - \left(\boldsymbol{v}_h, \frac{\Delta t}{\gamma_0}\{\{p_h^{n+1}\}\}\boldsymbol{n}_\Gamma\right)_{\partial\Omega_e}. \quad (5.24)\end{aligned}$$

In order to guarantee a stable scheme, we employ an amplification factor of $\zeta_D^* = 10$ in Equation (3.31) for all simulation cases presented with this RANS model. It is noted at this point, that the present RANS equations compute physical flows with dominant convection, but numerically the set of equations is viscous-dominated. Therefore, it may be considered in future work to take into account the viscous stabilization parameter contribution in [197], which was neglected in Equation (3.30). This modification may allow the use of the standard value of $\zeta_D^* = 1$. All DES examples presented in Sections 5.5 and 9.2 are computed with $\zeta_D^* = 1$.

Implicit Viscous Step. The viscous term may in this chapter be discretized either by the SIPG or NIPG method, as mentioned in Section 3.3.2. To this end, we will consider nonpolynomial shape functions in Part II of this thesis in conjunction with the present RANS model. As the standard interior penalty parameter definitions for the SIPG are only available for polynomial shape functions and the NIPG is theoretically stable with any penalty parameter $\tau_{\mathrm{IP}} > 0$ [220], we choose the NIPG at this instance. We get

$$\begin{aligned}\left(\boldsymbol{v}_h, \frac{\gamma_0}{\Delta t}\boldsymbol{u}_h^{n+1}\right)_{\Omega_e} + \left(\boldsymbol{\epsilon}\left(\boldsymbol{v}_h\right), \boldsymbol{\mathcal{F}}^{\nu+\nu_t}\left(\boldsymbol{u}_h^{n+1}\right)\right)_{\Omega_e} - s\left(w\boldsymbol{\mathcal{F}}^{\nu+\nu_t}\left(\boldsymbol{v}_h\right), \llbracket\boldsymbol{u}_h^{n+1}\rrbracket\right)_{\partial\Omega_e} \\ - \left(\boldsymbol{v}_h, \boldsymbol{\mathcal{F}}^{\nu+\nu_t *}\left(\boldsymbol{u}_h^{n+1}\right)\cdot\boldsymbol{n}_\Gamma\right)_{\partial\Omega_e} = \left(\boldsymbol{v}_h, \frac{\gamma_0}{\Delta t}\hat{\hat{\boldsymbol{u}}}_h\right)_{\Omega_e} \quad (5.25)\end{aligned}$$

and take $s = -1$. Again we deal with possible discontinuities present in the viscosity using harmonic weighting according to Equation (5.20) by replacing $\widetilde{\nu}_h$ with ν_t. The interior penalty flux is defined as

$$\boldsymbol{\mathcal{F}}^{\nu+\nu_t *}(\boldsymbol{u}_h^{n+1}) = \{\{\boldsymbol{\mathcal{F}}^{\nu+\nu_t}(\boldsymbol{u}_h^{n+1})\}\} - \tau_{\mathrm{IP}} \frac{2(\nu+\nu_t^+)(\nu+\nu_t^-)}{2\nu+\nu_t^- + \nu_t^+} [\![\boldsymbol{u}_h^{n+1}]\!] \tag{5.26}$$

and we use the same value for τ_{IP} as previously (see Equation (3.15)), as this penalization yields good results in the computational experiments. A drawback of this nonsymmetric formulation is that we may only obtain convergence rates of the baseline solver of order k in the L^2 norm [109], in contrast to $k+1$ of the original version of the solver employing the SIPG in Chapter 3.

With regard to the extensions of the present solver in Part II of this thesis, where very high gradients in the solution just at the wall would result in a distinct violation of the no-slip boundary condition, the results presented herein have been computed with a modified boundary condition. We employ strong velocity Dirichlet boundary conditions on the viscous step, prescribing $\boldsymbol{u}_h^+ = \boldsymbol{u}_h^- = \mathbf{0}$ with $\nabla \boldsymbol{u}_h^+ = \nabla \boldsymbol{u}_h^-$ on the face terms whereas no changes are made in the mass term. A drawback of this modification is that more solver iterations are required in the viscous term. These boundary conditions are therefore modified once again in Chapter 10, where the standard weak boundary conditions are considered, however using a ten times higher NIPG stabilization parameter. This further modification improves the fulfillment of the boundary conditions in a similar way while exhibiting almost no effect on the number of viscous solver iterations.

5.3.2. Time Stepping

The explicit formulation of both the convective and the diffusive term in the SA equation, as well as the convective term in the Navier–Stokes equation, restrict the time step size by the CFL condition (see Section 3.3.3) and the diffusion number D. An adaptive algorithm maximizing the time step size while fulfilling both conditions is described in the following. The time step size Δt^{NS} resulting from a constant Courant number is computed in each step using the adaptive method presented in Section 3.3.3 and applied for the time-advancement of the Navier–Stokes equations. A value of $\mathrm{Cr} = 0.08$ has been used for all RANS examples in this Chapter, where the inverse Jacobian represents a transformation from physical space to a parameter space of unit length $([0,1])$.

The SA equation is additionally subject to the diffusion number D, which is given as

$$\frac{D}{k^3} = \frac{(\nu + \widetilde{\nu}_h^n)\Delta t^{\mathrm{SA}}}{c_{b3} h^2} \tag{5.27}$$

with the cell-wise shortest edge length h and the exponent of k determined experimentally as 3 [71]. A value of $D = 0.03$ has been used for all computations in Section 5.4.1 and $D = 0.02$ for all examples in Section 5.4.2. As the SA equation is much cheaper to evaluate compared to the Navier–Stokes equation, a sub-cycling algorithm is employed in case the latter condition restricts the time-step size, keeping the convective velocity constant during the sub-cycles. The number of sub-cycles is given by $N^{\mathrm{SA}} = \max(1, \lceil \Delta t^{\mathrm{NS}} / \Delta t^{\mathrm{SA}} \rceil)$ and defines the final time step to be used for the SA equation within each Navier–Stokes time step to $\Delta t^{\mathrm{SA}} = \Delta t^{\mathrm{NS}} / N^{\mathrm{SA}}$. The time integrator constants α_i, β_i, and γ_0 for nonequally spaced time intervals may be computed as described in

Section 3.3.3 independently for the SA and the Navier–Stokes steps, and are recalculated when necessary. Sub-cycling counts in the simulations of the present chapter are typically in the range $N^{\mathrm{SA}} = [1, 50]$. If significantly larger N^{SA} occur at higher Reynolds numbers, one may consider to use an implicit implementation of the SA equation. Such an implicit implementation could be included in the velocity correction scheme by solving the SA equation together with the Helmholtz equation in a monolithic way, however requiring nonlinear iterations.

5.3.3. Matrix Formulation

Upon evaluation of all integrals in the Galerkin formulation according to Section 3.5, one arrives at the matrix formulation, which is presented in the following. The matrix formulation of the Navier–Stokes step is very similar to Section 3.4.4, the viscous term is now dependent on the solution of the SA equation, however.

The sub-steps are:

Explicit Spalart–Allmaras Step. The matrix formulation of the explicit SA step becomes

$$\begin{aligned}\gamma_0 \widetilde{\boldsymbol{N}}^{n+1} &= \sum_{i=0}^{J-1} \alpha_i \widetilde{\boldsymbol{N}}^{n-i} \\ &- \Delta t \widetilde{\boldsymbol{M}}^{-1} \sum_{i=0}^{J-1} \beta_i \left(\widetilde{\boldsymbol{F}}^{c} \left(\boldsymbol{U}^{n-i} \right) \widetilde{\boldsymbol{N}}^{n-i} - \widetilde{\boldsymbol{F}}^{\widetilde{\nu}} \left(\widetilde{\boldsymbol{N}}^{n-i} \right) \widetilde{\boldsymbol{N}}^{n-i} - \boldsymbol{Q} \left(\boldsymbol{U}^{n-i}, \widetilde{\boldsymbol{N}}^{n-i} \right) \right) \end{aligned} \tag{5.28}$$

with the eddy viscosity vector $\widetilde{\boldsymbol{N}}$, mass matrix $\widetilde{\boldsymbol{M}}$, convective as well as diffusive flux terms $\widetilde{\boldsymbol{F}}^{c}$ and $\widetilde{\boldsymbol{F}}^{\widetilde{\nu}}$, source terms $\boldsymbol{Q}$, and the familiar velocity vector $\boldsymbol{U}$.

Explicit Convective Step. We proceed with the matrix formulation for the Navier–Stokes steps as previously with the present velocity-correction method, see Section 3.4.4. The convective term is first integrated explicitly with

$$\gamma_0 \hat{\boldsymbol{U}} = \sum_{i=0}^{J-1} \alpha_i \boldsymbol{U}^{n-i} - \Delta t \boldsymbol{M}^{-1} \sum_{i=0}^{J-1} \beta_i \boldsymbol{F}^{c} \left(\boldsymbol{U}^{n-i} \right) \boldsymbol{U}^{n-i} + \Delta t \boldsymbol{F}^{n+1}, \tag{5.29}$$

where the variables $\hat{\boldsymbol{U}}$, $\boldsymbol{M}$, $\boldsymbol{F}^{c}$, and $\boldsymbol{F}$ are known from Chapter 3.

Pressure Poisson Equation and Projection. The pressure Poisson problem does in this case have Neumann boundary conditions only and yields

$$\boldsymbol{L} \boldsymbol{P}^{n+1} = \frac{\gamma_0}{\Delta t} \boldsymbol{A} \hat{\boldsymbol{U}}, \tag{5.30}$$

with the Laplace and Divergence operators $\boldsymbol{L}$ and $\boldsymbol{A}$, as well as the pressure vector $\boldsymbol{P}$. Using the pressure available, a divergence-free velocity vector $\hat{\hat{\boldsymbol{U}}}$ may be obtained by the local projection, including the div-div penalty matrix $\boldsymbol{D}$:

$$\left(\boldsymbol{M} + \tau_D \boldsymbol{D} \right) \hat{\hat{\boldsymbol{U}}} = \left(\boldsymbol{M} \hat{\boldsymbol{U}} + \frac{\Delta t}{\gamma_0} \boldsymbol{B} \boldsymbol{P}^{n+1} \right) \tag{5.31}$$

Table 5.1.: SA-RANS: Channel flow cases and resolutions. N_{ei} number of elements per spatial direction i, Re_τ friction Reynolds number, γ mesh stretching parameter, and Δy^+_{1e} width of first off-wall element. All computations employ $k = 4$ as the polynomial degree.

case	$N_{e1}\times N_{e2}$	Re_τ	γ	Δy^+_{1e}
ch_SA	16×16	180	1.8	5.5
	16×16	395	2.0	9.3
	16×16	590	2.25	9.8
	32×32	950	2.0	9.8
	32×64	2,000	2.0	9.7

with the gradient operator $\boldsymbol{B}$. As with the standard velocity-correction scheme, the projection is a local problem since the matrix is block-diagonal.

Implicit Viscous Step. Finally, we arrive at the velocity solution at time $t^{n+1} = t^n + \Delta t$ by solving the Helmholtz equation, given as

$$\left(\frac{\gamma_0}{\Delta t}\boldsymbol{M} - \boldsymbol{F}^{\nu+\nu_t}\left(\widetilde{\boldsymbol{N}}^{n+1}\right)\right)\boldsymbol{U}^{n+1} = \frac{\gamma_0}{\Delta t}\boldsymbol{M}\hat{\hat{\boldsymbol{U}}}, \tag{5.32}$$

where the viscous flux is denoted by $\boldsymbol{F}^{\nu+\nu_t}$ and is dependent on the SA solution.

The solution strategies for all substeps are equivalent to the ones of Chapter 3, except that a preconditioned GMRES instead of CG solver is employed for the nonsymmetric viscous problem.

5.4. Numerical Examples

The present approach is validated through the two numerical examples of turbulent channel flow and periodic hill flow. All computations are carried out with a scheme of temporal accuracy of second order (BDF2) including adaptive time stepping and we take the spatial polynomial degree of $k = 4$ for velocity, pressure, and the eddy viscosity working variable, which has performed best in terms of accuracy and time-to-solution in preliminary investigations.

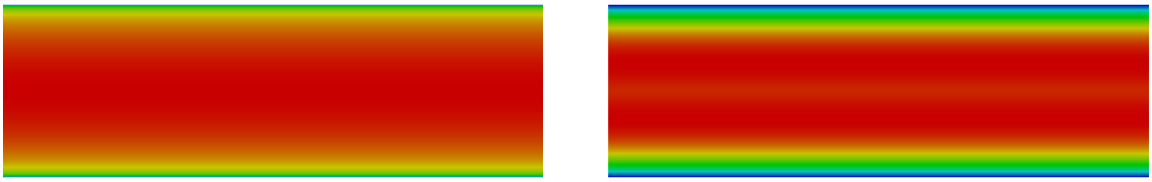


Figure 5.1.: SA-RANS: Turbulent channel flow at $\mathrm{Re}_\tau = 590$: Numerical solution of the streamwise velocity (left) and solution of the eddy-viscosity working variable (right). Red indicates high and blue low values.

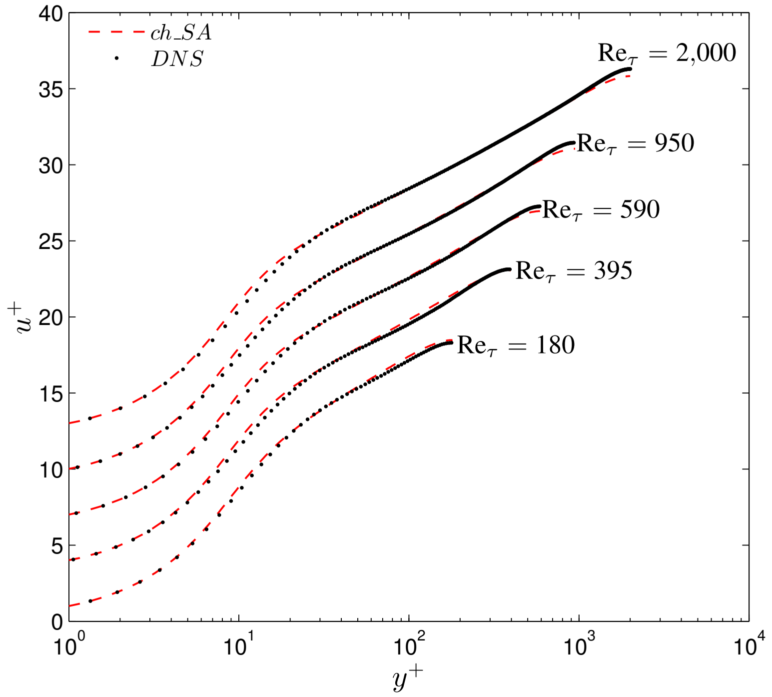


Figure 5.2.: SA-RANS: Velocity solution of turbulent channel flow $u^+ = u/u_\tau$ for $\mathrm{Re}_\tau = 180$, 395, 590, 950, and 2,000, each shifted upwards by three units for clarity. All computations have been carried out with $k = 4$ and meshes according to Table 5.1.

5.4.1. Turbulent Channel Flow

We consider turbulent channel flow using a similar setup as for the eddy-resolving simulations in the previous chapter, however in two space dimensions only. The computational domain is of the dimensions $2\pi \times 2\delta$ in streamwise and wall-normal direction, respectively. Periodic boundary conditions are specified in streamwise direction and no-slip boundary conditions are prescribed at the solid walls. The flow is driven by a constant body force derived from the nominal quantities and the velocity is normalized using the numerical wall shear stress in $u^+ = u/u_\tau$ with $u_\tau = \sqrt{\tau_w/\rho}$. We consider five Reynolds numbers according to the available reference data by [187] up to $\mathrm{Re}_\tau = 590$, by [6] ($\mathrm{Re}_\tau = 950$), and by [116] ($\mathrm{Re}_\tau = 2{,}000$). Regarding the resolution of the laminar sublayer, *a priori* investigations in Section 8.1 have shown that a sufficient criterion for RANS in the present high-order context is to require that the width of the first off-wall cell should be smaller than $\Delta y^+_{1\mathrm{e}} < 10$ if a polynomial degree of $k = 4$ is used, such that the first off-wall Gauss–Lobatto node is located below $y^+_{1,\mathrm{GL}} = 1.7$ according to Table 5.1. All simulation cases and resolutions are presented in Table 5.1. The time step is adaptively computed based on $\mathrm{Cr} = 0.08$ and $D = 0.03$. The solution for $\mathrm{Re}_\tau = 590$ is visualized in Figure 5.1 via the streamwise velocity and the eddy viscosity working variable.

Profiles of the mean streamwise velocity are shown in Figure 5.2 for all cases. Excellent agreement with the reference data is observed overall, in particular in the log-layer. In the buffer layer,

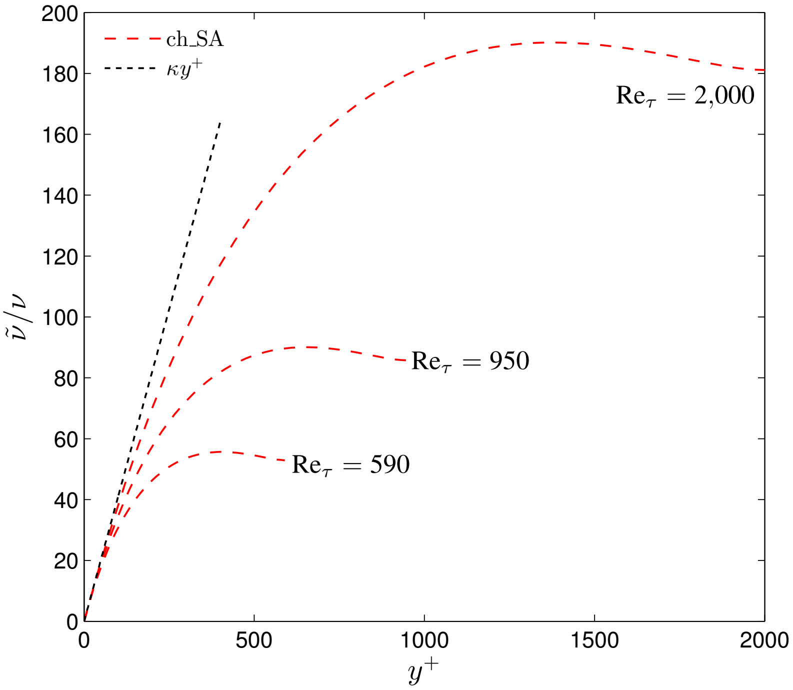


Figure 5.3.: SA-RANS: Profiles of $\widetilde{\nu}$, the eddy-viscosity-like variable of the SA model for $\mathrm{Re}_\tau =$ 590, 950, and 2,000 in comparison to the analytical log-layer solution of ν_t.

the SA model exhibits a small modeling error, since the mean velocity is slightly overestimated in comparison to the DNS.

Further, the profiles of the working variable of the SA model are plotted in Figure 5.3. Therein, the numerical results are compared to the analytical solution in the log-layer, which is given as a linear relation with respect to the wall coordinate by $\widetilde{\nu} = \kappa y^+ \nu$ [134]. This result may for example be obtained by considering Prandtl's mixing length model (Equation (2.16)) according to $\nu_t = (\kappa y)^2 \, |\mathrm{d}u/\mathrm{d}y|$. Inserting the velocity profile of the log-layer yields directly $\nu_t = \kappa y^+ \nu$ and we have $\widetilde{\nu} \approx \nu_t$ in the log-layer. A comparison with the numerical profiles of $\widetilde{\nu}$ shows good agreement between the curves within the log-layer up to approximately $y/\delta = 0.2$. Moreover, the working variable of the SA model extends the linear profile to the viscous sublayer such that the numerical resolution requirements of $\widetilde{\nu}_h$ in the inner layer are very low.

5.4.2. Flow over Periodic Hills

As a second numerical example, flow over periodic hills is considered according to Section 4.3 at a Reynolds number of $\mathrm{Re}_H = 10{,}595$ and according to Rapp and Manhart [211] of $\mathrm{Re}_H =$ 19,000. As noted earlier, this example has been investigated rigorously regarding RANS within the European ATAAC initiative [232] (see the final report by Jakirlić for a cross-comparison of the results) as well as Jakirlić and co-workers; see, e.g., [125].

The simulation setup is similar to Section 4.3, but considers a 2D domain only. The domain is of the dimensions $9H \times 3.036H$ in streamwise and vertical direction, respectively, with no-slip

Table 5.2.: SA-RANS: Simulation cases and resolutions of the periodic hill benchmark. $\mathrm{Re}_H = 10{,}595$: ph10595_SA_Ref wall resolved RANS simulation, NTS_SA_Ref reference using the SA model [123], DNS 10595 DNS according to Section 4.3. $\mathrm{Re}_H = 19{,}000$: ph10595_SA_Ref wall resolved RANS simulation, RM_EXP experiments [211]. Resolutions are specified in terms of elements per direction N_{ei} and grid points $N_i = N_{ei}(k+1)$.

case	Re_H	approach	$N_{e1}\times N_{e2}$	$N_1\times N_2\times N_3$	$x_{1,\text{reatt}}/H$
ph10595_SA_Ref	10,595	RANS (SA)	64×32	320×160	7.68
NTS_SA_Ref	10,595	RANS (SA)	-	161 × 161	7.7
DNS 10595	10,595	DNS	-	896×448×448	4.51
ph19000_SA_Ref	19,000	RANS (SA)	64×32	320×160	7.67
RM_EXP	19,000	experiment	-	-	3.94

boundary conditions at the top and bottom wall and periodic boundary conditions in streamwise direction. The meshes considered consist of 64×32 cells for both Reynolds numbers, and the grid is stretched towards the boundary in order yield a good resolution of the near-wall area. While almost all cells at the wall fulfill the requirement of $y_{1e}^+ < 10$ for the lower Reynolds number, the sharp peak present in the wall shear stress yields a minor violation of this relation. Although the grid stretching ratio is higher for the higher Reynolds number, the near-wall resolution lies at approximately $y_{1e}^+ = 15$ at the upper wall and reaches a peak of $y_{1e}^+ = 20$ near the hill crest at the lower wall, which can be considered to be just sufficient for the accurate prediction of the mean velocity. An overview of the meshes and cases is given in Table 5.2. The adaptive time stepping algorithm according to Section 5.3.2 with $\mathrm{Cr} = 0.08$ and $D = 0.02$ is used. The flow at $\mathrm{Re}_H = 10{,}595$ is visualized via the components of the mean velocity, the pressure, and the eddy viscosity working variable in Figure 5.4.

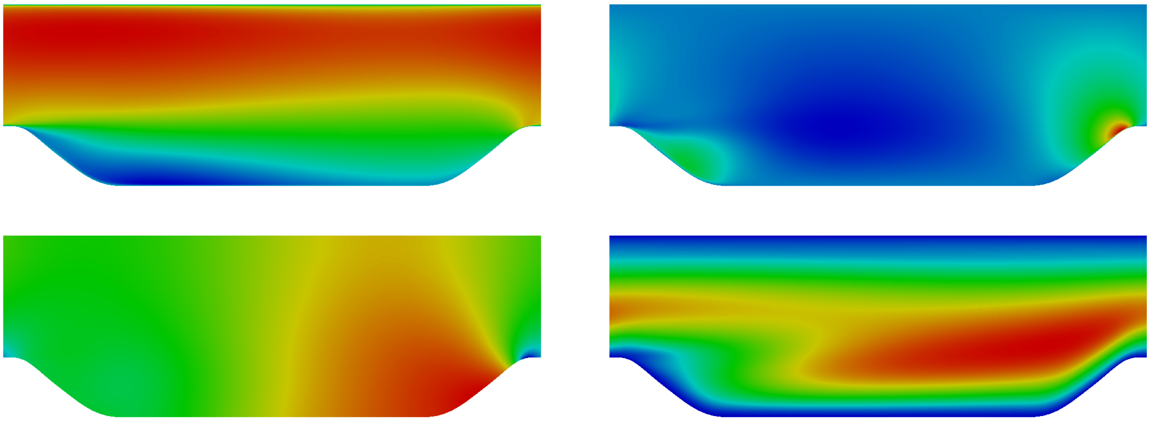


Figure 5.4.: SA-RANS: Turbulent flow over periodic hills at $\mathrm{Re}_H = 10{,}595$. Solution of the streamwise velocity (upper left), vertical velocity (upper right), pressure (lower left), and solution of the eddy-viscosity working variable (lower right). Red indicates high and blue low values.

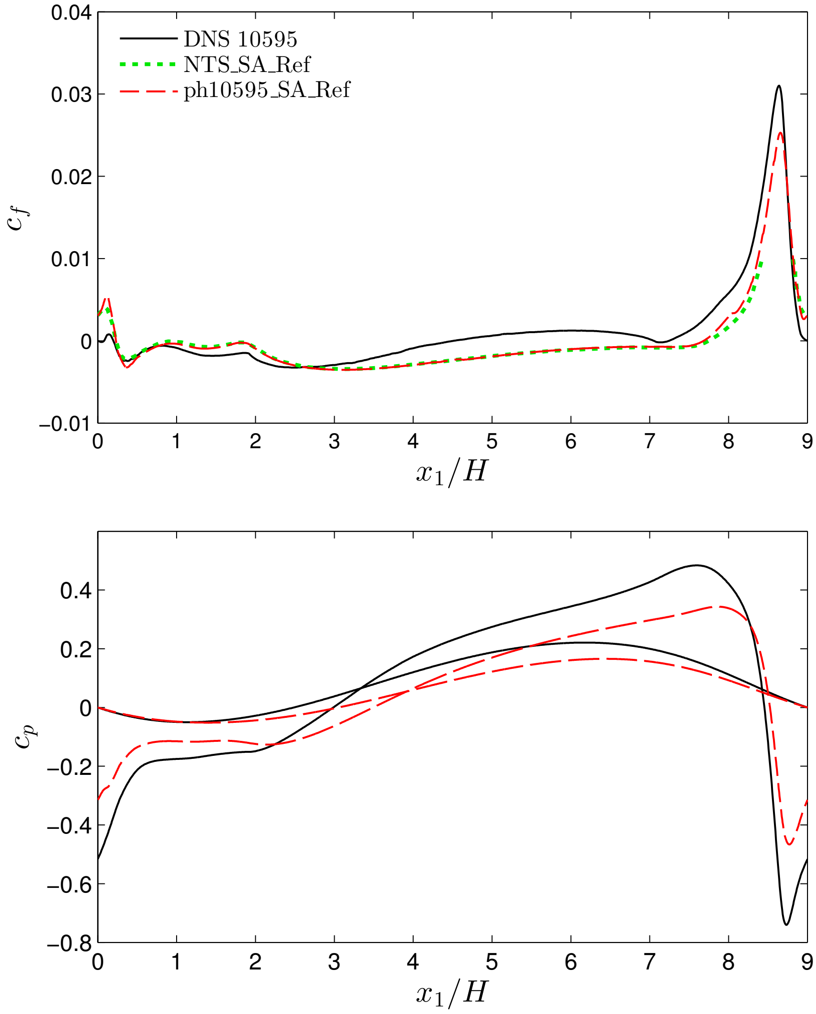


Figure 5.5.: SA-RANS: Friction coefficient (left) and pressure coefficients (right) for $\mathrm{Re}_H =$ 10,595 comparing resolved RANS simulations with DNS data. The shallower pressure curves correspond to the top wall. For the case NTS_SA_Ref, there is no c_f-reference data available between $x_1/H =$ 8.3 and $x_1/H =$ 8.8.

The results are compared to three data sets, which are also included in Table 5.2. For the case $\mathrm{Re}_H =$ 10,595, the DNS reference data presented in Section 4.3 is employed, which is labeled in the figures as DNS 10595. The experimental data of Rapp and Manhart [211] is used as a reference for $\mathrm{Re}_H =$ 19,000, labeled RM_EXP. As further reference data, we consider a wall-resolved simulation of the case $\mathrm{Re}_H =$ 10,595 using the SA model, which has been conducted by Strelets and Adamian within the ATAAC project and has been extracted from vector-graphics plots in [123]. This reference data is labeled NTS_SA_Ref. Since we use the same RANS approach, we expect good agreement to the latter RANS simulation. We also expect considerable differences between the SA solution and the DNS or experimental results since RANS turbulence models generally have difficulties in predicting the flow separation from the curved boundary of this flow example [125].

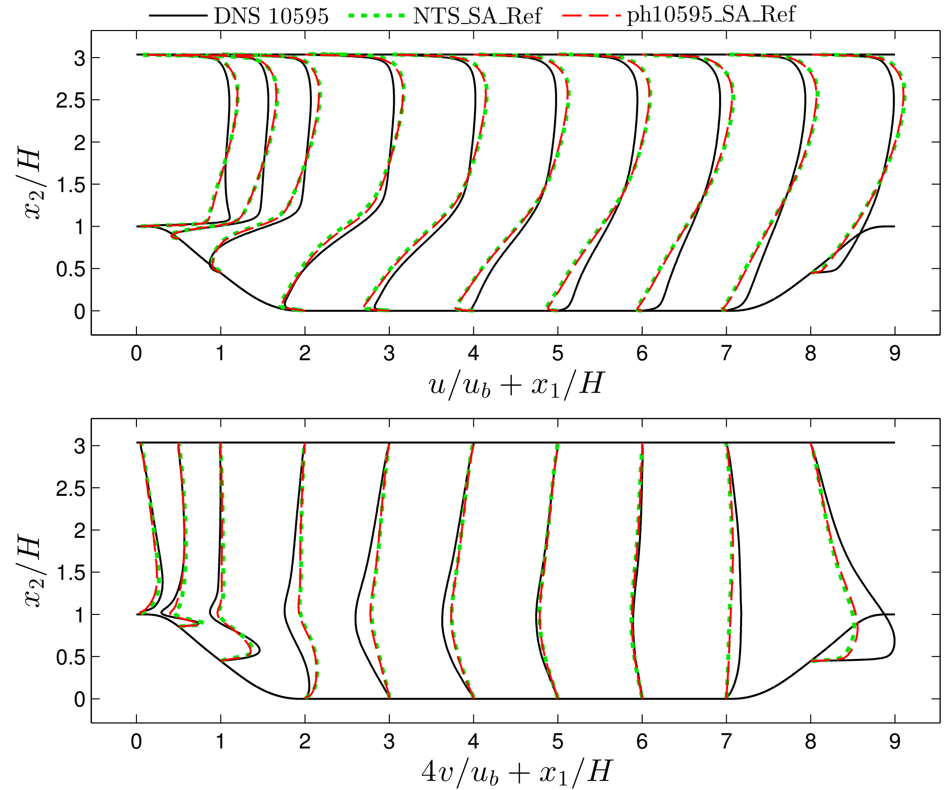


Figure 5.6.: SA-RANS: Streamwise (top) and vertical (bottom) velocity components u and v of flow past periodic hills at $\mathrm{Re}_H = 10{,}595$. Comparison of RANS and DNS data.

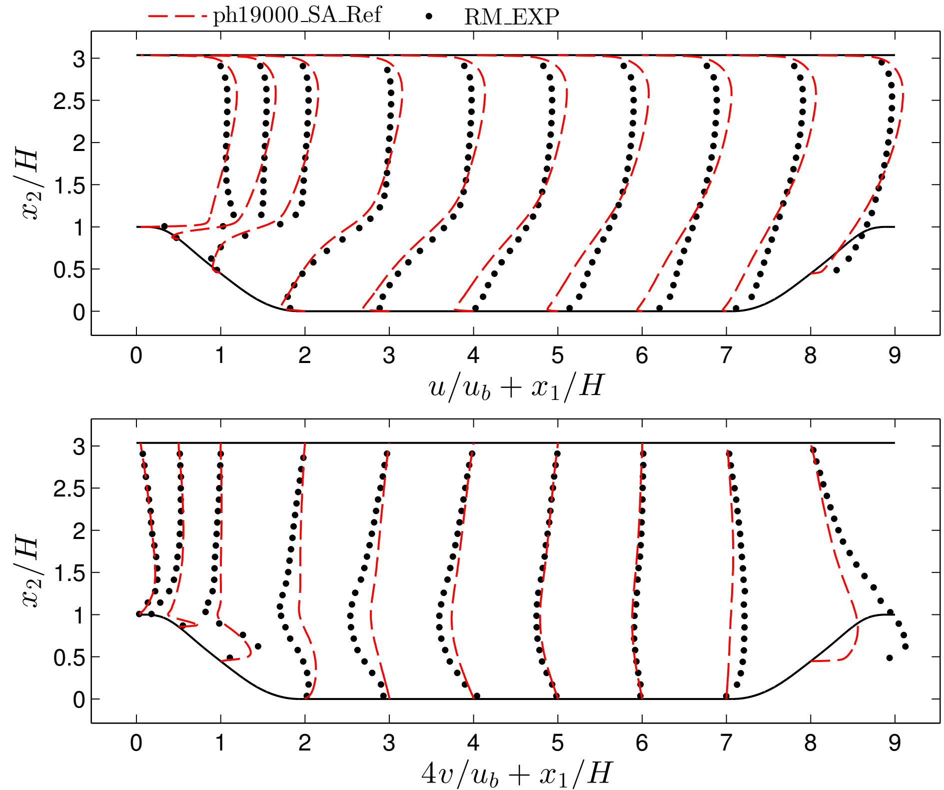


Figure 5.7.: SA-RANS: Streamwise (top) and vertical (bottom) velocity components u and v of flow past periodic hills at $\mathrm{Re}_H = 19{,}000$. Comparison of RANS and reference data.

The skin friction and pressure coefficients are shown in Figure 5.5 for $\mathrm{Re}_H = 10{,}595$. The comparison of the available skin friction data yields very good agreement with the case NTS_SA_Ref. The accordance with the DNS is less satisfactory, however. Due to the good agreement between the RANS cases, it may be concluded that this error stems from the SA model, and not from the numerical discretization. The conclusions drawn from the comparison of the SA results with the DNS regarding the pressure coefficient are accordingly. Reference data using the SA model are not available for these quantities, however.

The mean streamwise and vertical velocity is shown in Figure 5.6 for the Reynolds number $\mathrm{Re}_H = 10{,}595$ at ten streamwise stations. The results exhibit a similar quality as for the skin friction. The agreement with the RANS data is very good, the differences to the DNS are substantial, however. The picture is analogous for the mean velocity profiles at $\mathrm{Re}_H = 19{,}000$, shown in Figure 5.7.

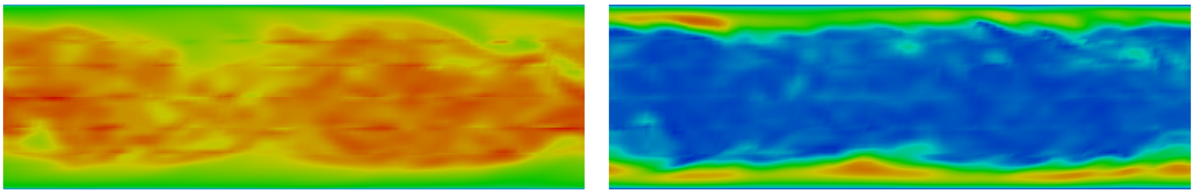


Figure 5.8.: SA-DDES: Turbulent channel flow at $\mathrm{Re}_\tau = 950$: Contour of the instantaneous velocity magnitude (left) and solution of the eddy-viscosity working variable (right). Red indicates high and blue low values.

5.5. Application to Detached-Eddy Simulation

The deficiencies of the SA-RANS model for the periodic hill flow example motivate an extension of the solver to DES according to Spalart et al. [244], see also the introduction in Section 2.3.4. In DES, the RANS model acts as a statistical model in the near-wall region only. Further away from the wall, the inhomogeneous energy carrying eddies are comparably large and their resolution is often affordable (see Section 2.1.3). The idea of DES is therefore to modify the SA model in such a way that it acts as an LES subgrid model in the bulk flow and the most important unsteady turbulent motions are resolved. The statistical modeling of the boundary layer turbulence allows very coarse cell sizes in the wall-parallel directions in comparison to the length scales of the turbulent eddies. The numerical methods presented herein can be used for both the WMLES and the classical DES branch (see Section 2.3.4), but the numerical tests in this section will focus on WMLES.

The original DES methodology modifies the SA model by limiting the wall distance function y present in the source term (Equation (5.4)) with a characteristic cell length h according to

$$y_{\mathrm{DES}} = \min(y, C_{\mathrm{DES}} h), \tag{5.33}$$

where the parameter C_{DES} has been calibrated to $C_{\mathrm{DES}} = 0.65$ and the grid length scale is defined as the maximum of the cell length over the space dimensions $h = \max(\Delta x_1, \Delta x_2, \Delta x_3)$ [236].

Table 5.3.: SA-DDES: Channel flow cases and resolutions. Specification of the first off-wall node through $y^+_{1,\mathrm{GL}} = y_{1,\mathrm{GL}} u_\tau / \nu$. Resolutions are specified in terms of elements per direction N_{ei} and grid points $N_i = N_{ei}(k+1)$.

Case	$N_{e1} \times N_{e2} \times N_{e3}$	k	$N_1 \times N_2 \times N_3$	Re_τ	γ	$\Delta y^+_{1,\mathrm{GL}}$
ch950_N16×16×8_k4_DDES	16×16×8	4	80×80×40	950	2.75	1.32
ch2000_N16×32×8_k4_DDES	16×32×8	4	80×160×40	2,000	2.75	1.16

As a result, the RANS model acts as a one-equation LES model if $y > C_{\mathrm{DES}} h$. In this work, we consider the enhanced version of this approach by Spalart et al. [243], which is denoted by delayed detached eddy simulation (DDES). This variant is given by the functions

$$r_d = \frac{\nu + \nu_t}{\sqrt{(\nabla \boldsymbol{u})_{ij} (\nabla \boldsymbol{u})_{ij} \kappa^2 y^2}}, \tag{5.34}$$

$$f_d = 1 - \tanh\left((8 r_d)^3\right), \tag{5.35}$$

which results in the new length scale

$$y_{\mathrm{DDES}} = y - f_d \max(0, y - C_{\mathrm{DES}} h). \tag{5.36}$$

The increasing resolution power of the DG scheme with increasing polynomial degree should be taken into account in the (D)DES grid length scale h [267]. Based on the analysis of the resolution power of DG schemes performed in [188], we choose

$$h = \frac{\max(\Delta x_{e1}, \Delta x_{e2}, \Delta x_{e3})}{k+1} \tag{5.37}$$

as a length scale, with the given cell size Δx_{ei} in the respective space dimension i, in contrast to the factor of $1/k$ chosen in [267]. The DDES methodology was previously successfully applied within high-order DG for example in [179, 267].

For the following assessment of the DDES approach, we consider the RANS solver presented in this chapter and employ BDF2 time integration with the corresponding adaptive time stepping algorithm using $\mathrm{Cr} = 0.14$ and $D = 0.02$ for all simulation cases. The Dirichlet boundary conditions are applied weakly as in the standard solver in Chapter 3.

5.5.1. Turbulent Channel Flow

The DDES approach is investigated using two turbulent channel flow simulations at $\mathrm{Re}_\tau = 950$ and 2,000 by considering the setup of Section 4.2. Table 5.3 summarizes the properties of the simulation cases and shows the resolution of the near-wall criterion. The spatial discretizations consist of 16 cells in streamwise and 8 cells in spanwise direction for both Reynolds numbers, yielding a switching location between the RANS and LES modes of the DDES model of $y/\delta \approx 0.05$. This choice of parameters clearly yields a WMLES, since solely the near-wall region is computed in RANS mode and the turbulent structures in the outer layer are explicitly computed. The number of wall-normal grid points is adjusted in order to enable a good resolution of the

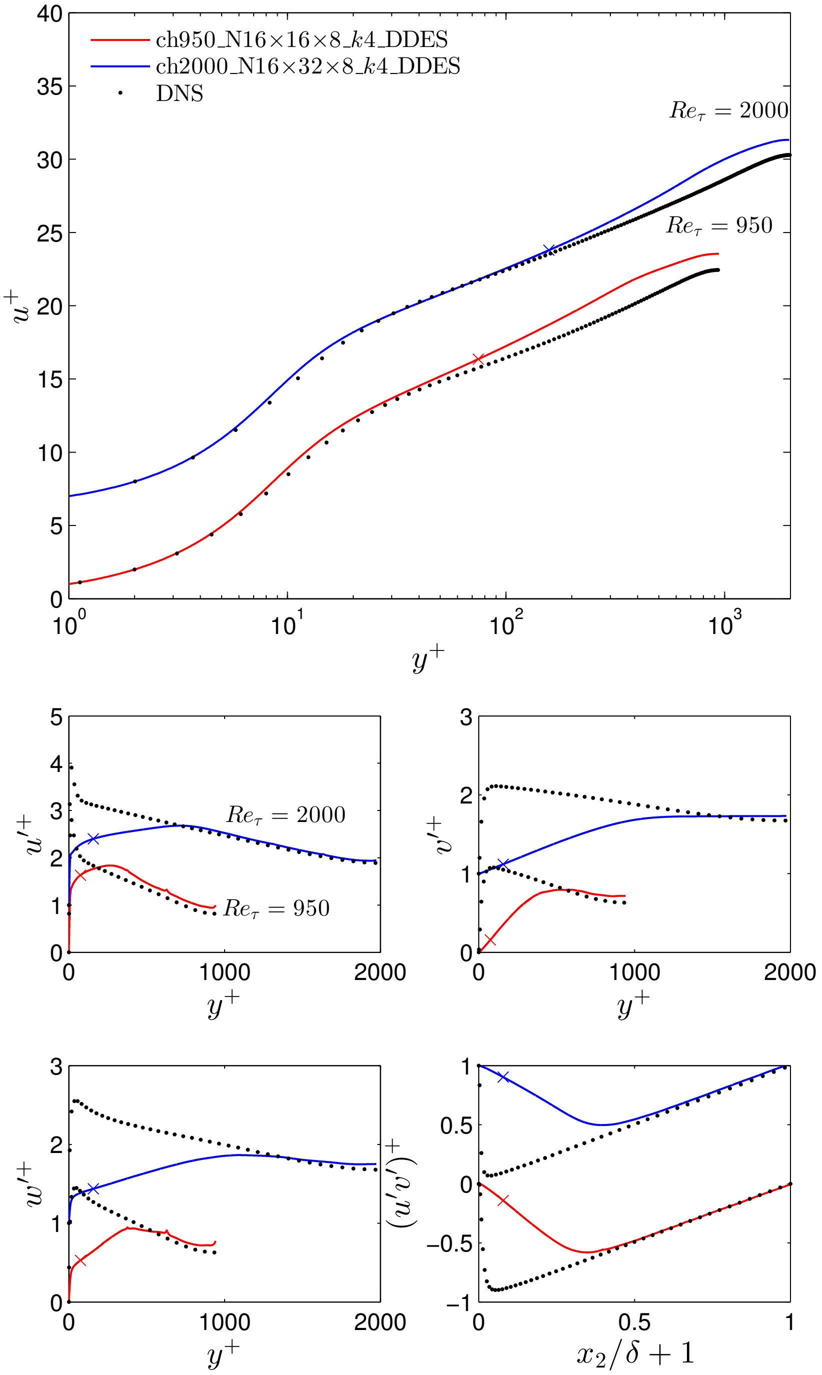


Figure 5.9.: SA-DDES of turbulent channel flow at $\mathrm{Re}_\tau = 950$ and 2,000: Mean velocity $u^+ = u_1/u_\tau$ (top) and RMS velocities $u'^+ = \mathrm{RMS}(u_1)/u_\tau$, $v'^+ = \mathrm{RMS}(u_2)/u_\tau$, and $w'^+ = \mathrm{RMS}(u_3)/u_\tau$ as well as RSS $(u'v')^+ = (u_1u_2)/u_\tau^2$ (bottom). For the case $\mathrm{Re}_\tau = 2{,}000$, the mean velocity is shifted upwards by six units and all other quantities by one unit for clarity. The symbol indicates the switching location between the RANS and LES modes of the DDES model ($y/\delta \approx 0.05$).

Table 5.4.: SA-DDES: Simulation cases and resolutions of the periodic hill benchmark at $\mathrm{Re}_H = 10{,}595$. DDES: ph10595_N32×16×16_k4_DDES coarse DDES simulation, ph10595_N64×32×32_k4_DDES fine DDES simulation; ILES according to Section 4.3: ph10595_N32×16×16_k4_ILES coarse ILES simulation, ph10595_N64×32×32_k4_ILES fine ILES simulation; DNS according to Section 4.3: DNS 10595. Resolutions are specified in terms of elements per direction N_{ei} and grid points $N_i = N_{ei}(k+1)$.

case	approach	$N_{e1}\times N_{e2}\times N_{e3}$	$N_1\times N_2\times N_3$	$x_{1,\text{reatt}}/H$
ph10595_N32×16×16_k4_DDES	DDES	32×16×16	160×80×80	5.02
ph10595_N64×32×32_k4_DDES	DDES	64×32×32	320×160×160	4.50
ph10595_N32×16×16_k4_ILES	ILES	32×16×16	160×80×80	3.90
ph10595_N64×32×32_k4_ILES	ILES	64×32×32	320×160×160	4.22
DNS 10595	DNS	-	896×448×448	4.51

viscous sublayer. The instantaneous flow field at the lower Reynolds number is visualized in Figure 5.8.

The velocity statistics are compared to the DNS according to [6] and [116] in Figure 5.9. The mean velocity agrees well with the DNS up to the outer edge of the inner layer, where the RANS mode is active, and subsequently sees a so-called log-layer mismatch [205, 268], which is typical for wall-modeling DES. The reason for this behavior is that the turbulence is modeled entirely in the inner layer, but the momentum transfer relies on the resolved turbulent eddies in the outer layer. Between these regions, the turbulent motions are not fully developed nor is the RANS model fully active, which yields an error in the sum of the modeled and resolved shear stress. Possible improvements of the DDES model that reduce the effect of this issue are presented in [237].

5.5.2. Flow over Periodic Hills

The DDES solver is further assessed via flow over periodic hills at $\mathrm{Re}_H = 10{,}595$ using the computational setup according to Section 4.3. Regarding DDES, this flow was assessed within the ATAAC project including several alternative hybrid RANS/LES methods (see the final report by Jakirlić for cross-comparison of results). Two meshes are considered, 32×16×16 cells and 64×32×32 cells, both with $k = 4$. These meshes are very similar to the coarse and medium mesh used for the ILES h refinement study in Section 4.3.5. The location of the first off-wall Gauss–Lobatto node is plotted in Figure 5.10. An instantaneous numerical solution is displayed in Figure 5.11.

The skin friction and pressure coefficients are compared in Figure 5.12 to the case DNS 10595 discussed in Chapter 4. The skin friction at the lower and upper walls is in excellent agreement with the DNS for both cases. In accordance with the accurate prediction of the skin friction curves at the lower wall, the length of the recirculation bubble is predicted well for both cases as $5.02H$ and $4.50H$ for the coarser and finer mesh, respectively, according to Table 5.4. These results are in fact much more accurate than the ILES cases presented in the preceding chapter

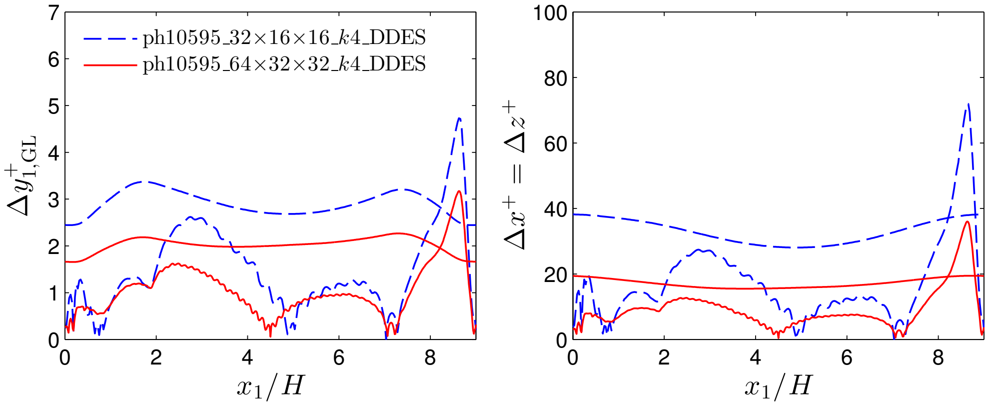


Figure 5.10.: SA-DDES: Flow over periodic hills: Location of first off-wall Gauss–Lobatto point $\Delta y^+_{1,\mathrm{GL}}$ (left) as well as normalized grid spacings in streamwise and spanwise direction $\Delta x^+ = \Delta z^+$ (right). The shallower curves correspond to the upper wall.

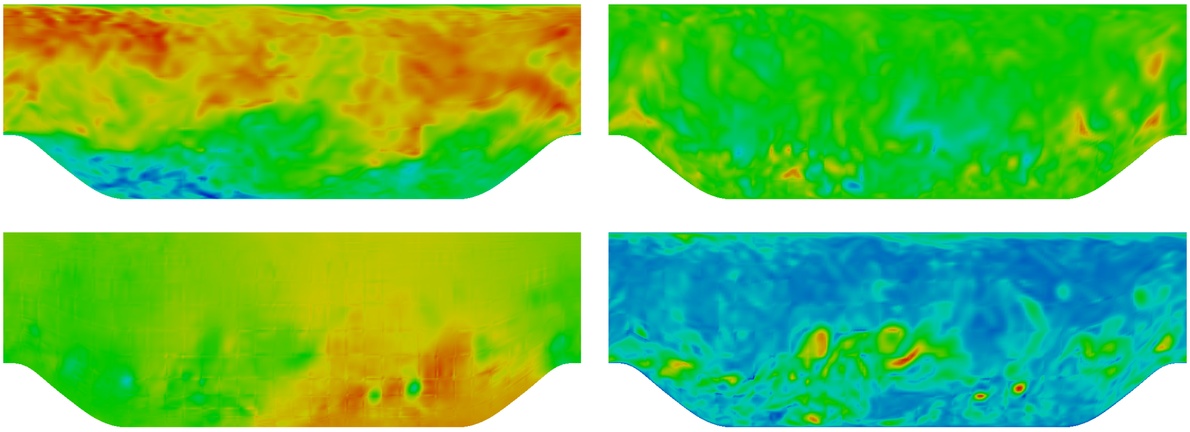


Figure 5.11.: SA-DDES of turbulent flow over periodic hills at $\mathrm{Re}_H = 10{,}595$ using the coarser mesh. Instantaneous DDES solution of the streamwise velocity (upper left), vertical velocity (upper right), pressure (lower left), and solution of the eddy-viscosity working variable (lower right). Red indicates high and blue low values.

with similar meshes, which predicted a shorter recirculation bubble (see reattachment lengths included in Table 4.5). The pressure coefficient is also in remarkable agreement with the DNS.

The profiles of the mean streamwise velocity, mean vertical velocity, RSS, and TKE are plotted in Figure 5.13. Analogous to the high accuracy of the skin friction curves, these profiles agree very well with the DNS. Solely in the TKE distributions, unphysical peaks are visible at the boundaries of the cells, these appear to be smaller than for the ILES of the corresponding meshes in the previous chapter, however.

5.6. Summary

In this chapter, we extended the dual-splitting scheme by an additional step, in which the SA equation is solved. This method provides a numerical approach that is relatively simple to im-

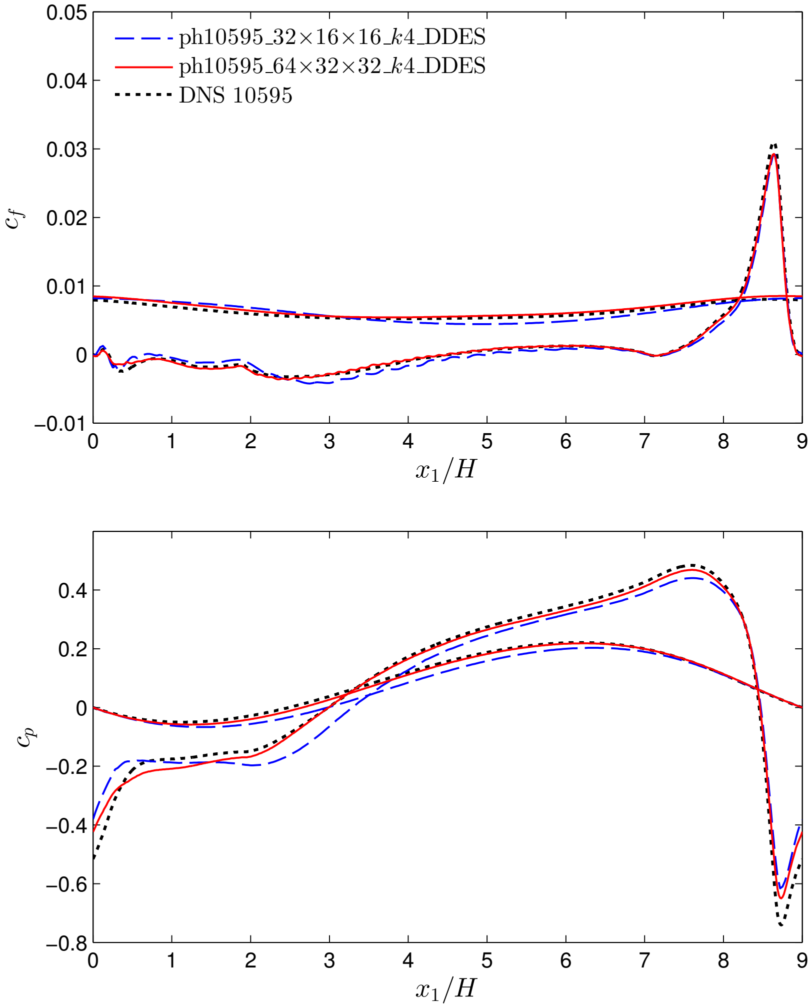


Figure 5.12.: SA-DDES: Skin friction (top) and pressure (bottom) coefficients of flow over periodic hills. The shallower curves correspond to the upper wall.

plement based on an existing flow solver using the velocity correction scheme. The approach has been applied to RANS and DES of turbulent channel flow as well as flow over periodic hills. A drawback may be that the scheme requires many time steps in comparison to the temporal time scales of unsteady RANS simulations. This fact renders implicit schemes more efficient for most applications, but the present method is sufficient for analyzing and validating the wall modeling approach presented in Part II of this thesis. The drawback is also relieved when DES is applied as a wall model, where turbulent motions are explicitly computed in the outer boundary layer, which have much smaller time scales.

The efficiency of the present DES method seems to be limited by two factors. If applied as a wall model for LES, the results show a log-layer mismatch in the turbulent channel flow example. In addition, while DES allows the application of relatively coarse meshes in the streamwise and spanwise direction, the wall-normal direction necessitates many grid points in order to resolve the laminar sublayer with $y_1^+ \sim 1$. The first issue may for example be improved by tuning the

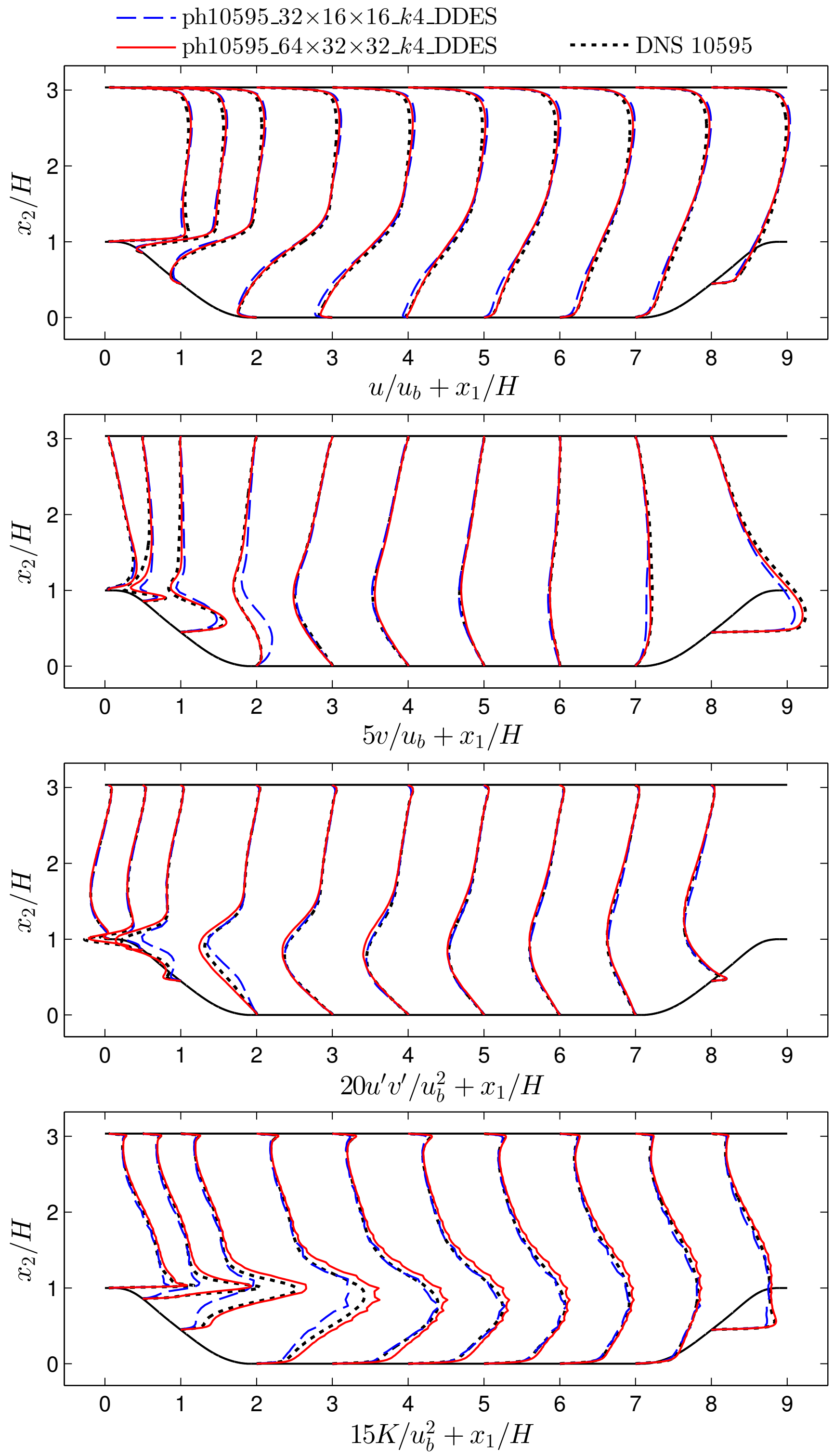


Figure 5.13.: SA-DDES: Profiles of mean velocity in streamwise and vertical direction u and v, RSS $u'v'$, as well as TKE $K = 1/2(u'u' + v'v' + w'w')$ (from top to bottom) of flow over periodic hills.

LES length scale in the DES according to [237]. As an alternative, both issues are addressed by the wall modeling approach presented in Part II of the present thesis.

II

Wall Modeling via Function Enrichment

6

Wall Modeling via Function Enrichment: Motivation, Formulation, Implementation

Wall modeling is the key technology for making industrial high-Reynolds-number flows accessible to computational analysis. In the second part of this thesis, a novel approach to computing the near-wall region in turbulent boundary layers is proposed, which significantly reduces the computational complexity compared to wall-resolved simulations at the same time as the full consistency in the numerical method is maintained. The function space of the Galerkin method is locally 'enriched' in such a way that it includes the law-of-the-wall in additional shape functions, besides the standard polynomial ones. The Galerkin method then automatically finds the optimal solution among all shape functions available. This construction allows the use of much larger cells in the near-wall region than in wall-resolved simulations since the numerical method is *a priori* capable of resolving a turbulent boundary layer. The idea gives the wall model vast flexibility in nonequilibrium boundary layers and is fully consistent, since the Navier–Stokes equations are computed in the whole boundary layer. The consistency of the wall model is necessary in order to guarantee the requirement of accuracy as defined in the introduction (Section 1.1).

The aim of this chapter is to introduce the basic concepts of this wall modeling technique. In order to allow for a comparison with other wall modeling approaches, we begin with the basic challenges related to the simulation of turbulent boundary layers and subsequently discuss how current flow simulation methodologies tackle these challenges. The new approach, wall modeling via function enrichment, is proposed in Section 6.3 for the DG and continuous FEM, including a review of potential enrichment functions as well as algorithms for spatial and temporal adaptation. The wall model is applied and assessed in the subsequent chapters in different scenarios, including the continuous FEM (Chapter 7) and DG (Chapters 8 to 10), as well as wall modeling in LES (Chapters 7, 9, and 10) and RANS (Chapter 8). This chapter represents a summary of the methods developed in Krank and Wall [153, 154], Krank et al. [150–152],and Kronbichler et al. [159, 160]).

6.1. The two Challenges in Computing Turbulent Boundary Layers

The simulation of turbulent boundary layers at high Reynolds number faces two fundamental challenges, which have to be addressed by any CFD approach. The velocity gradient is extremely sharp in the viscous sublayer just at the wall, and the whole boundary layer features a multitude of length and time scales, see Sections 2.1.3 and 2.1.4. Subsequent to the following discussion of

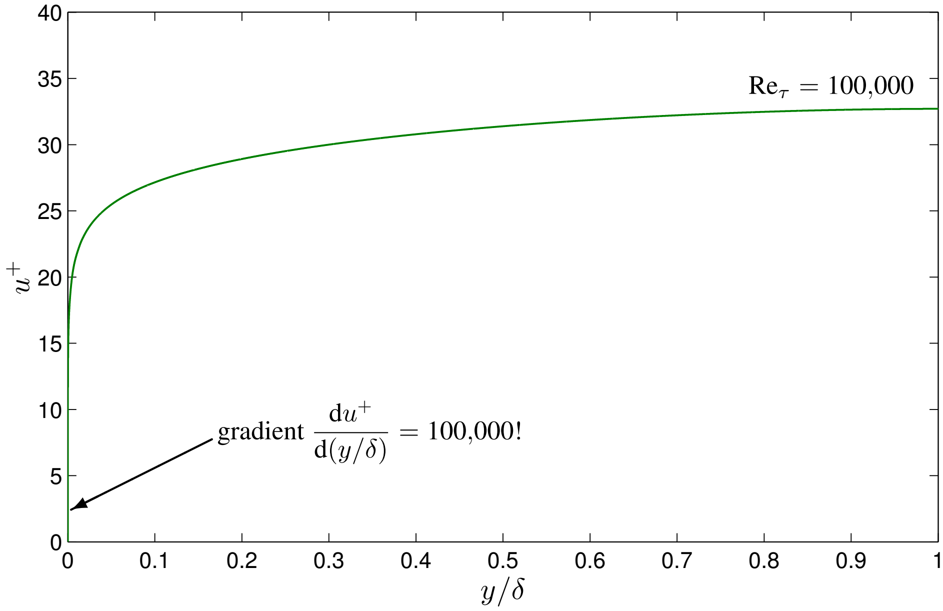


Figure 6.1.: Numerical velocity solution of turbulent channel flow $u^+ = u/u_\tau$ at $\mathrm{Re}_\tau = 100{,}000$.

these two challenges, the most common wall modeling approaches are reviewed with a particular focus on their strategies of tackling the challenges.

Velocity Gradient. In order to visualize the dimensions of the gradient, Figure 6.1 shows the numerical solution of turbulent channel flow at $\mathrm{Re}_\tau = 100{,}000$ plotted over the channel half height. Just at the wall, the velocity gradient is $\mathrm{d}u^+/\mathrm{d}(y/\delta) = 100{,}000$! This gradient has to be resolved or modeled in order to capture the skin friction accurately, which is of primary engineering interest. If this gradient has to be resolved by the numerical scheme, Nikitin et al. [195] suggest to place the first off-wall grid point around $y^+ = 1$ and to consecutively stretch the grid by a grid stretching factor of 1.15. This procedure yields a logarithmic scaling of the required number of grid points with Re_τ, which makes high Reynolds numbers in principle accessible. Yet, for a boundary layer spanning 100,000 wall units, 69 wall-normal grid layers would be required, and this number increases by 17 additional layers for every order of magnitude in Re_τ [195]. This is a very high cost in terms of computational complexity and storage requirements in comparison to the low engineering interest in the detailed velocity profile in the log-layer.

Turbulent Spatial and Temporal Scales. The turbulent scales present in the near-wall region have to be resolved or modeled in order to predict the velocity profile accurately, and thus the skin friction. The spatial mesh requirements for resolving the scales near the laminar sublayer are $\Delta x^+ = 100$ and $\Delta z^+ = 20$ according to Chapman [42]. This scaling in wall units yields an increase of the required number of grid points for LES with Re_τ^2 [12] and an additional scaling of the number of time steps with Re_τ due to the proportionality of the time step with the streamwise grid length according to the CFL condition, but also to resolve the temporal scales. Considering again the example of turbulent channel flow at $\mathrm{Re}_\tau = 100{,}000$, we may estimate the requirements by a comparison with the LES computations presented in Chapter 4, which yields approximately $1.5e10$ grid points and $1.7e8$ time steps. The spatial resolution alone seems to be feasible as may be seen from the strong scaling plot in Figure 3.8, where INDEXA was scaled up to $0.86e10$ grid points. However, each time step would with such a grid topology take more than one second wall time, which equals an overall computation time of approximately 5.4 years. Computation times of this order are not practicable, nor can a consumption of electricity of the order of $1e8$kWh in a single computation be justified. These excessive resolution requirements originate from the fact that the energetic eddies are very small in the near-wall area, in particular the streaks in the buffer layer, as it was discussed in Section 2.1.3 (see also Figure 2.5), and their size is universal in wall units. Within the log-layer, the largest scales are of similar size as the wall distance, so the resolution requirement for LES is much lower further away from the wall. The energy-carrying eddies in the outer layer scale in the length scale of the boundary conditions and it is affordable to compute these vortices. This example highlights the need for modeling the turbulent motions near the wall.

In addition to these two challenges, any wall modeling technique has to fulfill the underlying partial differential equations, the Navier–Stokes equations, as accurately as possible. In particular, none of the terms should be neglected entirely, since for example high longitudinal pressure gradients can lead to a nonequilibrium state in the boundary layer (see Section 2.1.3), where the velocity profile deviates from the equilibrium one. This has shown to be of significance especially in the inner boundary layer, since nonequilibrium conditions yield an immediate effect on the boundary layer in that region [47]. The next section discusses how state of the art CFD approaches deal with these two challenges.

6.2. A Review of Wall Modeling Approaches for CFD

In Section 2.3.4, we have introduced two wall modeling methodologies: hybrid RANS/LES and wall-stress models. In this section, we discuss how these wall models deal with the two challenges presented in the previous section. In addition, the problems resulting from the basic design of each wall modeling approach are reviewed. Three modeling approaches are considered in this discussion: DES, wall-stress models using wall functions, and the extension of the latter to the TLM. The arguments are summarized in Table 6.1, considering the categories: are the Navier–Stokes equations satisfied or modeled?, is the near-wall turbulence resolved or modeled?, is the velocity gradient resolved or modeled?, as well as the computational cost in terms of the scaling of the grid point requirements with the Reynolds number. For comparison, the DNS and LES approaches are included as well, where the Navier–Stokes equations are satisfied, the near-wall turbulence is essentially fully resolved in both approaches, the gradient is resolved, and the scaling of the computational cost is given according to Section 2.3.

6.2.1. Detached-Eddy Simulation

In DES, the Navier–Stokes equations are satisfied, since the governing equations are solved in the whole boundary layer. The near-wall turbulence is modeled in both approaches in a statistical sense by the RANS model. Estimates of the cost of DES lie in a wide range and depend on the

Table 6.1.: A comparison of eddy-resolving simulation approaches for turbulent boundary layers. The table shows the simulation methodology (DNS: direct numerical simulation, LES: large-eddy simulation, DES: detached-eddy simulation, TLM: two-layer model, WSM: wall-stress model, WME (WMLES/DES): wall modeling via function enrichment applied in conjunction with WMLES or DES), how the Navier–Stokes equations are considered (discretely satisfied or modeled), if the near-wall turbulence is resolved or modeled, if the velocity gradient is resolved or modeled, and the computational cost in required number of grid points.

methodology	Navier–Stokes	near-wall turb.	gradient	cost
DNS	satisfied	resolved	resolved	$\mathrm{Re}^{2.25}$ [40] to $\mathrm{Re}^{2.65}$ [45]
LES	satisfied	resolved	resolved	$\mathrm{Re}^{1.8}$ [42] to $\mathrm{Re}^{1.85}$ [45]
DES (WMLES)	satisfied	modeled	resolved	$\mathrm{Re}^{0.4}$ [42] to Re [45]
DES	satisfied	modeled	resolved	0.1 to $0.01 \cdot \mathrm{cost}(\mathrm{WMLES})$ [168]
WSM	modeled	modeled	modeled	$\mathrm{Re}^{0.4}$ [42] to Re [45]
TLM	satisfied (TBLE)	modeled	resolved	$\mathrm{Re}^{0.4}$ [42] to $\mathrm{Re}^{0.9}$ [204]
WME (WMLES)	satisfied	modeled	modeled	$\mathrm{Re}^{0.4}$ to Re
WME (DES)	satisfied	modeled	modeled	$< \mathrm{cost}(\mathrm{DES})$

location of the RANS/LES interface; DES may either be used to compute the whole boundary layer in RANS mode or as a wall model in LES (see Section 2.3.4). Regarding the WMLES branch, Chapman [42] estimates the spatial resolution requirements of simulations, in which the near-wall turbulence is modeled, to $\mathrm{Re}^{0.4}$, whereas Nikitin et al. [195] derive the relation of $\mathrm{Re}^{0.9}$ (see also [204]), and Choi and Moin [45] derive a scaling for WMLES with Re for a flat plate boundary layer. If DES is used to model the turbulence in the whole boundary layer, the cost can be significantly lower, since the wall-parallel grid length can be of the order of the boundary layer thickness, whereas approximately 10 times more cells are required in each wall-parallel direction in the WMLES branch [168].

What is often neglected in these estimates is the cost for resolving the gradient. As became clear in the previous section, the corresponding grid point requirements scale logarithmically, so they increase slowly indeed in the high-Reynolds number range ($y^+ > 5{,}000$), but the first $5{,}000$ wall units in the wall-normal direction come along with a fixed cost of approximately 48 layers of grid points (see Section 6.1). In addition, if larger cells could be used in the near-wall region, the cells in the subsequent layers could be larger as well due to the limitation of the grid stretching factor. Another issue with DES is the RANS/LES transition region if DES is used as a wall model for the inner layer only. The problem is that “LES content has to be generated in the outer part of the boundary layer” [223] since the momentum transfer is fully modeled in RANS but relies on the turbulent structures in LES. The result of this effect can be a so-called log-layer mismatch [205, 268], see the results obtained for turbulent channel flow in Section 5.5.1. A careful choice of the model parameters may solve this issue [237]; alternative approaches exist such as artificial forcing [139], which yields a narrower ‘gray area’ in the RANS/LES transition region.

6.2.2. WMLES Using Wall-Stress Models

The idea of wall-stress models is to model the inner layer by means of a wall function and to compute the outer layer via LES; see, e.g., [204]. The application of wall functions implies that the Navier–Stokes equations are modeled and the velocity profile of an equilibrium boundary layer can be used, although the wall functions may include an empirical correction for the pressure gradient. As an alternative to using wall functions, the wall shear stress can be modeled directly, as for example in the Werner and Wengle model [261]. By modeling the wall shear stress, the near-wall turbulence is modeled as well. One major advantage of wall-stress models is that the velocity gradient is not resolved explicitly, but included in the wall model. This fact allows the use of relatively uniform cell sizes in the whole computational domain. The computational cost scales as $\mathrm{Re}^{0.4}$ [42] to Re [45], and the cost can be of one to two orders of magnitude higher than in the classical DES, since the outer layer is resolved [168].

A clear drawback of this approach is that nonequilibrium boundary layers cannot be computed with the same fidelity as LES and DES (WMLES), since the Navier–Stokes equations are not discretized in the inner boundary layer. As it is discussed in [47], it is impossible to define a one-parameter wall function that takes into account all nonequilibrum effects. Some examples of failing wall-stress models will be shown in the context of the periodic hill flow in Chapters 7.3.2 and 10.5.2. Advanced wall functions seek to improve this problem by including corrections for convective and pressure-gradient effects; see, e.g., [235, 269] and also the assessment in [89]. An issue with wall-stress models is the choice of the coupling point at which the velocity is taken as input for the wall model. A location closer to the wall may improve the validity of the wall function, but the turbulent motions are marginally resolved in the first and second off-wall grid layer, so the coupling point should be placed further inside the domain.

6.2.3. WMLES Using the Two-Layer Model

More accurate TLM have been developed, which solve the simplified TBLE on a separate domain between the wall and the first off-wall node to predict the momentum transfer inside the boundary layer [14, 260], see also [203, 204] for a comprehensive overview. These models can be seen as a compromise between DES and wall-stress models. As the Navier–Stokes equations are solved in the near-wall region, they are satisfied with good approximation in the inner layer. The unresolved turbulent motions can be modeled by an algebraic RANS model, which provides sufficient accuracy in the inner layer [44, 204], and the velocity gradient is resolved as well. These characteristics make this methodology quite similar to DES (WMLES), but the RANS/LES coupling is handled differently. Accordingly, the computational cost is expected to lie in a similar range as wall-stress models and DES (WMLES). Recent developments in this field include an application with unstructured grids [26, 200], improvement of wall shear stress predictions in equilibrium boundary layers [137, 138, 200], laminar-turbulent transition [26, 27] and applications of the TBLE with further simplifications [44, 46]. If the simplifications neglect entire terms in the Navier–Stokes equations, the same problems in accuracy as with simple wall-stress models can arise, however. In this context, see also the assessment of the TLM in [89], which concludes that the convective term should not be neglected if the pressure gradient is considered, otherwise the results can even be worse than the predictions of simple equilibrium models.

6.2.4. Preview of Wall Modeling via Function Enrichment

Finally, we give a preview of the novel approach presented in Section 6.3 within these categories. Wall modeling via function enrichment satisfies the Navier–Stokes equations in a discrete sense, which makes it well-suited for nonequilibrium conditions, and the near-wall turbulence is modeled. What is unique with this method in comparison to all existing methods is that the velocity gradient is modeled to a certain extent, at the same time as the Navier–Stokes equations are satisfied. This may be seen in the overview in Table 6.1.

The method has at least two branches analogous to DES. It may be applied in conjunction with the classical DES turbulence modeling approach, where the turbulence in the whole boundary layer is modeled, or with a WMLES approach with RANS in the inner layer and LES in the outer layer on the same grid. Since the gradient is modeled in the inner layer, the computational complexity is significantly reduced in comparison to DES. For example, the first off-wall grid node can typically be placed in the region of 100 wall units, which yields a saving of 20 grid point layers according to the grid stretching discussed previously. Due to the lower aspect ratio of the cells in the computational domain, the actual saving should be much higher since the mesh can be coarser in the outer layer as well and due to the easier solution of linear systems in implicit schemes. In explicit schemes, the larger cells result in an even larger saving due to the significantly relaxed time step restrictions. Finally, despite the similarity with DES regarding the turbulence modeling, the approach enables the construction of a multiscale turbulence model, which does not show a log-layer mismatch by default.

6.2.5. Wall Modeling in RANS

We close this section with a brief comment on wall modeling in RANS. The advantage of wall modeling in this context is that the velocity gradient does not have to be resolved. Common techniques employ wall functions analogous to wall-stress models [64, 169], so both the gradient and the Navier–Stokes equations are modeled. If wall modeling via function enrichment is applied to RANS, the resolution requirements are reduced in a similar way, but the Navier–Stokes equations are satisfied in a discrete sense. These aspects are summarized in Table 6.2. A comparison of the computational cost is not possible, since the wall-parallel resolution requirements are given through the geometry of the flow problem rather than the boundary layer properties. However, the saving in the wall-normal direction is analogous to DES and WMLES, so wall modeling via function enrichment can model of the order of 20 to 40 grid layers plus significant advantages in the conditioning of the equation system.

6.3. Capturing the Gradient via Function Enrichment

The primary innovation of this work is a novel approach to wall modeling, which combines the advantage of large discretization cells at the wall with high accuracy in nonequilibrium boundary layers. This is done by constructing a problem-tailored numerical method that is inherently capable of resolving high velocity gradients with very coarse meshes ($y_1^+ \sim 10$ to 1000) similarly to the aforementioned wall-stress models, while preserving consistency and flexibility in nonequilibrium boundary layers. The basic idea is that a wall function is inserted into the function space

Table 6.2.: A comparison of RANS approaches for turbulent boundary layers. The table shows the simulation methodology (RANS: wall-resolved, WME (RANS): wall modeling via function enrichment with RANS), how the Navier–Stokes equations are considered (discretely satisfied or modeled), if the near-wall turbulence is resolved or modeled, and if the velocity gradient is resolved or modeled.

methodology	Navier–Stokes	near-wall turb.	gradient
RANS	satisfied	modeled	resolved
RANS (wall-modeled)	modeled	modeled	modeled
WME (RANS)	satisfied	modeled	modeled

of the Galerkin method in addition to the standard polynomial component. With such a function space available, it is a basic characteristic of all Galerkin methods that the method automatically tries to find an optimal solution using these shape functions. In this section, we first give a review of similar enrichment techniques applied to other physical phenomena. The general framework used for such a function enrichment is presented in Subsections 6.3.2.1 and 6.3.2.2, possible enrichment functions are reviewed in Subsection 6.3.2.3, and aspects related to the spatial and temporal adaptation of the method are discussed in Subsection 6.3.2.4. We comment on the potential regarding turbulence modeling with the present methodology in Subsection 6.3.2.5, which will be detailed in Chapters 7 to 10 using different approaches. Finally, aspects of implementing such a function enrichment are explained in Section 6.3.3.

6.3.1. Function Enrichment: Applications with High Gradients

The first framework for designing customized numerical methods was introduced by Melenk and Babuška [183] with their partition-of-unity method (PUM). Belytschko and Black [21] subsequently suggested a formalism that allows for the construction of a problem-tailored computational method in the application of crack propagation in solid mechanics, which is today known as extended FEM (XFEM) in the scientific community. An enrichment function representing an approximate analytical solution is usually used to extend the solution space of the method, besides the standard polynomial function space. For a comprehensive overview of the method we refer to the review articles in [22, 92]. Applications of this method in the field of fluid mechanics can be found in several academic examples such as enrichment with analytical high-gradient solutions of the convection-diffusion equation [1, 2, 252, 253], simulating a sharp corner in Stokes flow via an asymptotic expansion as enrichment [83] or resolving the bottom boundary layer of oceanic flow via a logarithmic enrichment function applied to a 1D water column [107]. A recent publication suggests an enrichment with modes obtained via proper orthogonal decomposition to resolve the boundary layer of a stochastically forced Burger's equation [43]. The general framework can also be used to resolve other features of the solution besides high gradients, such as jumps or kinks, and can even be used to cut elements (see, e.g., [228]). In general, the enrichment function is not prescribed as solution but the method 'chooses' the best solution among all functions available in its function space in a consistent manner.

The so-called discontinuous enrichment method (DEM) by Farhat and co-workers [69] with application to high-gradient enrichments for the convection-diffusion equation, e.g., in [70, 132, 133], represents another framework for constructing problem-tailored numerical schemes. A major distinction between these two methods is that XFEM is typically (but not necessarily) based on continuous functions both for the polynomial and the enrichment component, whereas in DEM the enrichment component is discontinuous at cell interfaces allowing for the static condensation of the enrichment via Lagrange multipliers. In addition, the enrichment function in the XFEM is usually (but not necessarily) weighted using a partition of unity given through standard polynomial shape functions, whereas the enrichment function is directly added to the solution in DEM.

In the present thesis, the XFEM framework is considered both within the continuous FEM and the DG method. Regarding DG, the cells are connected through appropriate fluxes as described in Chapters 3 and 5 instead of enforcing weak continuity as in the DEM. The XFEM may be considered more efficient within the particular matrix-free implementation described in Chapter 3 and due to the requirement of temporal adaptation of the shape functions in the context of turbulent flow simulations, which makes static condensation less attractive. The supplementary weighting of the enrichment function with standard polynomial shape functions introduce a higher level of flexibility in the function space of the XFEM. However, this weighting can be ‘switched off’ by considering a polynomial of degree zero, which would be equivalent to the DEM, if conditioning is an issue, as will become clear later. Despite these arguments in favor of the XFEM framework, wall modeling via function enrichment should be implemented in several methodologies in future work, including the DEM and the approaches in [29, 33, 219], in order to allow for a critical assessment of the efficiency for boundary layer enrichments.

6.3.2. Wall Modeling via Function Enrichment

Wall modeling via function enrichment is proposed and the necessary modeling steps are discussed in this subsection. The solution space of the method is capable of resolving high boundary layer gradients and adapts to local characteristics of the flow. This is done by extending the solution space with the help of an approximate analytical representation of the mean velocity profile given as the law-of-the-wall. With a solution space capable of resolving the high gradient, the solution is not prescribed but the numerical method is able to find an appropriate solution in the offered function space, given that the unresolved turbulence is modeled accurately. As a numerical method, a variant of the PUM or the XFEM is chosen as it provides a framework for designing such a customized space. In Subsection 6.3.2.1, it is shown how the enrichment is constructed in DG and in Section 6.3.2.2 in the continuous FEM. Subsequently, possible choices for enrichment functions are presented and a temporal adaptation algorithm are introduced.

The wall modeling approach achieves a drastic reduction in computational complexity through the large discretization cells employed in turbulent boundary layers. As a consequence, the cells near the wall are so large that not even the largest near-wall eddies are resolved. These turbulent motions have to be modeled by use of a statistical RANS-type model. In Subsection 6.3.2.5, we give an overview of the turbulence modeling approaches considered in the subsequent chapters.

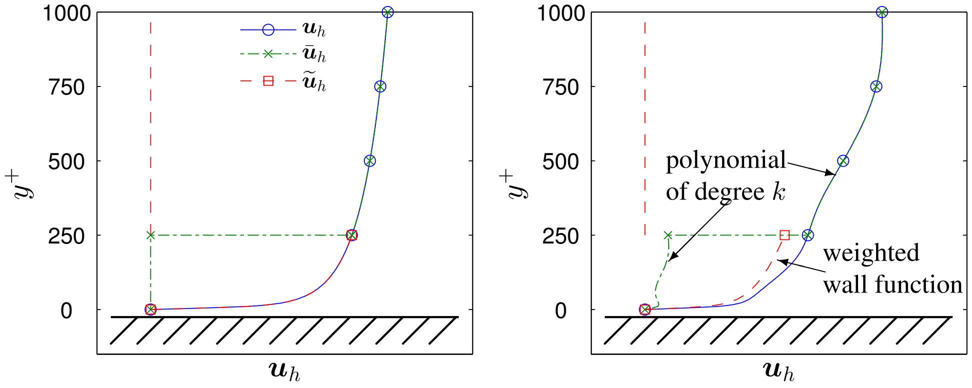


Figure 6.2.: High-order DG: composition of the velocity $\boldsymbol{u}_h$ in a boundary layer consisting of the polynomial and the enrichment component $\bar{\boldsymbol{u}}_h$ and $\widetilde{\boldsymbol{u}}_h$ for an ideal boundary layer given as Spalding's law (left) and a nonequilibrium boundary layer in a separated flow or with high pressure gradient (right). Symbols indicate cell boundaries, i.e., one cell near the wall is enriched.

6.3.2.1. Enriching the DG Solution Space

The standard polynomial function space of the DG method is extended by an additional component, yielding the following composition of the velocity variable $\boldsymbol{u}_h(\boldsymbol{x}, t)$:

$$\boldsymbol{u}_h(\boldsymbol{x}, t) = \bar{\boldsymbol{u}}_h(\boldsymbol{x}, t) + \widetilde{\boldsymbol{u}}_h(\boldsymbol{x}, t), \tag{6.1}$$

and assuming the direct sum decomposition of the underlying discrete solution spaces $\mathcal{V}_h^{\boldsymbol{u}} = \mathcal{V}_h^{\bar{\boldsymbol{u}}} \oplus \mathcal{V}_h^{\widetilde{\boldsymbol{u}}}$. Herein, the variable $\bar{\boldsymbol{u}}_h \in \mathcal{V}_h^{\bar{\boldsymbol{u}}}$ represents the polynomial function space that was exclusively used in Part I of this thesis, where it was defined as

$$\mathcal{V}_h^{\bar{\boldsymbol{u}}} = \{\bar{\boldsymbol{u}}_h \in (L^2)^d : \bar{\boldsymbol{u}}_h|_{\Omega_e} \in P_k(\Omega_e), \forall e \in \boldsymbol{\Omega}_h\} \tag{6.2}$$

using the notation introduced in Section 3.3.2. The polynomials P_k can both be defined through Lagrangian tensor-product elements as used in this work (see Section 3.5.1), through a modal basis, or any other kind of mixture between the two. The polynomial component $\bar{\boldsymbol{u}}_h(\boldsymbol{x}, t)$ is in each element represented by a polynomial FE expansion of degree k according to Section 3.5.1:

$$\bar{\boldsymbol{u}}_h(\boldsymbol{x}, t) = \sum_{B \in N^k} N_B^k(\boldsymbol{x}) \bar{\boldsymbol{u}}_B(t) \tag{6.3}$$

with shape functions N_B^k and DOFs $\bar{\boldsymbol{u}}_B(t)$. The enrichment space consists of an enrichment function $\psi(\boldsymbol{x}, t)$ times a polynomial of degree l:

$$\mathcal{V}_h^{\widetilde{\boldsymbol{u}}} = \{\widetilde{\boldsymbol{u}}_h \in (L^2)^d : \widetilde{\boldsymbol{u}}_h|_{\Omega_e} \in (\psi P_l(\Omega_e)), \forall e \in \widetilde{\boldsymbol{\Omega}}_h\}, \tag{6.4}$$

where $\widetilde{\boldsymbol{\Omega}}_h \subset \boldsymbol{\Omega}_h$ is a thin layer near the wall. As an enrichment function, we consider an *a priori* known approximate analytical or empirical solution to the underlying flow problem and several enrichment functions relevant for wall modeling are presented in Section 6.3.2.3. The enrichment function is the key ingredient to the overall approach, as it allows the numerical method to represent a solution of similar shape as the enrichment function with very coarse cells. The degree l can and should often be chosen independently of k and we consider $l = \{0, 1, 2\}$, whereas k should often be significantly higher due to reasons of efficiency, see Section 6.3.3.1. A degree of $l > 0$ weights the enrichment function with a polynomial, which adds flexibility to the approach in cases where the solution differs significantly from the enrichment function. If $l = 0$, the weighting polynomial becomes a constant function such that the enrichment is of similar form as in the DEM. Written in terms of shape functions, the enrichment expansion reads

$$\widetilde{\boldsymbol{u}}_h(\boldsymbol{x}, t) = \psi(\boldsymbol{x}, t) \sum_{B \in N^l} N^l_B(\boldsymbol{x}) \widetilde{\boldsymbol{u}}_B(t). \tag{6.5}$$

The enrichment is only taken into account where it is necessary. The thickness of the layer $\widetilde{\Omega}_h$ can either be user-specified (Chapter 7) or be determined by an adaptation algorithm, which evaluates if the enrichment is necessary in each cell and at each time instant (Chapters 8 to 10). In standard boundary layers, it is sufficient to enrich a few layers of cells in the vicinity of the no-slip boundary and we set $\widetilde{\boldsymbol{u}}_h(\boldsymbol{x}, t) = \boldsymbol{0}$ outside of $\widetilde{\Omega}_h$. The number of required cells depends significantly on k. If $k > 2$, a single layer of enriched elements would commonly be sufficient, as the high-order polynomial component can resolve the quickly decaying velocity gradient, whereas several element rows may be enriched if $k = 1$ and 2. Finally, we note that the number of additional DOFs required by the enrichment is often very small, due to the locality of the problem and in particular if $k \gg l$.

The composition of the resulting function space is illustrated in Figure 6.2 for an equilibrium boundary layer and for a nonstandard boundary layer profile, such as in separated flows. In an equilibrium boundary layer, the velocity profile may to the largest extent be captured by the enrichment while the full flexibility of a high-order polynomial plus a weighted wall function is available if the solution demands.

We discuss the treatment of the remaining variables p and $\widetilde{\nu}$, the latter in case the SA model is considered. In typical boundary layers, high-gradient solutions do not occur in the pressure variable, thus we employ only the standard polynomial function space of degree k to discretize $p_h \in \mathcal{V}^p_h$. Regarding the SA model, we have observed that the eddy-viscosity-like working variable results in a linear distribution of $\widetilde{\nu} \sim y^+$ between the wall and the outer edge of the logarithmic layer under equilibrium assumptions [134], see also the profiles plotted in Figure 5.3. We therefore assume that a polynomial element of degree k adds sufficient flexibility to $\widetilde{\nu}_h \in \mathcal{V}^{\widetilde{\nu}}_h = \mathcal{V}^p_h$ in order to capture more general profiles of $\widetilde{\nu}$ in nonequilibrium boundary layers as well. Finally, researchers may be interested in employing the present function-enrichment approach in the context of other turbulence models or flow physics. Considering the compressible Navier–Stokes equations, where the energy equation is discretized in addition, the approach does not require any modification as long as the energy varies smoothly within the boundary layer (which is always the case on adiabatic walls). The energy variable could then be resolved by the polynomial component analogous to the pressure. Likewise, other RANS turbulence models may be considered if the turbulence model quantities, which have to be discretized, vary slowly such that they can be resolved sufficiently with a polynomial of degree k in the first off-wall cells.

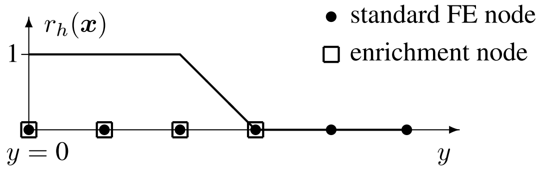


Figure 6.3.: Enriching the solution space of the continuous FEM: Ramp functions are a possibility of defining a conforming function space at the interface between enriched and nonenriched cells.

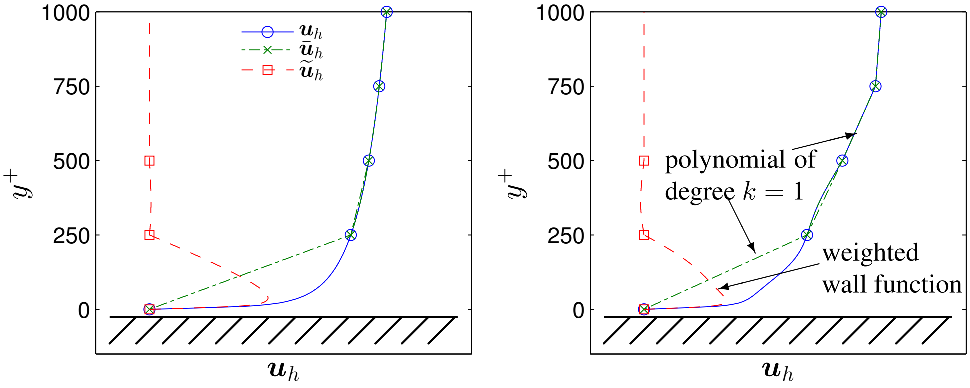


Figure 6.4.: Continuous FEM: composition of the velocity $\boldsymbol{u}_h$ in a boundary layer consisting of the polynomial and the enrichment component $\bar{\boldsymbol{u}}_h$ and $\widetilde{\boldsymbol{u}}_h$ for an ideal boundary layer given as Spalding's law (left) and a nonequilibrium boundary layer in a separated flow or with high pressure gradient (right). Symbols indicate cell boundaries, i.e., (usually at least) two cells near the wall are enriched of which one is the blending layer.

We note that the particular construction of the enrichment shape functions in Equation (6.5) multiplies the enrichment function by a standard FE partition-of-unity. This formulation enables a relatively simple implementation of the enrichment in common FE libraries such as the deal.II library [9], since the shape functions do not have to be modified on a basic level, but are recombined subsequently to existing evaluation routines. A separate section of this thesis explains the implementation of the enrichment in an existing solver within the deal.II library with minimal modification of the original code (see Section 6.3.3.4). The aspect of straightforward implementation is one of the reasons why the DG method is particularly suited for including enrichments, whereas the continuous FEM requires additional steps in the computation of the shape functions. These modifications are discussed in the subsequent section.

6.3.2.2. Enriching the Solution Space of the Continuous FEM

The above definition of the enrichment component of the velocity solution has to be modified in order to be compatible with the continuous FEM. It is a basic characteristic of the DG method that the cells are weakly coupled, which inherently allows different solution spaces in neighboring elements. In contrast, the continuous FEM connects neighboring cells by sharing the DOFs located on the faces and edges (see Section 2.2.2.3). This characteristic has to be taken into account at the interface where the enriched cells end and the standard polynomial cells begin, in order to obtain a conforming discretization. Such a blending of function spaces may be achieved by multiplying the enrichment function with a ramp function $r_h(\boldsymbol{x})$, which exhibits a linear distribution in the element layer in between fully enriched cells and nonenriched cells [91], according to Figure 6.3. Furthermore, it is desirable that the enrichment vanishes on the nodes of the shape functions, which reduces issues in the blending layer and simplifies postprocessing, as it is sufficient to postprocess the nodal values of the standard FE space. This modification also makes the enrichment similar to the concept of bubble functions (the enrichment does not vanish entirely on the cell faces, however). The enrichment FE expansion becomes

$$\widetilde{\boldsymbol{u}}_h(\boldsymbol{x},t) = \sum_{B \in N^l} N_B^{c,l}(\boldsymbol{x})(\psi(\boldsymbol{x},t) - \psi(\boldsymbol{x}_B,t))r_h(\boldsymbol{x})\widetilde{\boldsymbol{u}}_B(t), \tag{6.6}$$

with globally defined continuous shape functions $N_B^{c,l}$ of degree l. It is noted that the ramp function ‘distorts’ the enrichment space in the cell where the linear profile is active, such that the enrichment is less effective. Therefore, at least two layers of elements should be enriched with this approach. Such a decomposition of the solution into a polynomial and an enrichment component for the continuous FEM is depicted in Figure 6.4 and it is apparent that the enrichment solution vanishes on the nodes.

Ramp functions are not the only possibility of blending the function spaces. For example, the domains may be coupled weakly using a mortar approach [65] or a cut-FEM method, which couples two fluid domains with similar terms as two neighboring cells in DG [227]. As an alternative, constraint matrices may be used to achieve the blending of function spaces [57] within the deal.II library.

6.3.2.3. Wall Functions as Enrichment Function

An appropriate choice of the enrichment function $\psi(\boldsymbol{x},t)$ is the key feature of the overall methodology. This function provides the opportunity to include information *a priori* known about boundary layers in the function space without prescribing the solution itself. We propose to enrich the function space with an empirical single analytic function for the law-of-the-wall including the viscous sublayer and inner layer. This choice allows the Galerkin method to resolve the ensemble-averaged velocity profile of the turbulent boundary layer as well as the laminar sublayer, and thus the prediction of the accurate skin friction, with very coarse meshes. In addition, the resolution requirement becomes independent of wall units in a wide y^+-range.

Such mean velocity profiles have for example been suggested by Reichardt [217], van Driest [63], and more widely known by Spalding [245]. They satisfy the boundary conditions at the wall $\boldsymbol{u}(y=0) = 0$ and $\frac{\partial u^+}{\partial y^+}|_{y=0} = 1$ (see Section 2.1.3) which is necessary in order to predict the wall shear stress accurately. Thus, in the limit of $y^+ \to 0$, we get $u^+ = y^+$.

In the following, we give an overview of four wall functions, which are highly relevant as an enrichment function for the present wall modeling approach: Spalding’s law and van Driest’s law, which will be employed as an enrichment function in the subsequent chapters, as well as

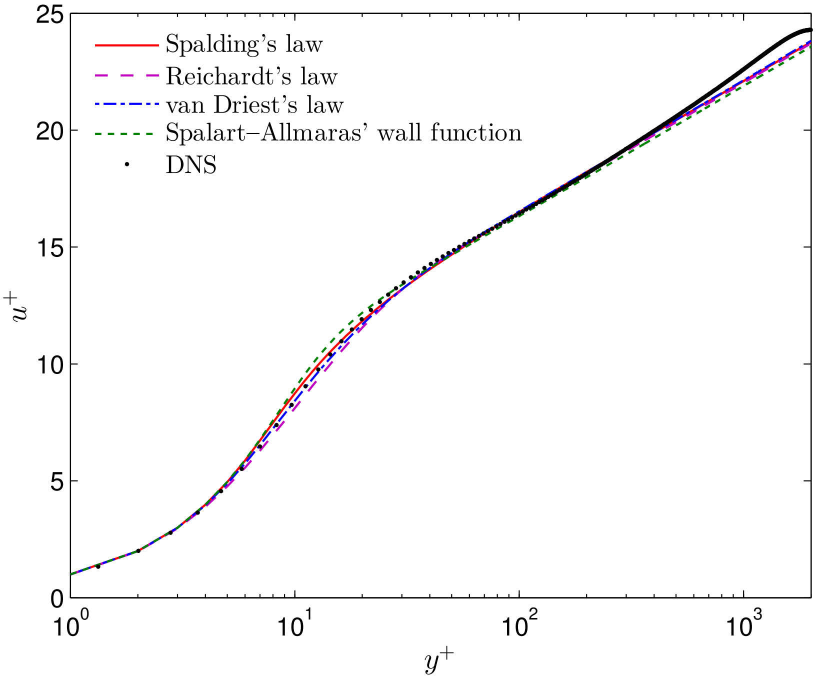


Figure 6.5.: Comparison of wall functions.

Reichardt's law and the wall function given through the SA model. These four wall functions are plotted and compared to DNS data of turbulent channel flow by [116] in Figure 6.5.

Spalding's Law. One of the enrichment functions proposed in this work is a minor modification of the wall function by Spalding, which is implicitly given as

$$y^+ = \frac{\psi}{\kappa} + e^{-\kappa B}(e^{\psi} - 1 - \psi - \frac{\psi^2}{2!} - \frac{\psi^3}{3!} - \frac{\psi^4}{4!}), \tag{6.7}$$

where the common formulation is recovered with $u^+ = \frac{\psi}{\kappa}$. The constants $\kappa = 0.41$ and $B = 5.17$ according to Section 2.1.3 are employed throughout this work. Details on how Equation (6.7) is evaluated numerically are given in Appendix A.1. The velocity profile shown in Figure 6.5 matches the DNS data very well in the laminar sublayer and the log-layer, whereas the distributions exhibit differences in the buffer layer. Further, the profiles deviate in the bulk flow, which is expected, since the log-law is only valid up to $y/\delta \sim 0.2$. If this limitation is considered to be too stringent, the modifications by Coles [51] and Dean [58] may be taken into consideration, which provide a model for the bulk flow.

Van Driest's Law. If a RANS eddy viscosity approach is used as a turbulence model, it would be highly relevant to employ the particular wall function as an enrichment function which is consistent with the RANS model. This wall function is obtained by considering a plain boundary layer in equilibrium state without pressure gradients. As a result, the method achieves full consistency between the RANS model and the solution space. Indeed, preliminary numerical investigations with the model proposed in Chapter 10 have shown that the predictions of the wall shear stress could be enhanced in accuracy and robustness if the consistent wall function is used instead of Spalding's law. As van Driest's [63] extension of Prandtl's algebraic near-wall RANS model (see Section 2.3.3) is employed in that chapter, we consider the corresponding wall function in the following, specified in the same publication. The enrichment function is defined as

$$\psi = \int_0^{y^+} \frac{2\,\mathrm{d}y^+}{1+\sqrt{1+\left(2\kappa y^+(1-e^{(-y^+/A^+)})\right)^2}}, \tag{6.8}$$

with $u^+ = \psi$, and we take $A^+ = 26$. Since the enrichment function has to be evaluated on every quadrature point of the numerical method in a layer of elements near the wall, we describe a possible approach in Appendix A.2. In Figure 6.5, this wall function predicts a lower velocity magnitude in the buffer layer, but the differences are moderate.

Spalart–Allmaras' Wall Function. Again due to reasons of consistency, it may be relevant to consider the wall function which results from the SA model (see Capter 5) in a wall-modeled RANS or DES approach. An explicit analytical expression for this wall function was derived in [7] and is given as

$$\begin{aligned} u^+ = \overline{B} + c_1 \ln\left((y^+ + a_1)^2 + b_1^2\right)\right) - c_2 \ln\left((y^+ + a_2)^2 + b_2^2\right) \\ - c_3\, \text{atan2}(b_1, y^+ + a_1) - c_4\, \text{atan2}(b_2, y^+ + a_2), \end{aligned} \tag{6.9}$$

with the parameters

$$\begin{aligned} \overline{B} &= 5.0333908790505579, & & \\ a_1 &= 8.148221580024245, & b_1 &= 7.4600876082527945, \\ a_2 &= -6.9287093849022945, & b_2 &= 7.468145790401841, \\ c_1 &= 2.5496773539754747, & c_2 &= 1.3301651588535228, \\ c_3 &= 3.599459109332379, & c_4 &= 3.6397531868684494, \end{aligned}$$

where atan2 is the four-quadrant inverse tangent function. Figure 6.5 reveals differences to Spalding's law and van Driest's law in the buffer layer, where the SA model results in a higher velocity magnitude. Also, this wall function converges to a different logarithmic wall law, namely $u^+ = \frac{1}{\kappa}\ln(y^+ + \frac{1}{\kappa}) + \overline{B}$, and the difference in the constant $\overline{B} \approx 5.03$ is noticeable in Figure 6.5 through a lower velocity profile in the logarithmic layer.

Reichardt's Law. Finally, Reichardt's wall function [217] is given according to

$$u^+ = \frac{1}{\kappa}\ln(1+\kappa y^+) + c_1\left(1 - e^{-y^+/11} - \frac{y^+}{11}e^{-0.33y^+}\right), \tag{6.10}$$

and the constants are adapted to match the present choice of κ and B, which yields $c_1 = 7.34$ instead of $c_1 = 7.8$ with the formula given in the original publication. The underprediction of the velocity inside the buffer layer in comparison to the DNS shown in Figure 6.5 is noticeable.

Other Wall Functions. There is a large number of alternative wall functions available in the literature. For example, wall functions with advanced capabilities in separated flows have been proposed in [143, 207, 235], or even the expected velocity profile in regions of backflow may be modeled [238]. Further, other RANS models than the SA model result in different velocity profiles and a derivation of the analytical profile may not always be possible. For such applications, a procedure described by Kalitzin et al. [134] would be relevant, which allows the implementation of arbitrary profiles via tables. Finally, it was pointed out by an anonymous reviewer in the context of the work presented in Chapter 8 that a function enrichment approach may be promising for RANS in regions just downstream of stagnation points, where the boundary layer is laminar. This is due to the fact that the newly developing boundary layer is extremely thin and typical discretizations employed in RANS computations are usually too coarse to capture this boundary layer. If an analytical profile for such regions is not available, the approach by Kalitzin et al. could be considered as well.

All four wall functions presented above are equally well-suited as an enrichment function in turbulent boundary layers. In general, it can be said that the particular shape inside the buffer layer is less important, since the detailed velocity profile in the buffer layer is usually not of interest for CFD engineers and researchers. However, as noted earlier, the wall function should be consistent with the particular eddy viscosity approach considered, meaning that the wall function should match the exact velocity profile that results from the turbulence model in a plane equilibrium boundary layer as close as possible.

6.3.2.4. Spatial and Temporal Adaptation

It is the nature of wall functions for turbulent boundary layers to be universal in the wall coordinate $y^+ = \frac{y}{\nu}\sqrt{\frac{\tau_w}{\rho}}$, see Section 2.1.3. The wall coordinate scales the enrichment function in wall-normal direction to match the local wall shear stress τ_w. As a consequence, the wall shear stress used to compute the wall function has to be adapted spatially and temporally in the numerical method in order to take the local features of the flow as well as their temporal evolution into account in the model. For that purpose, the wall shear stress is discretized based on linear continuous shape functions $N_B^{c,m}$ $(m = 1)$. Therein, the wall shear stress is computed on each node B in $N^{c,m}$ with

$$\tau_{w,B} = \frac{\| \int_{\partial\Omega^D} N_B^{c,m}(\boldsymbol{x}) \boldsymbol{\tau}_w \, \mathrm{d}A \|}{\int_{\partial\Omega^D} N_B^{c,m}(\boldsymbol{x}) \, \mathrm{d}A}, \tag{6.11}$$

where $\boldsymbol{\tau}_w$ is the wall shear stress vector, which may be computed as $\boldsymbol{\tau}_w = \rho\nu \left(\nabla \boldsymbol{u}_h\right) \cdot \boldsymbol{n}$. Note that we compute the integral for each component of the wall shear stress vector and subsequently take the vector norm. The nodal values are interpolated using

$$\tau_{w,h} = \sum_{B \in N^{c,m}} N_B^{c,m} \tau_{w,B}. \tag{6.12}$$

It is observed that the quantity $\tau_{w,h}$ includes a spatial coarsening of the wall shear stress distribution if $k > 1$ due to $m = 1$. This coarsening is physically motivated and required for the eddy-resolving simulation approaches considered subsequently, since the wall function represents a relation for the mean quantities, i.e., the mean wall shear stress gives a mean velocity profile. Without this coarsening, one would observe a statistical overprediction of $\tau_{w,h}$ as the velocity vector does not always point in the global streamwise direction. Similar approaches have been considered in the literature, for example, Schumann [231] averaged the wall shear stress over the walls of channels and pipes. As we are seeking to develop a flexible tool, which also can account for local variations present in complex flow geometries, we average the stresses only locally by the different choice of the polynomial degrees if $k > 1$ and $m = 1$. A continuous polynomial results in a smoother variation of $\tau_{w,h}$ (and a smoother variation of the function space) in comparison to, for example, a constant distribution (DG of degree $m = 0$). Additionally, high-order polynomials may result in large localized variations of $\tau_{w,h}$ inside one cell, which would be demanding regarding the efficient numerical integration of weak forms. The traction computed at the no-slip nodes is communicated to the respective off-wall nodes in $\widetilde{\Omega}_h$, where the node pairs are determined via the shortest distance to the respective node on the wall, analogous to the following definition of the wall distance.

The wall distance is likewise defined using a finite element expansion in order to facilitate the evaluation of the enrichment shape functions. The discrete distance from the wall y_h is given through the weights y_B, which represent the shortest distance between the current node B and the closest wall node and are interpolated using the same linear continuous shape functions as for the wall shear stress:

$$y_h = \sum_{B \in N^{c,m}} N_B^{c,m} y_B. \tag{6.13}$$

We note that this definition of the wall distance is robust even for surfaces where the wall-normal vector is not unique.

For the application within the incompressible Navier–Stokes equations, the first and possibly the second derivative of the enrichment component with respect to Cartesian coordinates are required. Their derivation is straight forward and included in Appendix A.3.

The wall shear stress is evaluated prior to every time step using the velocity solution of the previous time step to allow for temporal adaptivity. As an alternative, one may compute the wall shear stress using an extrapolated velocity solution, for example via extrapolation formulas employed in the right hand side of the explicit step in Equation (3.2), for improved accuracy in time-resolved computations. Since the discrete function space varies from time step to time step, the velocity based on the old function space $\boldsymbol{u}_h^{n,\text{old}}$ is projected onto the new FE space $\boldsymbol{u}_h^{n,\text{new}}$ via local L^2 projection according to

$$\left(\boldsymbol{v}_h^{n,\text{new}}, \boldsymbol{u}_h^{n,\text{new}}\right)_{\Omega_e} = \left(\boldsymbol{v}_h^{n,\text{new}}, \boldsymbol{u}_h^{n,\text{old}}\right)_{\Omega_e}, \tag{6.14}$$

with n indicating the current time step number. Supplementary vectors required by the time integration scheme, such as older velocity components or the vorticity vector, are projected using the same approach or recomputed, if possible.

6.3.2.5. Turbulence Modeling

Since the full incompressible Navier–Stokes equations are solved in the boundary layer including the no-slip boundary condition, the unresolved turbulent motions have to be modeled. In other words, wall modeling via function enrichment is a spatial discretization technique, no turbulence model. Note that this is a fundamental difference to other applications of high-gradient enrichments. For example, in crack-tip enrichments [22], the high gradient occurs due to the singularity in the geometry, and in high-Péclet-number convection-diffusion problems [253], the high gradient is the solution of the underlying partial differential equation. In contrast, the Navier–Stokes equations do not show turbulent boundary layer profiles if the turbulent motions are not resolved or modeled (see Section 6.1). In the subsequent Chapters 7 to 10, the wall model is applied to different scenarios including RANS, DES, and WMLES.

RANS. The wall modeling approach may be employed in a pure RANS framework, where all turbulent motions are modeled by use of a statistical turbulence model. As a consequence, no modifications of the turbulence modeling approach are required, and the enrichment approach reduces the number of required grid points in wall-normal direction. This approach is detailed in Chapter 8.

DES. If the classical detached-eddy simulation methodology is considered, the turbulence model takes the unresolved turbulent motions in the near-wall layer into account and provides a statistical model in those cells which are too coarse to resolve the energy containing vortices. DES models automatically switch between the RANS and LES mode according to the spatial resolution provided by the scheme, such that no further modification is required in the turbulence modeling approach. This method is applied in Chapter 9.

Wall-Modeled LES and Multiscale Wall Modeling. The enrichment approach may also be interpreted as a separation of the solution vector into two scale groups. The two scales are given by the standard polynomial component $\bar{\boldsymbol{u}}_h$ and the enrichment component $\widetilde{\boldsymbol{u}}_h$. With regard to WMLES, the standard polynomial scale resolves large eddies that are at least of the size of the characteristic element length. The enrichment scale, however, represents flow features in a statistical sense where the large eddies cannot be resolved due to the coarseness of the mesh. This interpretation and the framework of the variational multiscale method allow tailored turbulence models for each of the scales in separate equations – a RANS turbulence model for the RANS scale and an LES turbulence model for the LES scale. This approach is employed in Chapter 10 including a rigorous derivation of the respective models. Several alternative WMLES techniques are thinkable within the framework of wall modeling via function enrichment. In Chapter 7, a turbulence model is presented which relies on a residual-based method in conjunction with a structural LES turbulence model in the outer boundary layer within the continuous FEM.

6.3.3. Implementation of High-Gradient Enrichments

The present wall modeling approach may be embedded in any steady or unsteady, compressible or incompressible, RANS, DES, or LES flow solver based on a Galerkin method, i.e., the DG method or the continuous FEM. In this section, it is explained how weak forms including the enrichment shape functions can be evaluated. Several algorithmic aspects of the overall solution procedure are detailed, including the matrix-free application of the inverse mass matrix and the computation of the residuals in iterative solvers. Further details on this work will be given in a separate article [160]. Finally, a strategy is presented that allows the straightforward implementation of the wall model in an existing code written in an object-oriented programming language.

Table 6.3.: Guide lines for the choice of quadrature rules using Gaussian integration. The quantity n_q denotes the number of quadrature points in wall-normal direction.

$\Delta y^+_{1e} <$	n_q
90	8
110	10
130	12
160	14
200	17
300	20
500	35
2,500	200
5,000	300

6.3.3.1. Evaluation of Weak Forms

If the enriched function space in Equation (6.1) is inserted in the Galerkin formulations presented in Sections 3.3.2 and 5.3.1, the arising integrals contain polynomial and nonpolynomial terms, the latter due to the nonpolynomial character of the considered enrichment functions. The polynomial terms can be exactly evaluated on affine cells using Gaussian quadrature with $k+1$ quadrature points for linear terms and $\lfloor \frac{3k}{2} \rfloor + 1$ points for nonlinear terms as it was described in Section 3.5.1. These quadrature formulas are also employed in all nonenriched cells and faces.

The terms arising from the enrichment shape functions require special attention due to the nonpolynomial shape functions. As discussed in [92], it may be efficient to construct problem-tailored quadrature formulas for applications of high-gradient enrichments, especially regarding the high-gradient direction, i.e., the wall-normal direction. There are several approaches for computing tailored quadrature formulas, for example moment-fitting, see [191, 247]. This idea was assessed within the present study in a student project [163]. Therein, it was found that quadrature formulas that use the enrichment function and its spatial derivatives as basis functions indeed reduce the required number of quadrature points in comparison to Gaussian quadrature. However, the construction of such tailored quadrature formulas is expensive and the associated equation systems are badly conditioned. In view of a function space that changes every time step and varies from element to element, it does not seem to be practicable to recompute tailored quadrature rules dynamically. Further, the tabulation of such quadrature formulas is challenging, since the width of the cells in y^+- units is a continuous function. As a very promising alternative, an adaptive algorithm may be employed in order to integrate the nonpolynomial terms, for example the one described in [190]. Further, hexahedral cells could be subdivided in the wall-normal direction into several layers, which are each integrated with Gaussian quadrature. Such an approach would have the potential of significantly reducing the required number of quadrature points, as the slices could be thinner in the region of the high gradient near the wall (see the evaluation algorithm of van Driest's law in Appendix A.2).

In this work, we employ Gaussian quadrature at higher order of accuracy for the integration of all enriched cells. Typically, at least $n_q = 8$ quadrature points are necessary in the wall-normal direction, and the requirement increases if the enriched element spans a wider range in y^+-units. As shown in Chapter 7, the quadrature rules may be direction-dependent as a higher accuracy is required in the wall-normal direction. To this end, $n_q = 4$ to 6 quadrature points were sufficient in each wall-parallel dimension for the simulations carried out in Chapter 7. Based on the extensive experience gained in hundreds of simulations carried out in this thesis, rough guidelines for the choice of the required number of quadrature points are shown in Table 6.3. However, the quadrature requirements vary considerably for different pairs of k, l and m. It is noted that the present enrichment shape functions are well-behaved regarding underintegration, as the algorithm indicates underintegration through an elevated number of solver iterations (see Section 6.3.3.3) or minor oscillations in the wall shear stress from time step to time step long before the underintegration is visible in the averaged results. Further, these requirements on the quadrature motivate an application of the enrichment with moderately high polynomial degrees $2 \lesssim k \lesssim 6$. If $k = 1$, the excess in quadrature points due to the enrichment is very high. In contrast, if $k = 4$, 5 quadrature points per dimension would be used in linear and 7 in nonlinear terms, so only a few additional points in wall-normal direction would be necessary (considering $\Delta y^+_{1e} \lesssim 200$). Finally, despite the successful application of these quadrature formulas, it can be said that the efficient integration of the enriched weak forms is currently the bottleneck of the methodology with respect to the independence of the resolution requirements with wall units, i.e., the cell size is somewhat limited in the y^+-direction. Further research in this field should aim at an adaptive quadrature algorithm that reduces the required number of quadrature points.

6.3.3.2. Matrix-Free Application of the Inverse Mass Matrix

The inverse mass matrix $\boldsymbol{M}^{-1}$ is applied repeatedly in each time step in the solver presented in Chapters 3 and 5, and it is considered a crucial ingredient in other explicit and semi-explicit numerical schemes. Besides the use for the explicit convective step, we use inverse mass preconditioners for the local projection solver as well as the viscous solver, and the operator is used to compute L^2 projections; the use of the inverse mass operator accumulates to about 25 applications per time step. The mass matrix in DG is block diagonal and an inverse mass operator can therefore be applied in each cell separately. With regard to enriched elements, we consider the problem of

$$\begin{bmatrix} \bar{\boldsymbol{U}} \\ \widetilde{\boldsymbol{U}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{M}^{\bar{V}\bar{U}} & \boldsymbol{M}^{\bar{V}\widetilde{U}} \\ \boldsymbol{M}^{\widetilde{V}\bar{U}} & \boldsymbol{M}^{\widetilde{V}\widetilde{U}} \end{bmatrix}^{-1} \begin{bmatrix} \bar{\boldsymbol{R}} \\ \widetilde{\boldsymbol{R}} \end{bmatrix}, \tag{6.15}$$

where the input and output vectors $\boldsymbol{R}$ and $\boldsymbol{U}$ are split into two parts, the polynomial DOFs, indicated with the superscript $\bar{(\cdot)}$, and the enrichment DOFs, indicated with $\widetilde{(\cdot)}$. Accordingly, the mass matrix is split into four blocks corresponding to the polynomial and enrichment rows and columns. It is noted that the matrix-free operator in [161], which is used in the nonenriched cells, corresponds to $(\boldsymbol{M}^{\bar{V}\bar{U}})^{-1}$ and an extension of the idea of that operator to enriched cells is not possible. In the classical spectral DG method, where the nodes and quadrature points coincide (i.e., they are co-located), the mass matrix is even diagonal and its inverse is trivial; this characteristic is lost in enriched cells. Due to these reasons, the development of an efficient application of the inverse mass operator is crucial for the efficiency of the overall methodology. Two methods of computing and applying the inverse mass operator have been developed: a matrix-based variant that uses LU factorization for computing and applying the inverse mass matrix and a matrix-free variant via a Schur complement. These two approaches are detailed and compared in the following.

Matrix-Based via LU Factorization. Since the mass matrix is block-diagonal, the inverse may be calculated on each enriched element independently. In this variant, a scalar mass matrix is precomputed for each element ahead of every time step and an LU factorization is used for applying the action of its inverse on each velocity component. To this end, it is noted that the higher-degree quadrature rules described in the previous section are used to compute the integral of the weak form. The complexity of such a matrix-based evaluation is $\mathcal{O}((k+1)^6)$ rather than $\mathcal{O}((k+1)^4)$ in the matrix-free case. In addition, it is expected that a matrix-based application of the inverse is computationally bound by the speed of the memory access on modern computers, see Section 3.5.

Matrix-Free via Schur Complement. An alternative strategy for the action of the application of the inverse mass matrix has been developed. The method is based on two considerations: Firstly, the matrix terms that consist only of polynomial terms should be integrated using $k+1$ quadrature points per space dimension, which are sufficient for the exact integration of these terms on affine cells. This aspect is crucial since the polynomial matrix block $\boldsymbol{M}^{\bar{\boldsymbol{V}}\bar{\boldsymbol{U}}}$ has many more matrix entries than the remaining rows and columns, so the largest portion of the cost comes from the evaluation of the polynomial shape functions on the high quadrature rules in the matrix-based approach. Secondly, as many terms as possible should be evaluated in a matrix-free manner in order to shift effort to arithmetics rather than loading large data structures from memory, see Section 3.5.

By making use of an inner Schur complement, the action of the matrix inverse may be computed by a scheme proposed in [161] in the context of the HDG method. The DOFs of the enriched component of the vector are first calculated via

$$\widetilde{\boldsymbol{U}} = \left(\boldsymbol{M}^{\widetilde{\boldsymbol{V}}\widetilde{\boldsymbol{U}}} - \boldsymbol{M}^{\widetilde{\boldsymbol{V}}\bar{\boldsymbol{U}}}\left(\boldsymbol{M}^{\bar{\boldsymbol{V}}\bar{\boldsymbol{U}}}\right)^{-1}\boldsymbol{M}^{\bar{\boldsymbol{V}}\widetilde{\boldsymbol{U}}}\right)^{-1}\left(\widetilde{\boldsymbol{R}} - \boldsymbol{M}^{\widetilde{\boldsymbol{V}}\bar{\boldsymbol{U}}}(\boldsymbol{M}^{\bar{\boldsymbol{V}}\bar{\boldsymbol{U}}})^{-1}\bar{\boldsymbol{R}}\right), \tag{6.16}$$

and the DOFs of the polynomial component are obtained by

$$\bar{\boldsymbol{U}} = \left(\boldsymbol{M}^{\bar{\boldsymbol{V}}\bar{\boldsymbol{U}}}\right)^{-1}\left(\bar{\boldsymbol{R}} - \boldsymbol{M}^{\bar{\boldsymbol{V}}\widetilde{\boldsymbol{U}}}\widetilde{\boldsymbol{U}}\right). \tag{6.17}$$

Further details on this solution strategy are given in the following:

- The matrix blocks $\boldsymbol{M}^{\widetilde{\boldsymbol{V}}\widetilde{\boldsymbol{U}}}$, $\boldsymbol{M}^{\widetilde{\boldsymbol{V}}\bar{\boldsymbol{U}}}$, and $\boldsymbol{M}^{\bar{\boldsymbol{V}}\widetilde{\boldsymbol{U}}}$ include the enrichment and have to be computed using quadrature rules with more points. Since the mass matrix is symmetric with $\boldsymbol{M}^{\widetilde{\boldsymbol{V}}\bar{\boldsymbol{U}}} = (\boldsymbol{M}^{\bar{\boldsymbol{V}}\widetilde{\boldsymbol{U}}})^T$, it is sufficient to compute the matrix columns of the blocks $\boldsymbol{M}^{\bar{\boldsymbol{V}}\widetilde{\boldsymbol{U}}}$ and $\boldsymbol{M}^{\widetilde{\boldsymbol{V}}\widetilde{\boldsymbol{U}}}$ of a scalar mass matrix, which equals one single matrix column in the case of $l = 0$ and 8 matrix columns in the case of $l = 1$. In the algorithm, these matrix columns are precomputed for all enriched cells once per time step and stored.

- The matrix $\left(\boldsymbol{M}^{\widetilde{\boldsymbol{V}}\widetilde{\boldsymbol{U}}} - \boldsymbol{M}^{\widetilde{\boldsymbol{V}}\bar{\boldsymbol{U}}}\left(\boldsymbol{M}^{\bar{\boldsymbol{V}}\bar{\boldsymbol{U}}}\right)^{-1}\boldsymbol{M}^{\bar{\boldsymbol{V}}\widetilde{\boldsymbol{U}}}\right)$ is computed by applying the matrix-free polynomial inverse $\left(\boldsymbol{M}^{\bar{\boldsymbol{V}}\bar{\boldsymbol{U}}}\right)^{-1}$ on each column of the precomputed matrix block $\boldsymbol{M}^{\bar{\boldsymbol{V}}\widetilde{\boldsymbol{U}}}$. The resulting matrix is of the dimensions 1×1 for $l = 0$ and 8×8 for $l = 1$ and the inverse

$\left(\boldsymbol{M}^{\tilde{\bar{V}}\tilde{\bar{U}}} - \boldsymbol{M}^{\tilde{\bar{V}}\bar{U}} \left(\boldsymbol{M}^{\bar{V}\bar{U}}\right)^{-1} \boldsymbol{M}^{\bar{V}\tilde{\bar{U}}}\right)^{-1}$ has to be computed explicitly. This is done once per time step for each enriched element via Gauss–Jordan elimination and the result is stored. Again in the classical spectral DG method, the diagonal inverse mass matrix block $\left(\boldsymbol{M}^{\bar{V}\bar{U}}\right)^{-1}$ is obtained by element-wise inverting the diagonal.

- When the equations are applied as an inverse mass operator, the polynomial inverse $(\boldsymbol{M}^{\bar{V}\bar{U}})^{-1}$ has to be applied twice (once in Equation (6.16) and once in Equation (6.17)), which enables the use of the standard quadrature rules and the same matrix-free inverse mass operator for these terms as in the nonenriched elements according to [161].

In summary, the resulting algorithm consists of two steps. Once in each time step, the matrix columns corresponding to the enrichment DOFs are precomputed and stored. Furthermore, a small local matrix is computed and inverted explicitly. The action of applying the matrix inverse requires two matrix-free applications of the polynomial inverse mass matrix block plus some further terms based on the precomputed matrices, the cost of which is small. Note that four cells with individual parameters are processed simultaneously in the present implementation via vector instructions (on the Intel Sandy-Bridge and Haswell CPU architectures) within the deal.II library [9].

Performance Evaluation. The two inverse mass operators are in the following thoroughly compared with regard to their performance. To this end, a computational setup similar to the turbulent channel flow case in Section 4.2 is used, which consists of 32 cells at varying polynomial degrees $k = \{2, 3, 4, 6, 8\}$ and $l = \{0, 1\}$, which are all enriched using van Driest's law. The physical parameters are chosen such that $\mathrm{Re}_\tau = 100$. For this case the first off-wall cell spans a relatively small y^+-range and thus ensures low requirements on the quadrature. Short simulations are run of approximately three flow-through times, and the performance numbers are averaged over several hundred time steps. The numerical experiments were carried out on a single core of an Intel Haswell CPU at 2.4GHz taking turbo boost into account.

Figure 6.6 shows the performance of the two inverse mass operators regarding their setup phase (one-time cost per time step). Herein, the performance is measured in throughput in terms of DOFs/s and the graphs are plotted over the number of quadrature points used for the enrichment terms n_q per space dimension, which is varied in the range of $[6, 30]$ (the total number of quadrature points is n_q^3 per cell). In this graph, it may be seen that the throughput of all curves reduces with an increasing number of quadrature points, which is an expected behavior. The matrix-free approach yields a larger throughput for higher polynomial degrees k for a fixed number of enrichment quadrature points, whereas this relation is inverted for the LU-based approach. The setup step takes approximately one to two orders of magnitude longer for the matrix-based LU approach than for the matrix-free approach; this behavior is expected since only 1 matrix column has to be computed in the matrix-free case for $l = 0$ and 8 columns for $l = 1$, and the cheap matrix-free polynomial inverse needs to be applied with the same frequency. In contrast, the full mass matrix for a scalar consists of $(k + 1)^3 + (l + 1)^3$ matrix columns, which have to be computed in the LU-based method. Accordingly, the setup phase for the matrix-free method should take approximately 8 times longer in the case of $l = 1$ compared to $l = 0$, whereas the dependence of the performance of the LU-based method on l is minor.

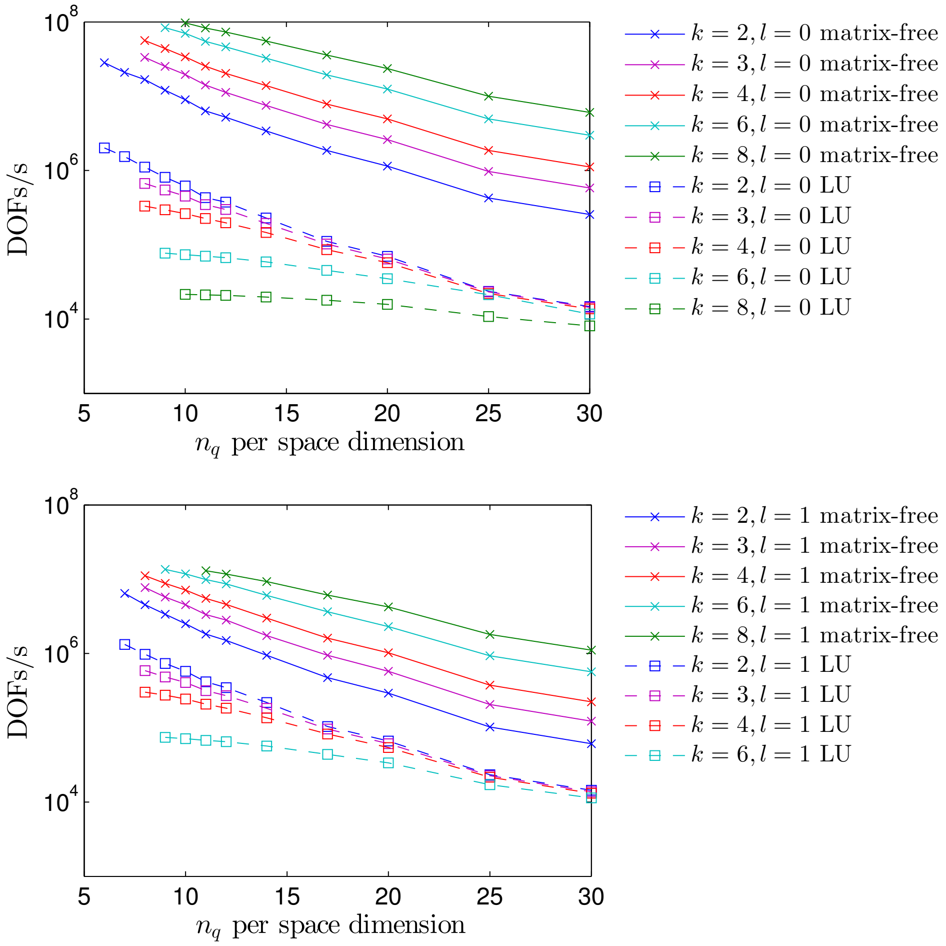


Figure 6.6.: Enriched inverse mass operator: performance of 'precompute' routines in DOFs/s over number of quadrature points per space dimension. Top: $l = 0$, bottom: $l = 1$.

The corresponding performance of the two inverse mass operators in applying the operator once is shown in Figure 6.7 considering the same settings. It is first noted that all curves are horizontal, meaning that the cost of applying both inverse mass operators is independent of the applied quadrature formula. This is an expected behavior, since the numerical integration of the enriched terms of the weak form is carried out in the setup step in both approaches. The performance of the LU-based method deteriorates for increasing polynomial degree k, a typical behavior of matrix-based methods, whereas the matrix-free approach yields almost no dependence on the polynomial degree, a typical behavior of matrix-free methods. Further, the matrix-free approach is faster than the matrix-based method by one to two orders of magnitude, but the advantage is somewhat reduced in the case of $l = 1$ compared to $l = 0$.

From this performance analysis we draw the conclusion that the presented matrix-free inverse mass operator via a Schur complement is much faster compared to a straightforward matrix-based approach. The inverse mass operator requires a setup step in each time step, in which the columns of the enrichment DOFs of a scalar mass matrix are precomputed. The cost of applying the inverse mass operator is independent of the number of quadrature points used for the

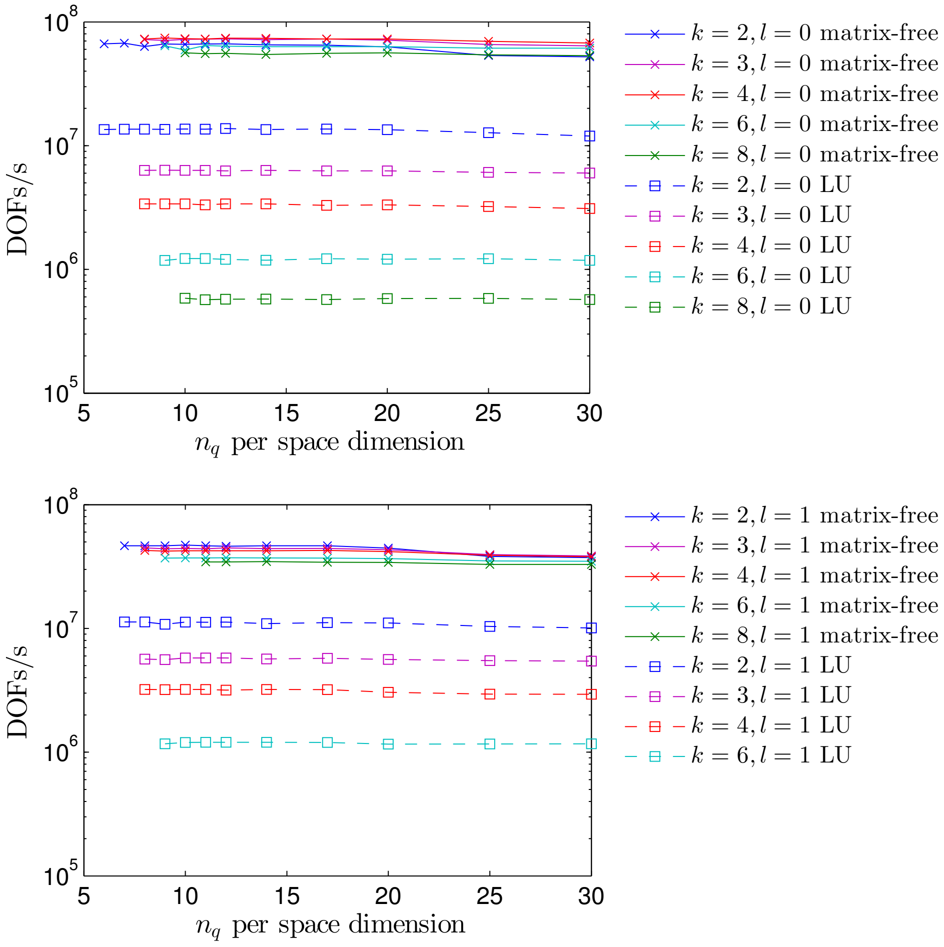


Figure 6.7.: Enriched inverse mass operator: performance of ‘apply’ routines in DOFs/s over number of quadrature points per space dimension. Top: $l = 0$, bottom: $l = 1$.

integration of the enriched weak form, at approximately twice the cost of a matrix-free inverse mass operator in a nonenriched cell.

6.3.3.3. Mixed Matrix-Free Projection Step

The projection step in Equation (3.40) according to *V3c* is also analyzed in detail regarding the efficiency of several solution strategies. The locality of the problem allows a variety of solution approaches. The equation may either be solved by a direct method, where the element matrices are evaluated and the equation system is solved via LU factorization. As an alternative, the projection step may be solved iteratively in a matrix-free manner with the procedure according to Section 3.5.2. We further consider a mixed approach, which applies different evaluation approaches for the cheap polynomial and expensive enrichment components.

In view of this discussion, we summarize the terms of the matrix formulation of the projection step in Equation (3.40) as $\boldsymbol{K} = \boldsymbol{M} + \tau_D \boldsymbol{D}$. This matrix is block-diagonal due to the absence of coupling terms between the elements and may further be split into four blocks, which correspond to the polynomial and enrichment DOFs analogous to the previous subsection, such that

Equation (3.40) becomes

$$\begin{bmatrix} \boldsymbol{K}^{\bar{V}\bar{U}} & \boldsymbol{K}^{\bar{V}\widetilde{U}} \\ \boldsymbol{K}^{\widetilde{V}\bar{U}} & \boldsymbol{K}^{\widetilde{V}\widetilde{U}} \end{bmatrix} \begin{bmatrix} \bar{\boldsymbol{U}} \\ \widetilde{\boldsymbol{U}} \end{bmatrix} = \begin{bmatrix} \bar{\boldsymbol{R}} \\ \widetilde{\boldsymbol{R}} \end{bmatrix}, \tag{6.18}$$

where the velocity vector corresponds to the second intermediate velocity and the vectors $\bar{\boldsymbol{R}}$ and $\widetilde{\boldsymbol{R}}$ to the right hand side.

Matrix-Based. The equation system may be solved in a straightforward manner by computing the element matrices and applying the action of its inverse via LU factorization. In this variant, all integrals are computed using the high quadrature rules required for the enrichment terms.

Standard Matrix-Free. The local system may be computed by the procedure presented in Section 3.5.2. Therein, an iterative conjugate gradient approach is considered, in which the vector-matrix products are computed in a matrix-free manner. It is noted that all terms have to be evaluated by the high quadrature rules required for the enrichment terms. In this variant, the matrix-free inverse mass operator via Schur complement according to the previous section is considered as a preconditioner.

Mixed Matrix-Free. The latter scheme may be modified as follows. The expensive matrix rows and columns corresponding to the enrichment are only computed once in each time step and stored prior to the iterative solver. Therein, it is sufficient to compute the matrix columns corresponding to the enrichment DOFs, $\widetilde{\boldsymbol{U}}$, due to the symmetry of the problem with $\boldsymbol{K}^{\bar{V}\widetilde{U}} = (\boldsymbol{K}^{\widetilde{V}\bar{U}})^T$. This means that 3 matrix columns are precomputed in the case of $l = 0$ and 24 matrix columns in the case of $l = 1$. These precomputed matrix blocks are used to compute the respective contributions to the matrix-vector product in each solver iteration. The remaining purely polynomial terms of the matrix-vector product are computed in a matrix-free manner in each solver iteration using $k + 1$ quadrature points. For this variant, the inverse mass operator via Schur complement is employed as a preconditioner as well.

Performance Evaluation. The performance of these three solution strategies is evaluated using the same numerical example as for the benchmarking of the inverse mass matrix routines presented in the previous section. We consider the whole projection step including the precomputation of the matrix terms in the mixed matrix-free variant, and the iterative solvers are converged to a relative accuracy of 10^{-6}. In Figure 6.8, the throughput in DOFs/s is plotted over the number of quadrature points used for the enrichment terms n_q separately for $l = 0$ and $l = 1$. Regarding $l = 0$, we note that the throughput increases with higher polynomial degree k for the two matrix-free variants, whereas the throughput decreases for the matrix-based variant. The mixed matrix-free variant is faster by approximately a factor of 3 to 5 than the matrix-free procedure, and that includes the precomputation of the matrix blocks containing enrichment shape functions. The matrix-based variant is in the range of one to two orders of magnitude slower compared to the matrix-free variants.

The picture is similar for the case $l = 1$, the advantage of the mixed matrix-free approach is much less significant, though. This is due to the relatively high cost of precomputing 24 matrix

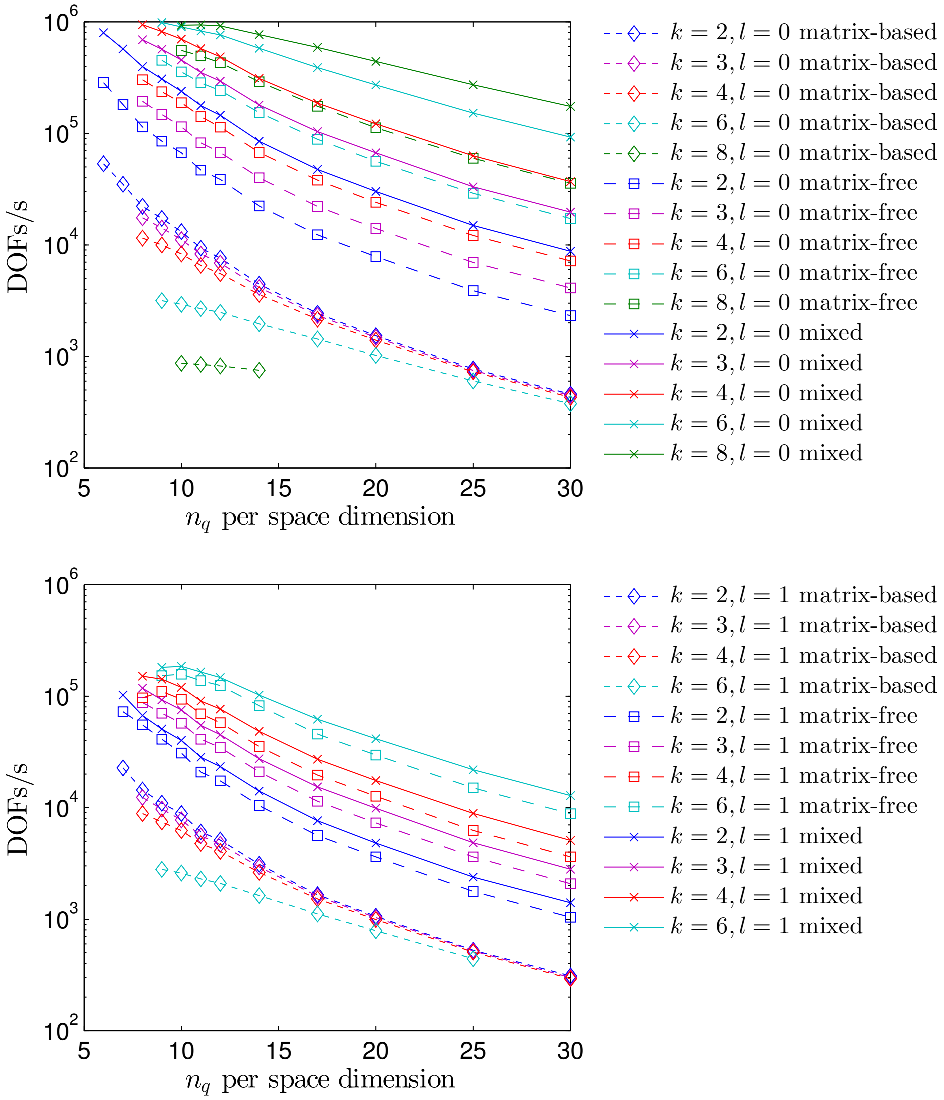


Figure 6.8.: Enriched projection solver: performance comparison of matrix-based, standard matrix-free, and mixed-matrix-free solution in DOFs/s over number of quadrature points per space dimension. Top: $l = 0$, bottom: $l = 1$.

columns. In this case, the graphs reveal another interesting aspect of the iterative solver: the iterative solution strategies yield curves that flatten in the range below $n_q = 12$. This fact is assumed to be due to inexact integration of the residual if few quadrature points are used. The number of required solver iterations of an iterative solver, which includes enrichment shape functions, can therefore be seen as an indicator of insufficient quadrature.

From this discussion, we conclude that the mixed matrix-free approach is the most efficient method for the solution of the local projection step. As a rule of thumb, it can be said that such a solver is promising in situations where the required number of solver iterations is larger than the number of enriched matrix columns. For example, in the viscous step, the number of solver iterations is usually in the range of 3 to 5 and 3 matrix columns would have to be precomputed for $l = 0$. These numbers do not make an application of a similar mixed matrix-free approach attractive in the viscous step (in addition to the nonsymmetry of the matrix in the NIPG formulation, which would require a larger number of precomputed terms). Finally, it is noted that the number of solver iterations in the viscous step lies in a similar range as in nonenriched simulations. From this fact we conclude that the conditioning of the equation system is in general not degraded through the enrichment shape functions, in particular in the case $l = 0$.

6.3.3.4. Implementation of Enriched Integrals in C++

The cell and face integrals of the Galerkin formulations in Sections 3.3.2 and 5.3.1 are evaluated using the computational kernels by Kronbichler and Kormann [156] within the deal.II finite element library [9] implemented in C++ according to Section 3.5. In this framework, the evaluation of the finite element interpolation in a quadrature point $\boldsymbol{u}_h(\boldsymbol{x}_q, t)$ as well as multiplication by test functions $\boldsymbol{v}_h(\boldsymbol{x}_q, t)$ and summation over quadrature points is provided by a class called `FEEvaluation`. The computational kernels include read (gather) and write (scatter) operations into global vectors, evaluation and integration routines based on sum factorization, as well as the combination of values and gradients on quadrature points.

For the evaluation of solutions from enriched function spaces, a modular extension to `FEEvaluation` has been developed. As this extension can be used generically and is not restricted to the current setting, it is detailed in the following. According to the definitions in Equations (6.1) and (6.5), the evaluation of the enriched function $\boldsymbol{u}_h(\boldsymbol{x}_q, t)$ combines the interpolation of a standard polynomial space $\bar{\boldsymbol{u}}_h$ of degree k with the interpolation $\widetilde{\boldsymbol{u}}_h$ of degree l. For the combined evaluation according to Equation (6.1), the two polynomial representations underlying $\bar{\boldsymbol{u}}_h$ and $\widetilde{\boldsymbol{u}}_h$ are each evaluated in the quadrature point location $\boldsymbol{x}_q$ with index q. The following C++ code shows the implementation of the function that computes the enriched interpolation in the q-th quadrature point, `get_value`, where the two components are combined and the enrichment function is multiplied. The second method described here concerns integration where the action on a quadrature point with index q is to *submit* a value prior to the multiplication by all the test functions in the quadrature point, `submit_value` [9, 156]. Note that the value to be tested is submitted to both the test function slot associated with the polynomial function space `function_space_1` and the slot of the enrichment polynomials `function_space_2` both receiving a contribution in the respective DOFs in the residual. After the loop over quadrature points, the actual multiplication by all basis functions and summation over all basis functions via sum factorization is done in a function called `integrate`. If gradients are evaluated as well in the same routine, as it is necessary in the Navier–Stokes equations, the quantities entering the `submit_value` and `submit_gradient` routines have to be temporally stored on each quadrature point, since the gradients generate contributions to the polynomial `submit_value` paths as well due to the product rule (see Appendix A.3). When the function `evaluate` is called, the temporally stored values are submitted to the `submit_value` and `submit_gradient` functions of the polynomial evaluators.

```
template <...> class EnrichedEvaluation
{
  typedef typename FEEvaluation<...>::value_type  value_type;
  typedef typename FEEvaluation<...>::scalar_type scalar_type;
```

```
  value_type get_value(const unsigned int q) const
  {
    return function_space_1.get_value(q) +
           enrichment_function[q] * function_space_2.get_value(q);
  }

  void submit_value (const value_type value_to_test,
                     const unsigned int q)
  {
    function_space_1.submit_value(value_to_test, q);
    function_space_2.submit_value(enrichment_function[q]
                                  *value_to_test, q);
  }
  ...
  FEEvaluation<...> function_space_1;
  FEEvaluation<...> function_space_2;
  scalar_type *enrichment_function;
};
```

The implementation of `FEEvaluation` uses templates on the space dimension, polynomial degree, the number of integration points, and number of components that are omitted for brevity. For the fluid velocity, the type `value_type` denotes a tensor with d components but the same code can be used for scalar enrichments when the `value_type` is a scalar. Furthermore, the particular implementation in `deal.II` combines the evaluation of several elements at once for making use of SIMD instructions (vectorization) in modern CPUs [156], which is why the inner quantity in a component of the tensor is not simply a `double` field but rather a short array of `double` variables.

In the evaluator, the wall function is a scalar quantity of type `scalar_type` that is accessed via a pointer `enrichment_function` to a table of the values on all enriched cells and all quadrature points. This factor can be precomputed prior to each time step using another `FEEvaluation` evaluator accessing the continuous finite element vectors of degree m for $\tau_{w,h}$ and y_h and resolving the formula for $\psi(\boldsymbol{x}_q)$. The latter quantity, for example Spalding's implicitly given wall function or van Driest's law, is evaluated numerically with the algorithms described in Appendices A.1 and A.2.

This enrichment evaluator `EnrichedEvaluation` is included in the existing Navier–Stokes code INDEXA (see Chapters 3 and 5) that also contains standard polynomial code paths where the additional operations due to the enrichment are undesirable. In order to avoid reimplementing all the weak forms of the Navier–Stokes equations with different evaluators, the generic programming capabilities of the C++ programming language are used via templates. To this end, a wrapper class `FEEvaluationWrapper` is introduced that contains a template argument to switch between a basic `FEEvaluation` object in standard simulations and `EnrichedEvaluation` in simulations including enriched elements. Both classes have the same interface, allowing for a seamless integration into the solver with a simple change in types and zero computational overhead in nonenriched simulations.

6.4. Summary

In this chapter, the high near-wall velocity gradient and the multitude of spatial and temporal scales present in turbulent boundary layers were identified as the two challenges in the simulation of turbulent wall-bounded flow. Wall modeling via function enrichment has been introduced as an approach which allows the computation of the sharp velocity gradient including the laminar sublayer with much coarser cells than can be used in standard numerical schemes. The basic idea is that a wall function is included in the function space of the numerical method as an additional shape function, besides the standard polynomial ones. As a result, the numerical method can inherently represent equilibrium boundary layers, but it is also sufficiently flexible to resolve nonequilibrium boundary layers, for example with high longitudinal pressure gradient. In general, the Galerkin method will find the optimal solution among the enrichment shape functions and the polynomial component in a least squares sense.

The formulation of the wall model and its efficient implementation have been presented. This wall modeling approach is applied in four different scenarios in the subsequent chapters, including the continuous FEM, the high-order DG method, RANS, DES, and WMLES.

7

Application I: Wall Modeling for LES in the Continuous FEM

The high computational cost of wall-resolved LES is the primary incentive for developing wall modeling via function enrichment, so the first application considers WMLES. As it was detailed in the previous chapter, the grid-resolution requirements of wall-resolved LES depend on the friction Reynolds number approximately as Re_τ^2 [12]. The goal of wall modeling is to reduce this dependency of the resolution requirements on wall units.

The methodology of wall modeling via function enrichment as introduced in the previous chapter is applied in the context of the continuous FEM. The continuous FEM is less frequently used in applications of high Reynolds number as compared to the DG method, and publications on wall modeling approaches are rare in this context; see the review article in [216] for an overview of available studies. One example is the weak imposition of no-slip boundary conditions [18, 95], in which the no-slip conditions are applied with similar terms as in the DG method presented in Chapter 3, but a certain amount of slip is allowed at the boundary. As a consequence, the gradient is modeled in the near-wall region, which enables the use of coarser meshes. A standard equilibrium wall function has for example been implemented in an FEM solver in [41].

With respect to subgrid scale modeling for LES, a variety of subgrid models is available within the continuous FEM [4]. The turbulent subgrid motions are in the present chapter modeled via a multifractal subgrid scale model embedded in a variational multiscale method. This model gives excellent results for wall-resolved LES [215] and has been successfully extended to passive-scalar mixing [214], low-Mach-number flow with variable density [213], and two-phase flow [212]. In the application to wall modeling via function enrichment, the idea is that the residual-based method provides numerical dissipation, which is both appropriate to stabilize the scheme and to model the unresolved energetic scales at the wall. As the work presented in the present chapter is historically the first application of high-gradient enrichments for the simulation of turbulent flow governed by the incompressible Navier–Stokes equations, it is the least developed compared to the subsequent chapters, in particular with respect to turbulence modeling. The insights gained in this study were yet essential in the further development of the wall model and triggered the development of a novel multiscale wall model, which will be presented in Chapter 10.

In this chapter, we begin with a short summary of the considered enrichment function space and introduce a minor modification of the adaptation algorithm described in Section 6.3.2.4, which is better consistent in the context of the continuous FEM. Since the numerical method used in this chapter is independent of the code presented in Part I of this thesis, several aspects are detailed, in particular the turbulence modeling approach (Section 7.2). Numerical results are discussed in Section 7.3. The work presented in this chapter was previously published in Krank and Wall [153, 154].

7.1. Enrichment Space

The present chapter considers the continuous FEM with eight-noded trilinearly interpolated hexahedral finite elements ($k = 1$). The enrichment space $\widetilde{\boldsymbol{u}}_h$ includes ramp functions according to Equation (6.6) in order to obtain a conforming discretization at the interface, where the enrichment ends, and is also based on linear finite elements ($l = 1$). As enrichment function, we employ Spalding's law according to Equation (6.7) with spatial derivatives according to Appendix A.3 and the numerical evaluation of the wall function according to Appendix A.1. The choice of $l = 1$ enables the method to weight the wall function linearly, which adds substantial flexibility to the method, in particular in recirculation zones. The enrichment function is modified by subtracting by its nodal values $\psi(\boldsymbol{x}_B, t)$ as in Equation (6.6) such that the enrichment vanishes on the nodes, which facilitates postprocessing and the application of boundary conditions. The decomposition of the proposed function space into a linear standard and enrichment component is visualized in Figure 6.4. Only a few cell layers at the wall are enriched, in the present chapter between 2 and 4, and the width of $\widetilde{\Omega}_h$ is user-specified. In Chapter 8, an adaptation scheme is introduced, which evaluates in each time step, which of the cells should be enriched.

Any other Lagrangian finite element could be employed as well, including higher-order elements and unstructured grids. For example, unstructured tetrahedral meshes were used in a student thesis carried out as part of this research [111], in which linearly interpolated four-noded tetrahedral elements were used both for the standard polynomial and enrichment space; the application of function enrichment in that context is possible without modification of the method. It is noted that the doubling of the DOFs per element (with $k = 1$ and $l = 1$) gives the numerical method a turbulence-resolving power well beyond linear elements despite its formal accuracy of second order. This fact may contribute to the success of the present method and the ability of using comparably coarse meshes for the boundary layer flows presented in Section 7.3. The pressure is discretized using the standard polynomial space with $k = 1$.

The wall shear stress is adapted spatially and temporally in order to take into account local fluctuations and their temporal evolution. To this end, the adaptation algorithm presented in Section 6.3.2.4 is modified in the present chapter, to increase the consistency in the context of the continuous FEM and to take into account the special case of $k = l = 1$ as follows.

In the FEM, the wall shear stress can be computed with Equation (6.11) via the velocity gradient. An alternative method is to calculate the wall shear stress in terms of a nodal wall-parallel force vector $\boldsymbol{r}^{\boldsymbol{v}}_{\|B}$ on the strongly enforced Dirichlet boundary $\partial\Omega^D$ divided by the nodally defined local area A_B. The result is interpolated according to Equation (6.12) with a standard linear FEM expansion of degree $m = 1$,

$$\tau_{w,h} = \sum_{B \in N^{c,m}} N_B^{c,m} \frac{\|\boldsymbol{r}^{\boldsymbol{v}}_{\|B}\|}{A_B}, \tag{7.1}$$

with the norm $\|\cdot\|$ of the three components corresponding to the space dimensions. The force vector equals the right-hand-side residual vector of the final matrix system as discussed in Section 7.2.5. The nodal area is given as the integral of the polynomial partition of unity on the boundary:

$$A_B = \int_{\partial\Omega^D} N_B^{c,m} \mathrm{d}A. \tag{7.2}$$

Both the wall shear stress via velocity gradient and the present force-based method represent an accurate definition of the instantaneous traction and are equivalent for the continuous case, but differences arise on discrete level. One of the differences is that the latter force-based method in Equation (7.1) requires the residual to be converged to give an accurate prediction. Therefore, the gradient-based method is applied in the first five time steps of the transient simulation as a converged residual is not available in the first time step and the gradient-based method is more robust if the initial field is not divergence-free.

Another aspect is the spatial coarsening introduced in the wall shear stress by taking $m = 1$ and $k > 1$ (see Section 6.3.2.4), which does not apply in the present chapter due to $k = 1$. This coarsening is physically motivated since Spalding's law is a relation for mean quantities, i.e., the mean velocity is related to the average wall shear stress. The wall shear stress would be statistically overpredicted without this coarsening, since the nodal traction vectors do in general not point in streamwise direction. It is suggested to calculate the stress via a locally averaged force field with a characteristic length scale αh instead of h. Such a local averaging operation allows for spatial variations of the traction and yet local fluctuations are smoothed. This averaging is realized via level-transfer operators from plain aggregation algebraic multigrid methods for separating scales, similar to the method used by [102] to explicitly separate the velocity scales in LES. A discrete wall shear stress $\tau_{w,\alpha h}$ with a coarser characteristic element length αh as a multiple of the element length h is obtained.

For this method a prolongation matrix $\boldsymbol{P}_{\alpha h}^{h}$ is generated and the restriction matrix is defined as the transpose of the prolongation matrix resulting in $\boldsymbol{R}_h^{\alpha h} = (\boldsymbol{P}_{\alpha h}^{h})^T$ and implying $\boldsymbol{R}_h^{\alpha h}\boldsymbol{P}_{\alpha h}^{h} = \boldsymbol{I}$ with the identity matrix $\boldsymbol{I}$. A scale-separation or aggregation operator is defined as

$$\boldsymbol{S}_h^{\alpha h} = \boldsymbol{P}_{\alpha h}^{h}\boldsymbol{R}_h^{\alpha h} \tag{7.3}$$

yielding a coarse-scale force field via a vector-matrix multiplication according to

$$\boldsymbol{r}_{\alpha h}^{\boldsymbol{v}} = \boldsymbol{S}_h^{\alpha h}\boldsymbol{r}_h^{\boldsymbol{v}}. \tag{7.4}$$

This result is applied to calculate the shear stress $\tau_{w,\alpha h}$ in Equation (7.1) with the usual value given by the algebraic multigrid algorithms of $\alpha = 3$. Figure 7.1 compares $\tau_{w,h}$ and $\tau_{w,3h}$ showing that the field variable is averaged locally but still may take larger variations into account. Thus, $\tau_{w,3h}$ is an appropriate representation of the wall-shear stress for the spatial adaptation of Spalding's law.

Further, special treatment in the case of $\tau_{w,3h}$ approaching zero, such as at reattachment points, is required, as the function space would become linearly dependent in the limit of $\tau_{w,3h} \to 0$. At such locations, we prescribe a minimum wall shear stress of 1% of the nominal value for computation of Spalding's law, which has shown to be sufficient to circumvent this issue without loss of accuracy. This clipping is replaced in the subsequent chapters by an adaptation algorithm, in which the enrichment would automatically be switched off in such locations. Finally, the fact that the mass matrix is a global equation system (see Section 2.2.2.3) modifies the L^2 projection in Equation (6.14) to yield a global equation system. However, the special construction of the

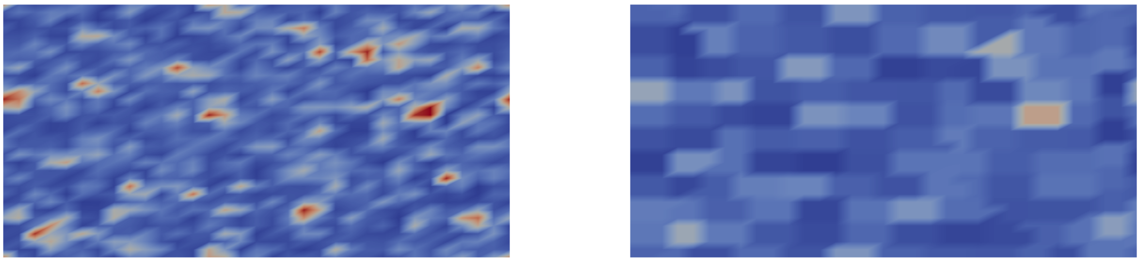


Figure 7.1.: Comparison of the wall shear stress (left) with the aggregated wall shear stress (right) of turbulent channel flow on a mesh with 32^3 elements. Red indicates high and blue low values.

enrichment suggests to solely project the enrichment component of the solution according to

$$\left(\widetilde{\boldsymbol{v}}_h^{n,\mathrm{new}}, \widetilde{\boldsymbol{u}}_h^{n,\mathrm{new}}\right)_\Omega = \left(\widetilde{\boldsymbol{v}}_h^{n,\mathrm{new}}, \widetilde{\boldsymbol{u}}_h^{n,\mathrm{old}}\right)_\Omega . \tag{7.5}$$

This modification guarantees that the solution on the nodes remains unchanged in the projection. Complementary vectors required by the discrete time integration procedure such as a potential acceleration vector are projected with the same matrix system.

7.2. Subgrid-Scale Modeling

Wall modeling via function enrichment is designed to use coarse meshes in the near-wall region, implying that not even the largest turbulent motions are resolved at the wall. In the bulk flow, the unresolved turbulent scales on subgrid level are derived via a scale separation by variational projection as suggested originally for LES by Hughes et al. [120] instead of the more widely used filter-based one [175]. The scale separation gives rise to unresolved scales that we model by a structural reconstruction via a multifractal subgrid scale embedded in a residual-based variational multiscale method as outlined in the following. At the wall, the residual-based approach shows to be sufficiently general to act as a statistical turbulence model. In Chapters 9 and 10, the near-wall turbulence is instead modeled by a classical RANS model. In this section, we begin with a derivation of the weak form and subsequently derive the multifractal subgrid terms as well as the residual-based method.

7.2.1. Weighted Residual Formulation

The incompressible Navier–Stokes equations are considered in this work as given in Section 2.1.1 with the momentum equation (2.1) and the continuity equation (2.2) including the convective formulation of the convective term (Equation (2.3)), as well as the corresponding initial and boundary conditions. A weighted-residual formulation is obtained with a standard procedure by multiplying the momentum equation (2.1) with a weighting function $\boldsymbol{v} \in \mathcal{V}^{\boldsymbol{u}}$ and the continuity equation (2.2) with $q \in \mathcal{V}^p$, where $\boldsymbol{u} \in \mathcal{V}^{\boldsymbol{u}}$ and $p \in \mathcal{V}^p$. Appropriate continuous

FEM spaces for $\mathcal{V}^{\boldsymbol{u}}$ and $\mathcal{V}^{p}$ based on a conforming FE discretization are assumed, and the discrete velocity space may include the high-gradient enrichment according to Equation (6.6). In contrast to the DG formulation derived in Section 3.3.2, the equations are not only integrated over one cell but over the whole domain Ω, the pressure and viscous terms are integrated by parts and the Neumann boundary conditions (Equation (2.6)) are applied to the arising boundary integrals. The resulting variational formulation reads

$$\mathcal{B}_{\text{NS}}(\boldsymbol{v}, q; \boldsymbol{u}, p) = \ell(\boldsymbol{v}) \tag{7.6}$$

with the left hand side of the momentum equation and the contribution of the continuity equation

$$\mathcal{B}_{\text{NS}}(\boldsymbol{v}, q; \boldsymbol{u}, p) = \left(\boldsymbol{v}, \frac{\partial \boldsymbol{u}}{\partial t}\right)_{\Omega} + (\boldsymbol{v}, \boldsymbol{u} \cdot \nabla \boldsymbol{u})_{\Omega} - (\nabla \cdot \boldsymbol{v}, p)_{\Omega} + (\boldsymbol{\epsilon}(\boldsymbol{v}), 2\nu\boldsymbol{\epsilon}(\boldsymbol{u}))_{\Omega} + (q, \nabla \cdot \boldsymbol{u})_{\Omega} \tag{7.7}$$

and the right hand side of the momentum equation

$$\ell(\boldsymbol{v}) = (\boldsymbol{v}, \boldsymbol{f})_{\Omega} + (\boldsymbol{v}, \boldsymbol{h})_{\partial\Omega^N}. \tag{7.8}$$

The L^2-inner product is defined as in Section 3.3.1, and $(\cdot, \cdot)_{\partial\Omega^N}$ defines an integral over the Neumann boundary $\partial\Omega^N$.

7.2.2. Scale Separation for Large-Eddy Simulation

In addition to the decomposition of the solution space into standard and enriched components according to Equation (6.1), the velocity space is further separated into resolved $\boldsymbol{u}_h$ and unresolved scales $\boldsymbol{u}'$, reading

$$\boldsymbol{u} = \boldsymbol{u}_h + \boldsymbol{u}' = \bar{\boldsymbol{u}}_h + \widetilde{\boldsymbol{u}}_h + \boldsymbol{u}'. \tag{7.9}$$

The equivalent separation of scales of the pressure into a resolved p_h and unresolved component p' is also performed, resulting in

$$p = p_h + p'. \tag{7.10}$$

The direct sum decomposition of the underlying spaces into resolved and unresolved ones is assumed as $\mathcal{V}^{\boldsymbol{u}} = \mathcal{V}_h^{\boldsymbol{u}} \oplus \mathcal{V}^{\boldsymbol{u}'} = \mathcal{V}_h^{\bar{\boldsymbol{u}}} \oplus \mathcal{V}_h^{\widetilde{\boldsymbol{u}}} \oplus \mathcal{V}^{\boldsymbol{u}'}$ and $\mathcal{V}^{p} = \mathcal{V}_h^{p} \oplus \mathcal{V}^{p'}$. In Chapter 10, a similar composition of the solution of resolved and unresolved scales is developed, which increases the consistency in the context of wall-modeling via function enrichment. Inserting Equations (7.9) and (7.10) into the weighted residual formulation (7.6) gives rise to the following relation:

$$\mathcal{B}_{\text{NS}}(\boldsymbol{v}, q; \boldsymbol{u}_h, p_h) + \mathcal{B}_{\text{NS}}^{\text{lin}}(\boldsymbol{v}, q; \boldsymbol{u}', p') + \mathcal{C}(\boldsymbol{v}; \boldsymbol{u}_h, \boldsymbol{u}') + \mathcal{R}(\boldsymbol{v}; \boldsymbol{u}') = \ell(\boldsymbol{v}). \tag{7.11}$$

The term $\mathcal{B}_{\text{NS}}(\boldsymbol{v}, q; \boldsymbol{u}_h, p_h)$ constitutes the part of the formulation that is represented by the resolved solution space. The contribution $\mathcal{B}_{\text{NS}}^{\text{lin}}(\boldsymbol{v}, q; \boldsymbol{u}', p')$ summarizes the linear terms dependent on the subgrid scales $\boldsymbol{u}'$ and p':

$$\mathcal{B}_{\text{NS}}^{\text{lin}}(\boldsymbol{v}, q; \boldsymbol{u}', p') = \left(\boldsymbol{v}, \frac{\partial \boldsymbol{u}'}{\partial t}\right)_{\Omega} - (\nabla \cdot \boldsymbol{v}, p')_{\Omega} + (\boldsymbol{\epsilon}(\boldsymbol{v}), 2\nu\boldsymbol{\epsilon}(\boldsymbol{u}'))_{\Omega} + (q, \nabla \cdot \boldsymbol{u}')_{\Omega}. \tag{7.12}$$

The nonlinear convective term gives rise to the cross- and Reynolds stresses as

$$\mathcal{C}(\boldsymbol{v};\boldsymbol{u}_h,\boldsymbol{u}') = (\boldsymbol{v}, \boldsymbol{u}_h \cdot \nabla \boldsymbol{u}' + \boldsymbol{u}' \cdot \nabla \boldsymbol{u}_h)_\Omega \tag{7.13}$$

and

$$\mathcal{R}(\boldsymbol{v};\boldsymbol{u}') = (\boldsymbol{v}, \boldsymbol{u}' \cdot \nabla \boldsymbol{u}')_\Omega\,. \tag{7.14}$$

A basic characteristic of the variational multiscale method is that the solution and weighting function spaces have the same structure, i.e., the spaces for the weighting function may likewise be decomposed into the corresponding resolved and unresolved contributions:

$$\boldsymbol{v} = \boldsymbol{v}_h + \boldsymbol{v}' = \bar{\boldsymbol{v}}_h + \widetilde{\boldsymbol{v}}_h + \boldsymbol{v}', \tag{7.15}$$

$$q = q_h + q'. \tag{7.16}$$

Since Equation (7.11) is linear with respect to the weighting functions, it may be separated into an equation for the resolved scales and an equation for the unresolved scales. According to the variational multiscale methodology, only the equation for the resolved scales is taken into further consideration:

$$\mathcal{B}_{\text{NS}}(\boldsymbol{v}_h, q_h; \boldsymbol{u}_h, p_h) + \mathcal{B}_{\text{NS}}^{\text{lin}}(\boldsymbol{v}_h, q_h; \boldsymbol{u}', p') + \mathcal{C}(\boldsymbol{v}_h; \boldsymbol{u}_h, \boldsymbol{u}') + \mathcal{R}(\boldsymbol{v}_h; \boldsymbol{u}') = \ell(\boldsymbol{v}_h). \tag{7.17}$$

This result still contains the unresolved quantities $\boldsymbol{u}'$ and p' which are unknown and have to be modeled. In the following sections, this is done via multifractal scale similarity and residual-based modeling.

Remark: The enrichment approach may also be interpreted as a separation of the solution vector in three scale groups as for example described by Gravemeier et al. [104] and indicated in Equation (7.9). The three scales are represented by the standard resolved scale $\bar{\boldsymbol{u}}_h$, the enrichment scale $\widetilde{\boldsymbol{u}}_h$, as well as the unresolved scale $\boldsymbol{u}'$. With regard to LES, the standard scale resolves large eddies that are at least of the size of the characteristic element length. The enrichment scale, however, represents flow features in a statistical sense and without resolving large eddies in the near-wall region explicitly. The physical interpretation of the unresolved scales are fluctuations on the subgrid level. This interpretation and the framework of the variational multiscale method allows for tailored turbulence models for each scale range: A RANS turbulence model for the statistical part of the solution and an LES model for the eddy-resolving part of the solution. Such a turbulence modeling approach is elaborated in Chapter 10. This turbulence model could be considered in the present continuous FEM framework as well.

7.2.3. Subgrid Modeling with Multifractal Subgrid Scales

The cross and Reynolds stress terms (Equation (7.13)) and (Equation (7.14)) are modeled explicitly by reconstruction of the unresolved scale $\boldsymbol{u}'$ via a multifractal subgrid-scale model as proposed by Rasthofer and Gravemeier [215]. The multifractal subgrid scale model follows the idea that turbulence originates from repeated stretching and folding of vortical structures and that this process is scale-invariant. The model attempts to reconstruct the subgrid-scale vorticity and computes the subgrid velocity through the law of Biot-Savart, indicating that the large eddies of the flow have to be resolved explicitly. For a detailed derivation of the governing relations it is referred to Burton and Dahm [38]. The subsequent description summarizes the model as it was used by Rasthofer and Gravemeier [215].

The subgrid velocity scales with the small-resolved velocity field $\delta\bar{\boldsymbol{u}}_h$ and a proportionality factor B as

$$\boldsymbol{u}' \approx B\delta\bar{\boldsymbol{u}}_h. \tag{7.18}$$

The small-scale velocity is determined by an explicit filtering of the standard FE component of the resolved velocity, yielding

$$\boldsymbol{u} = \bar{\boldsymbol{u}}_{\alpha h} + \delta\bar{\boldsymbol{u}}_h + \widetilde{\boldsymbol{u}}_h + \boldsymbol{u}' \tag{7.19}$$

for the overall composition of the velocity space. The large-scale velocity field $\bar{\boldsymbol{u}}_{\alpha h}$ is identified by a length scale of αh as a multiple of the element length. This decomposition is chosen due to the physical interpretation of the standard finite element space as eddies whereas the enrichment space represents a statistical velocity profile that does not resolve eddies by nature.

The explicit scale separation of $\bar{\boldsymbol{u}}_{\alpha h}$ and $\delta\bar{\boldsymbol{u}}_h$ is performed analogous to [215] via level-transfer operators from a plain aggregation algebraic multigrid method as proposed in [102] and applies similar relations as used for smoothing of the wall shear stress in Section 6.3.2.4. The standard parameter of $\alpha = 3$ is applied.

The proportionality factor in Equation (7.18) is given as

$$B = C_{\text{sgs}} \left(1 - \alpha^{-4/3}\right)^{-1/2} 2^{-2N/3} \left(2^{4N/3} - 1\right)^{1/2}. \tag{7.20}$$

In the current application of convection-dominated high-Reynolds-number flow, the constant $C_{\text{sgs}} = 0.15$ is chosen. This value is significantly lower than the one suggested in [215], which compensates for the fact that the near-wall limit as suggested in the original publication is not considered herein. B is evaluated at the quadrature points during the evaluation of the discrete formulation (Equation (7.17)). The number of cascade steps N from the smallest resolved scales of size h to the viscous scale λ_ν is approximated via the local element Reynolds number $\text{Re}_h = \frac{\|\boldsymbol{u}_h\| h}{\nu}$ and a proportionality constant c_ν resulting in

$$N = \log_2\left(\frac{h}{\lambda_\nu}\right) = \log_2\left(c_\nu \text{Re}_h^{3/4}\right). \tag{7.21}$$

A value for the proportionality constant $c_\nu = 0.1$ is used, which is close to the value of $\frac{1}{12.3}$ determined experimentally by Mullin and Dahm [192], and h is approximated by the cube root of the local element volume.

The final result for the modeled cross and Reynolds stress terms (Equation (7.13) and (7.14)) with the presented relation for the subgrid-scale velocity (Equation (7.18)) is

$$\mathcal{C}(\boldsymbol{v}_h; \boldsymbol{u}_h, \boldsymbol{u}') \approx (\boldsymbol{v}_h, B(\boldsymbol{u}_h \cdot \nabla\delta\bar{\boldsymbol{u}}_h + \delta\bar{\boldsymbol{u}}_h \cdot \nabla\boldsymbol{u}_h))_\Omega \tag{7.22}$$

and

$$\mathcal{R}(\boldsymbol{v}_h; \boldsymbol{u}') \approx \left(\boldsymbol{v}_h, B^2(\delta\bar{\boldsymbol{u}}_h \cdot \nabla\delta\bar{\boldsymbol{u}}_h)\right)_\Omega. \tag{7.23}$$

7.2.4. Residual-Based Modeling

The multifractal subgrid scale model as presented in the previous section enables the reconstruction of the subgrid velocity field in order to model the cross and Reynolds stress terms. As reported in [39], the model allows both for dissipation and backscatter of energy, which may result in destabilizing effects. Therefore, it was embedded in the residual-based variational multiscale method providing a stable numerical method in [215], and this approach is chosen in the current work as well.

To the left hand side of the scale separation in Equation (7.17), the following additional terms are added:

$$\underbrace{\left(\boldsymbol{u}_h \nabla \cdot \boldsymbol{v}_h, \tau_M \boldsymbol{R}_{M,h}\right)_\Omega}_{\text{SUPG}} + \underbrace{\left(\nabla \cdot \boldsymbol{v}_h, \tau_D \boldsymbol{R}_{C,h}\right)_\Omega}_{\text{grad-div}} + \underbrace{\left(\nabla q_h, \tau_M \boldsymbol{R}^{M,h}\right)_\Omega}_{\text{PSPG}}, \tag{7.24}$$

which are partly taken into account by the term $\mathcal{B}^{\text{lin}}_{\text{NS}}(\boldsymbol{v}_h, q_h; \boldsymbol{u}', p')$. The included terms consist in an SUPG, a grad-div and a Pressure Stabilizing Petrov–Galerkin (PSPG) term. The SUPG term stabilizes the method with respect to convection by introducing a certain amount of artificial dissipation [34], see also Section 2.2.2.3. A better fulfillment of the divergence-free constraint (Equation (2.2)) and improved convergence of the iterative solver is obtained via the grad-div term [197] which also introduces a certain amount of dissipation in the system. This term is analogous to the div-div term presented in Chapter 3, which penalizes the divergence error with the same idea. The PSPG contribution enables circumventing the inf-sup condition (see e.g. [82]) and allows velocity and pressure spaces of equal order [250].

The momentum residual $\boldsymbol{R}_{M,h}$ is defined as

$$\boldsymbol{R}_{M,h} = \frac{\partial \boldsymbol{u}_h}{\partial t} + \boldsymbol{u}_h \cdot \nabla \boldsymbol{u}_h + \nabla p_h - 2\nu \nabla \cdot \boldsymbol{\epsilon}(\boldsymbol{u}_h) - \boldsymbol{f}_h + B(\boldsymbol{u}_h \cdot \nabla \delta \bar{\boldsymbol{u}}_h + \delta \bar{\boldsymbol{u}}_h \cdot \nabla \boldsymbol{u}_h) + B^2(\delta \bar{\boldsymbol{u}}_h \cdot \nabla \delta \bar{\boldsymbol{u}}_h). \tag{7.25}$$

In contrast to [215], it is suggested to include the modeled cross and Reynolds stress terms in Equations (7.13) and (7.14) in the residual for better consistency. The discrete continuity residual $\boldsymbol{R}_{C,h}$ is

$$\boldsymbol{R}_{C,h} = \nabla \cdot \boldsymbol{u}_h. \tag{7.26}$$

The stabilization parameters τ_M and τ_D are designed to take into account the nonpolynomial character of the element space. A definition inspired, among others, by Codina [50] and Gravemeier et al. [101] is chosen including a transient, convective and viscous contribution for τ_M as

$$\tau_M = \frac{1}{\frac{1}{\Delta t} + 2\sqrt{\frac{\lambda_h}{3}}\|\boldsymbol{u}_h\|_2 + 4\lambda_h \nu} \tag{7.27}$$

with the time step Δt and a reciprocal scaling of τ_M and τ_D, yielding

$$\tau_D = \frac{1}{4\lambda_h \tau_M}. \tag{7.28}$$

It is noted that $\left(2\sqrt{\lambda_h/3}\right)^2 \leq 4\lambda_h$ which has been reported to be a requirement for example in [50].

The parameter λ_h generally incorporates the characteristics of the element, for example the polynomial order of the underlying function space. For standard Lagrangian shape functions with well-defined polynomial order, such as the nonenriched elements, values are for instance for the polynomial orders $p = \{1, 2, 3\}$ given as $\lambda_h = \{\frac{3}{h^2}, \frac{12}{h^2}, \frac{60}{h^2}\}$ with the characteristic element length h [84, 108]. In the current application, element spaces of the enriched elements are nonpolynomial making it impossible to specify appropriate values *a priori*. Therefore, an element-specific value for λ_h is computed in a consistent way via inverse estimate as suggested by Harari and Hughes [108] ensuring stability for convection-dominated flows by solving the local generalized eigenvalue problem for its maximum eigenvalue λ_h and v_h given as

$$(\Delta w_h, \Delta v_h)_{\Omega_e} - \lambda_h (\nabla w_h, \nabla v_h)_{\Omega_e} = 0. \tag{7.29}$$

Ω_e represents the element domain and the solution v_h and weighting function space w_h are defined similarly as the enriched velocity space in Equation (6.1), e.g., for v_h:

$$v_h(\boldsymbol{x}, t) = \bar{v}_h(\boldsymbol{x}, t) + \widetilde{v}_h(\boldsymbol{x}, t) \tag{7.30}$$

The standard and enrichment components are given with a single DOF per node as

$$\bar{v}_h(\boldsymbol{x}, t) = \sum_{B \in N^{\boldsymbol{u}}} N_B^{\boldsymbol{u}}(\boldsymbol{x}) \bar{v}_B(t) \tag{7.31}$$

and

$$\widetilde{v}_h(\boldsymbol{x}, t) = \sum_{B \in N^{\boldsymbol{u}}_{enr}} N_B^{\boldsymbol{u}}(\boldsymbol{x})(\psi(\boldsymbol{x}, t) - \psi(\boldsymbol{x}_B, t)) r_h(\boldsymbol{x}) \widetilde{v}_B(t). \tag{7.32}$$

In Equation (7.29), we only consider the diagonal part of Δw_h and Δv_h such that the transition to the values of λ_h used in standard elements, which are derived based on the 1D assumption, becomes smoother.

Evaluating the inverse estimate with one DOF per node results in matrix dimensions of only 16×16 for the elements presently considered, such that the eigenvalue-related computation time is negligible. A favorable characteristic of the presented stabilization parameter is highlighted: τ_M and τ_D are completely free of the element length if λ_h is computed via Equation (7.29). Especially for anisotropic elements, the definition of h is not obvious and many definitions have been proposed. The advantages of such a definition are also discussed for example by Franca and Madureira [85].

Solely for linear elements, the standard value of $\lambda_h = 3/h^2$ is applied with the volume-equivalent diameter $h = (6V(\Omega_e)/\pi)^{1/3}/\sqrt{d}$ with the element volume $V(\Omega_e)$ and the number of space dimensions d for simplicity [119]. Due to large variations of the stabilization parameters τ_M and τ_D within one element, especially in the first element at the no-slip boundary condition, the parameters are evaluated on the quadrature points.

The present approach for turbulence modeling has been presented as a subgrid-scale model for LES assuming that the largest eddies are resolved by the scheme. However, inside the first element at the wall, with the first off-wall node placed at $y^+ > 100$, this is certainly not fulfilled in the wall-region. The presented turbulence model has yet proven to be able to model the necessary subgrid scales even if the largest eddies are not resolved everywhere.

7.2.5. Final Discrete Problem

Since the present continuous FEM solver is independent of the numerical methods developed in Chapters 3 and 5, further details on the solution procedure are given. The final semi-discretized problem becomes

$$\begin{aligned}
&\left(\boldsymbol{v}_h, \frac{\partial \boldsymbol{u}_h}{\partial t}\right)_\Omega + (\boldsymbol{v}_h, \boldsymbol{u}_h \cdot \nabla \boldsymbol{u}_h)_\Omega - (\nabla \cdot \boldsymbol{v}_h, p_h)_\Omega + (\boldsymbol{\epsilon}(\boldsymbol{v}_h), 2\nu\boldsymbol{\epsilon}(\boldsymbol{u}_h))_\Omega \\
&+ \underbrace{(\boldsymbol{v}_h, B(\boldsymbol{u}_h \cdot \nabla \delta\bar{\boldsymbol{u}}_h + \delta\bar{\boldsymbol{u}}_h \cdot \nabla \boldsymbol{u}_h))_\Omega}_{\mathcal{C}} + \underbrace{\left(\boldsymbol{v}_h, B^2(\delta\bar{\boldsymbol{u}}_h \cdot \nabla \delta\bar{\boldsymbol{u}}_h)\right)_\Omega}_{\mathcal{R}} \\
&+ \underbrace{(\boldsymbol{u}_h \nabla \cdot \boldsymbol{v}_h, \tau_M \boldsymbol{R}_{M,h})_\Omega}_{\text{SUPG}} + \underbrace{(\nabla \cdot \boldsymbol{v}_h, \tau_D \boldsymbol{R}_{C,h})_\Omega}_{\text{grad-div}} \\
&+ (q_h, \nabla \cdot \boldsymbol{u}_h)_\Omega + \underbrace{(\nabla q_h, \tau_M \boldsymbol{R}_{M,h})_\Omega}_{\text{PSPG}} \\
&= (\boldsymbol{v}_h, \boldsymbol{f}_h)_\Omega + (\boldsymbol{v}_h, \boldsymbol{h}_h)_{\partial\Omega^N}
\end{aligned} \tag{7.33}$$

where the contributions of the multifractal and residual-based subgrid-scales are labeled. The residuals $\boldsymbol{R}_{M,h}$ and $\boldsymbol{R}_{C,h}$ are defined in Equations (7.25) and (7.26), the stabilization parameters τ_M and τ_D are given in Equations (7.27) and (7.28) and the coefficient B in Equation (7.20). The terms are integrated in space applying direction-dependent Gaussian quadrature rules of appropriate order, as it is detailed in Section 6.3.3.1, that enable an accurate integration despite the nonpolynomial function space. Equation (7.33) is integrated in time by means of an implicit second-order accurate generalized-α time integration scheme with $\rho_\infty = 0.5$ [103, 126]. Adaptive time stepping is employed such that the maximum Courant number is kept constant at $\mathrm{Cr} = 0.5$ for all simulations presented.

The final matrix system is linearized and iteratively solved via a Picard-iteration scheme, yielding

$$\boldsymbol{K}_i^{n+1} \Delta \boldsymbol{z}_{i+1}^{n+1} = -\boldsymbol{r}_i^{n+1} \tag{7.34}$$

for the current time step $n + 1$ and nonlinear iteration $i + 1$, omitting the subscript h for simplicity. The increment includes both velocity and pressure increments from the current nonlinear iteration such that

$$\Delta \boldsymbol{z}_{i+1} = \begin{bmatrix} \Delta \boldsymbol{U}_{i+1} \\ \Delta \boldsymbol{P}_{i+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{U}_{i+1} - \boldsymbol{U}_i \\ \boldsymbol{P}_{i+1} - \boldsymbol{P}_i \end{bmatrix}. \tag{7.35}$$

The matrix $\boldsymbol{K}$ contains the linearization of all contributions of Equation (7.33) except $\mathcal{C}$, $\mathcal{R}$ and the respective stabilization terms, which are treated in a fixed-point-like procedure [215]. The residual $\boldsymbol{r}$ summarizes all terms of Equation (7.33) at the previous nonlinear iteration i. In Equation (7.36), $\boldsymbol{K}$ is split into four parts including $\boldsymbol{K}^{vu}$, $\boldsymbol{K}^{vp}$, $\boldsymbol{K}^{qu}$, and $\boldsymbol{K}^{qp}$ and $\boldsymbol{r}$ is split into two vectors $\boldsymbol{r}^v$ and $\boldsymbol{r}^q$:

$$\boldsymbol{K} = \begin{bmatrix} \boldsymbol{K}^{vu} & \boldsymbol{K}^{vp} \\ \boldsymbol{K}^{qu} & \boldsymbol{K}^{qp} \end{bmatrix} \qquad \boldsymbol{r} = \begin{bmatrix} \boldsymbol{r}^v \\ \boldsymbol{r}^q \end{bmatrix} \tag{7.36}$$

$\boldsymbol{K}^{vu}$ contains the transient, convective, and viscous term as well as terms of SUPG and grad-div. $\boldsymbol{K}^{vp}$ comprises the pressure term and the respective part of SUPG. $\boldsymbol{K}^{qu}$ and $\boldsymbol{K}^{qp}$ summarize the continuity contribution and the PSPG terms. The nodal values of the momentum-residual vector $\boldsymbol{r}^v$ on the Dirichlet boundary are equivalent to the nodal forces and are used to calculate the wall-shear stress $\tau_{w,\alpha h}$ in Equation (7.1).

Table 7.1.: WMLES: Channel flow cases and resolutions.

Case	$N_{x1}\times N_{x2}\times N_{x3}$	Re_τ	y_1^+	N_{wm}
$Ch8wm2$	8×8×8	590; 950; 2,000	147.5; 237.5; 500	2
$Ch12wm2$	12×12×12	590; 950; 2,000	98.3; 158.3; 333.3	2
$Ch16wm3$	16×16×16	590; 950	73.8; 118.8	3
$Ch16wm2$	16×16×16	2,000	250	2
$Ch24wm3$	24×24×24	950	79.2	3
$Ch24wm2$	24×24×24	5,000	416.7	2
$Ch32wm2$	32×32×32	5,000	312.5	2
$Ch128$	128×128×128	950	1.54	-

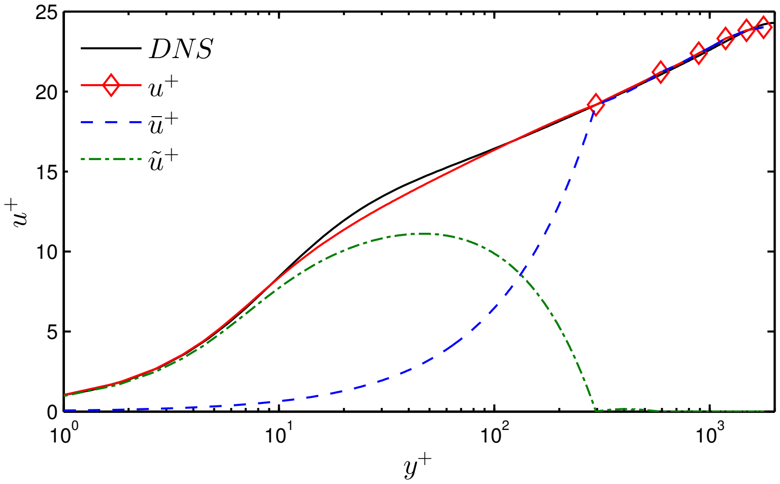


Figure 7.2.: WMLES: Decomposition of the mean velocity $u^+ = u_1/u_\tau$ with $u_\tau = \sqrt{\tau_w/\rho}$ of case $Ch12wm2$ at $\mathrm{Re}_\tau = 2{,}000$ into the linear and enrichment components and comparison to DNS data. Symbols indicate nodes. The first off-wall node is located at $y_1^+ = 333.3$.

7.3. Numerical Examples

In this section, the performance of wall modeling via function enrichment within the continuous FEM is investigated for turbulent channel flow at various Reynolds numbers, flow over periodic hills and backward facing step flow. The latter two examples assess the performance under separated boundary layer conditions and adverse pressure gradients.

7.3.1. Turbulent Channel Flow

A channel flow example analogous to Section 4.2 is considered, with a domain of the dimensions $2\pi\delta\times2\delta\times\pi\delta$ in streamwise, wall-normal, and spanwise direction, respectively. We discuss flows at friction Reynolds numbers $\mathrm{Re}_\tau = 590$, 950, 2,000, and 5,000 on coarse uniform meshes starting with 8×8×8 and up to 32×32×32 elements, see Table 7.1 for an overview. The column N_{wm} indicates the number of layers of enriched elements employed next to the solid bound-

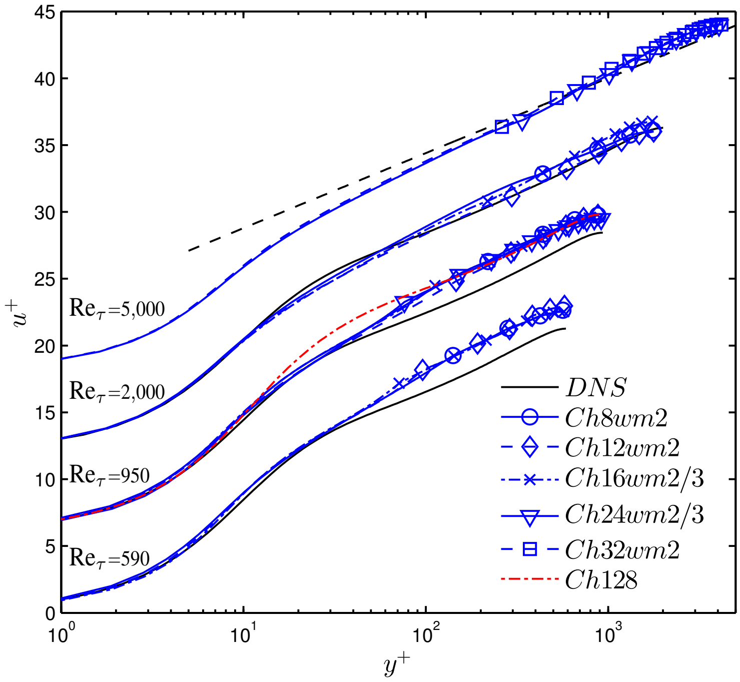


Figure 7.3.: WMLES: Normalized mean velocity for Re$_\tau$ = 590, 950, 2,000, and 5,000, each shifted upwards by 6 units for clarity. Symbols indicate nodes.

aries. Table 7.1 also compares the location of the first off-wall nodes in wall units, which are located between $y_1^+ = 73.8$ and $y_1^+ = 500$ for the different flows and given discretizations. For comparison, we also include an LES with resolved near-wall region, i.e., without wall model, at Re$_\tau$ = 950 on a discretization with 128×128×128 elements. The mean velocity is postprocessed on a large number of wall-parallel planes within the elements using the definition of the resolved flow field $\boldsymbol{u}_h$ (Equation (6.6)) such that the full velocity profile inside the enriched elements is made visible. The results presented in the following are labeled according to Table 7.1. They are compared with DNS data at Re$_\tau$ = 590 [187], Re$_\tau$ = 950 [6], and Re$_\tau$ = 2,000 [116]. The results for Re$_\tau$ = 5,000 are compared with $u^+ = \frac{1}{\kappa}\ln(y^+) + B$ with κ and B defined as in Section 2.1.3.

We commence the discussion of the results with Figure 7.2 showing the decomposition of the mean velocity of case $Ch12wm2$ at Re$_\tau$ = 2,000 similar to Figure 6.4. The normalized mean velocity profile $u^+ = u_1/u_\tau$ follows the DNS data closely and provides an excellent match despite the relatively coarse resolution. With the first off-wall node located at $y_1^+ = 333.3$, a large part of u^+ is in the first element represented by the enrichment component of the flow, $\widetilde{u}^+$, which also constitutes the largest part of the gradient at the boundary. The enrichment enables an accurate representation of the viscous sublayer and the buffer layer within the first off-wall cell. Further away from the wall in the second element layer, the contribution of the standard space $\bar{u}^+$ constitutes almost the whole solution.

The nondimensional mean velocity profiles for all 13 simulations are included in Table 7.1 is shown in Figure 7.3. A significant mesh independence is observed for all Reynolds numbers.

Even for discretizations consisting of only $8{\times}8{\times}8$ elements, the resolving power of the additional DOFs together with the law-of-the-wall shape functions yield results of acceptable to very good agreement with the DNS data. The normalized mean velocity is slightly overestimated for the friction Reynolds numbers $\mathrm{Re}_\tau = 590$ and 950. The simulation results at $\mathrm{Re}_\tau = 2{,}000$ and $5{,}000$ exhibit an excellent match with the reference data, however. Comparing the wall-resolved LES of case $Ch128$ at $\mathrm{Re}_\tau = 950$, a mean velocity of approximately equal quality as the WM-LES is obtained. The differences to LES data presented by Rasthofer and Gravemeier [215] are due to the fact that we do not consider the near-wall limit suggested in the original publication of the subgrid-scale modeling approach and possibly due to the smaller proportionality parameter C_{sgs} chosen here, as explained in Section 7.2.3.

The performance of the present wall model is further assessed via RMS profiles of the fluctuations u'^{+} in streamwise, v'^{+} in wall-normal, and w'^{+} in spanwise direction, as well as the RSS $(u'v')^{+}$ at $\mathrm{Re}_\tau = 950$ displayed in Figure 7.4. Considering u'^{+}, a distinct tendency to convergence for an increasing number of elements is observed. For w'^{+}, the predictions show a similar behavior as observed for u'^{+} whereas v'^{+} is generally predicted too small. The RSS is predicted quite accurately for all discretizations. The fact that the near-wall fluctuations are not of the same quality as the mean velocities is inherent in all wall models since the turbulent motions in the inner layer are not resolved. As expected, the RMS curve of the case $Ch128$ is in favorable agreement with the DNS data, since a significant amount of the near-wall turbulent structures is resolved.

From the results presented in this section it is concluded that relatively coarse resolutions can be used for the simulation of turbulent channel flow. The first off-wall node can be located at up to $y_1^+ = 500$ wall units.

7.3.2. Flow over Periodic Hills

We consider flow over a smoothly curved 2D-periodic hill as described and analyzed in Section 4.3 with a Reynolds number based on the hill height of $\mathrm{Re}_H = 10{,}595$ and $19{,}000$ to validate the wall modeling approach. Wall models for LES are challenged by this flow with a strong adverse pressure gradient that causes many models to produce deficient results. For example, Chen et al. [44] have found that their wall model based on the simplified TBLE underestimates the skin friction in the recirculation region as the convective term is neglected in that model. In wall modeling via function enrichment, all terms of the Navier–Stokes equations are satisfied discretely such that a better performance with respect to adverse pressure gradients may be expected. Temmerman et al. [249] investigated several wall functions and subgrid closures for LES and found that the location of the separation point has a major impact on the reattachment location. Also, the accurate prediction of the separation point of this flow is challenging employing steady RANS simulations [124]. Hybrid RANS/LES techniques have been analyzed by Breuer et al. [31] and Šarić et al. [224] who have shown that the RANS/LES interface should be located inside the boundary layer on the crest of the hill. Due to the construction of the present technique, there is no explicit interface between the statistical and the LES region such that these problems are not expected to occur.

A domain of the same dimensions as in Section 4.3 with periodic boundary conditions in the streamwise and spanwise direction and no-slip boundary conditions at the top and bottom walls is considered. A coarse mesh comprising $64{\times}32{\times}32$ elements and a refined, yet relatively

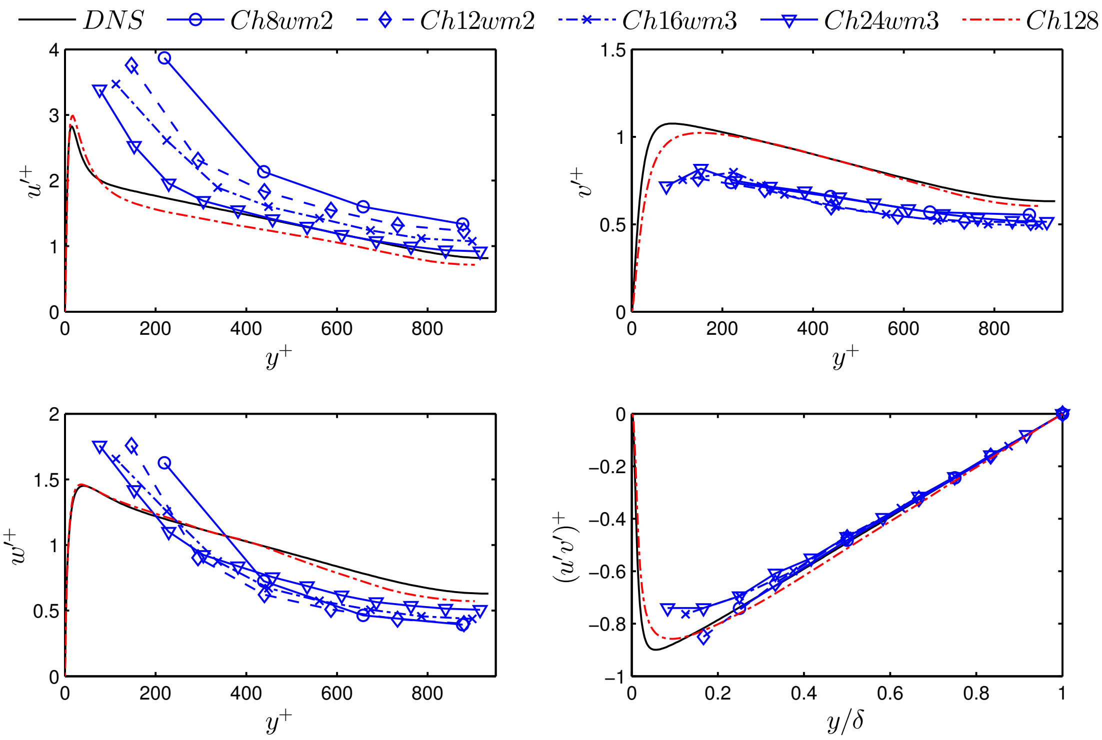


Figure 7.4.: WMLES: RMS of velocity fluctuations $u'^+ = \mathrm{RMS}(u'_1)/u_\tau$, $v'^+ = \mathrm{RMS}(u'_2)/u_\tau$, and $w'^+ = \mathrm{RMS}(u'_3)/u_\tau$, as well as the RSS normalized according to $(u'v')^+ = (u'_1u'_2)/u_\tau^2$ for $\mathrm{Re}_\tau = 950$.

coarse, mesh with 96×48×48 cells with uniform grid spacings in all directions and vertical grid lines as depicted in Figure 7.5 are employed. An overview of the simulation cases is given in Table 7.2, where the coarser grid is labeled PhC and the finer grid PhF for $\mathrm{Re}_H = 10{,}595$. A simulation without wall model is also investigated for comparison, which is labeled PhFNWM and employs the finer mesh. Considering $\mathrm{Re}_H = 19{,}000$, the same meshes are applied and labeled PhC19, PhF19, and PhFNWM19, respectively. The number of enriched element layers is $N_{\mathrm{wm}} = 4$ for both meshes at the upper and lower walls. Figure 7.6 shows the location of the first off-wall grid point over the x_1-coordinate. The first off-wall node is located at varying distance depending on resolution and Reynolds number up to approximately $y_1^+ = 216$ with minima near the zero-crossings of the wall shear stress. The mass flow is kept approximately constant over the simulation time by the adaptive algorithm in Equation (4.2) and statistics are sampled over 10,000 time steps, corresponding to approximately 45 flow-through times for the coarse mesh and 25 flow-through times for the fine mesh. For postprocessing, we exploit the fact that the enrichment vanishes on the nodes in Equation (6.6) such that only the nodal values of the standard FE space $\bar{\boldsymbol{u}}_B$ are taken into account. A preliminary setup including postprocessing routines for the present simulations was developed by Jäger [122] as part of this research.

An overview with respect to the reference data considered is also given in Table 7.2. The results for $\mathrm{Re}_H = 10{,}595$ are compared to the DNS data according to Chapter 4, labeled as DNS 10595, and the coarse mesh results discussed by Chen et al. [44] (CHDBA WMLES) with 64

Table 7.2.: WMLES: Simulation cases and resolutions of the periodic hill flow. $\mathrm{Re}_H = 10{,}595$: PhC coarse mesh with wall modeling, PhF refined mesh with wall modeling, PhFNWM refined mesh without wall modeling, DNS 10595 DNS, CHDBA WMLES and CHDBA WMLES F wall modeling based on a simplified TBLE and immersed interface method [44]. $\mathrm{Re}_H = 19{,}000$: PhC19 coarse mesh with wall modeling, PhF19 refined mesh with wall modeling, PhFNWM19 refined mesh without wall modeling, RM Exp experiments.

Case	$N_1 \times N_2 \times N_3$	Re_H	$x_{1,\mathrm{sep}}/H$	$x_{1,\mathrm{reatt}}/H$	N_{wm}
PhFNWM	96×48×48	10,595	0.2	3.68	-
PhC	64×32×32	10,595	0.25	3.77	4
PhF	96×48×48	10,595	0.25	4.91	4
DNS 10595	896×448×448	10,595	0.20	4.51 ± 0.06	-
CHDBA WMLES [44]	96×64×32	10,595	0.65	4.0	-
CHDBA WMLES F [44]	192×72×48	10,595	0.5	4.42	-
PhFNWM19	96×48×48	19,000	0.2	3.4	-
PhC19	64×32×32	19,000	0.24	2.58	4
PhF19	96×48×48	19,000	0.26	3.94	4
RM Exp [211]	-	19,000		3.94	-

cells in vertical direction. The separation and reattachment points are further compared to the fine mesh results by Chen et al. [44] (CHDBA WMLES F). The results for $\mathrm{Re}_H = 19{,}000$ are compared to experiments by Rapp and Manhart [211] (RM Exp).

We begin with a discussion of the results for the flow at $\mathrm{Re}_H = 10{,}595$. The skin friction coefficient c_f at the lower wall and pressure coefficients c_p at the upper and lower wall are compared to the DNS data in Figure 7.7. They are defined according to Equations (4.3) and (4.4). The skin friction is computed via the right-hand side residual (Equation (7.1)) and the pressure at the upper wall at $x_1 = 0$ is chosen as the reference pressure p_{ref}. The skin friction profiles computed by the WMLES are in close agreement with the reference data over large parts of the domain. Solely on the crest of the hill, the peaks between $x_1 = 8$ and 9 as well as $x_1 = 0$ and 1 are significantly overpredicted, but improve for the case PhF with higher resolution. This overprediction may be related to the local averaging operation of the wall shear stress applied during the construction of the shape functions. The minor recirculation at the top of the hill observed in the DNS data is not visible in the results of PhC and PhF. The separation and reattachment points are predicted with acceptable agreement via the zero-crossing of c_f as well and are summarized in Table 7.2. For the case PhFNWM without enrichment, large discrepancies including high peaks are visible near the hill crest. In the recirculation region, the skin friction is overestimated and the reattachment length is predicted shorter than for the cases with wall modeling. The skin friction coefficient is also compared to the results of Chen et al. [44], where c_f is significantly underestimated due to the neglected convective term as aforementioned.

The pressure curves of the present wall model are also in good agreement with the reference data and improve with resolution. The case PhC shows discrepancies both at the lower and upper wall, which are due to the coarseness of the resolution. The case PhFNWM overpredicts the pressure in the recirculation bubble and exhibits negative peaks on the hill crest. In the recovery

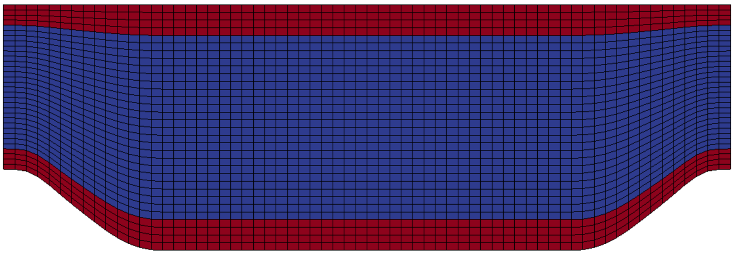


Figure 7.5.: Grid of case PhC. Enriched elements are colored red and standard linear elements blue.

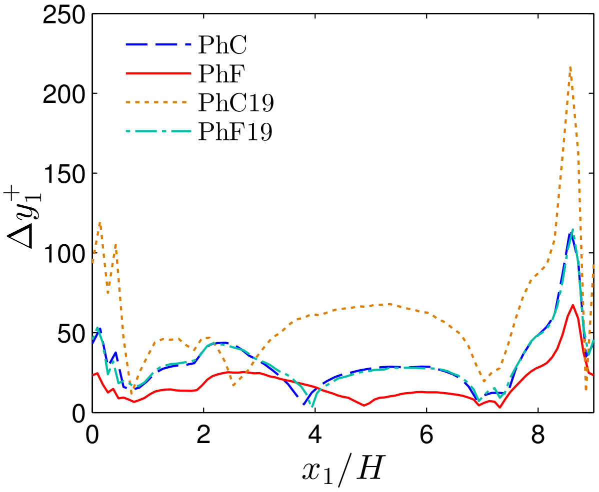


Figure 7.6.: WMLES: Location of the first off-wall node in wall units over the streamwise coordinate of the periodic hill case.

region, the estimation is comparable to the coarse mesh with wall modeling, PhC. At the upper wall, the prediction with wall model is superior compared to the one without.

The profiles of the mean velocity in streamwise and vertical direction, the RSS $u_1'u_2'$, and the TKE of the case $\mathrm{Re}_H = 10{,}595$ are compared with the DNS data of Chapter 4 in Figure 7.8. The mean velocity u_1 exhibits discrepancies with the reference data for case PhC in the reattachment and recovery region whereas the finer mesh PhF results in good agreement with the reference data. Without wall model, PhFNWM predicts u_1 with similar quality as the coarse mesh with wall modeling, PhC. Also regarding the mean velocity u_2 in vertical direction, good results are obtained for the fine mesh including wall modeling, PhF. For the other cases, u_2 is underestimated above the recirculation bubble due to the shorter reattachment length. The RSS $u_1'u_2'$ is heavily overpredicted for the cases PhC and PhFNWM near the crest of the hill and in the shear layer between the recirculation region and the bulk flow. Refinement leads to a good match with the reference data for the case PhF. Finally, the TKE distributions are only predicted accurately everywhere with PhF whereas PhC and PhFNWM overpredict its magnitude inside the recirculation zone.

The good results observed for $\mathrm{Re}_H = 10{,}595$ motivate an application of the wall model to a higher Reynolds number. For this second assessment, we choose the next higher Reynolds num-

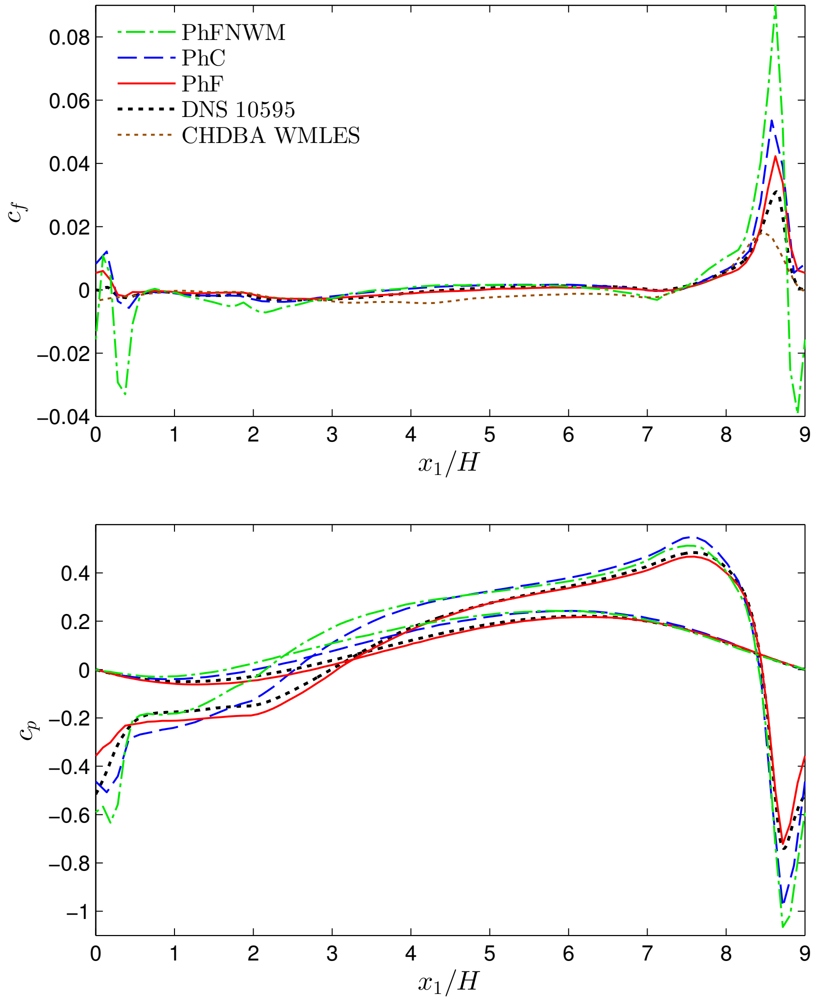


Figure 7.7.: WMLES: Skin friction (top) and pressure (bottom) coefficients for flow over periodic hills at $\mathrm{Re}_H = 10{,}595$. The shallower c_p curves correspond to the upper wall.

ber, $\mathrm{Re}_H = 19{,}000$, and consider the same meshes. The results in Figure 7.9 include the mean velocities u_1 and u_2 as well as RSS profiles $\overline{u_1'u_2'}$ and are compared to the experimental data, labeled RM Exp. The quality of the fine mesh with wall modeling is very similar to the results at Reynolds number $\mathrm{Re}_H = 10{,}595$ discussed above. The reattachment length is also predicted perfectly with this mesh. The coarser case PhC shows slightly worse predictions in the recirculation, reattachment and recovery region for the mean velocity u_1 and in the recirculation bubble for the RSS. Also, the reattachment length is predicted significantly too short, which may be due to an overly coarse mesh. The case without wall model, PhFNWM19, exhibits results of quality between the coarse and fine wall-modeled simulations. Here, another defect is highlighted: significant oscillations in the mean velocity profiles are visible especially at the lower wall and in the vicinity of the hill. Such oscillations are not visible for the wall-modeled computations and show another advantage of the wall model.

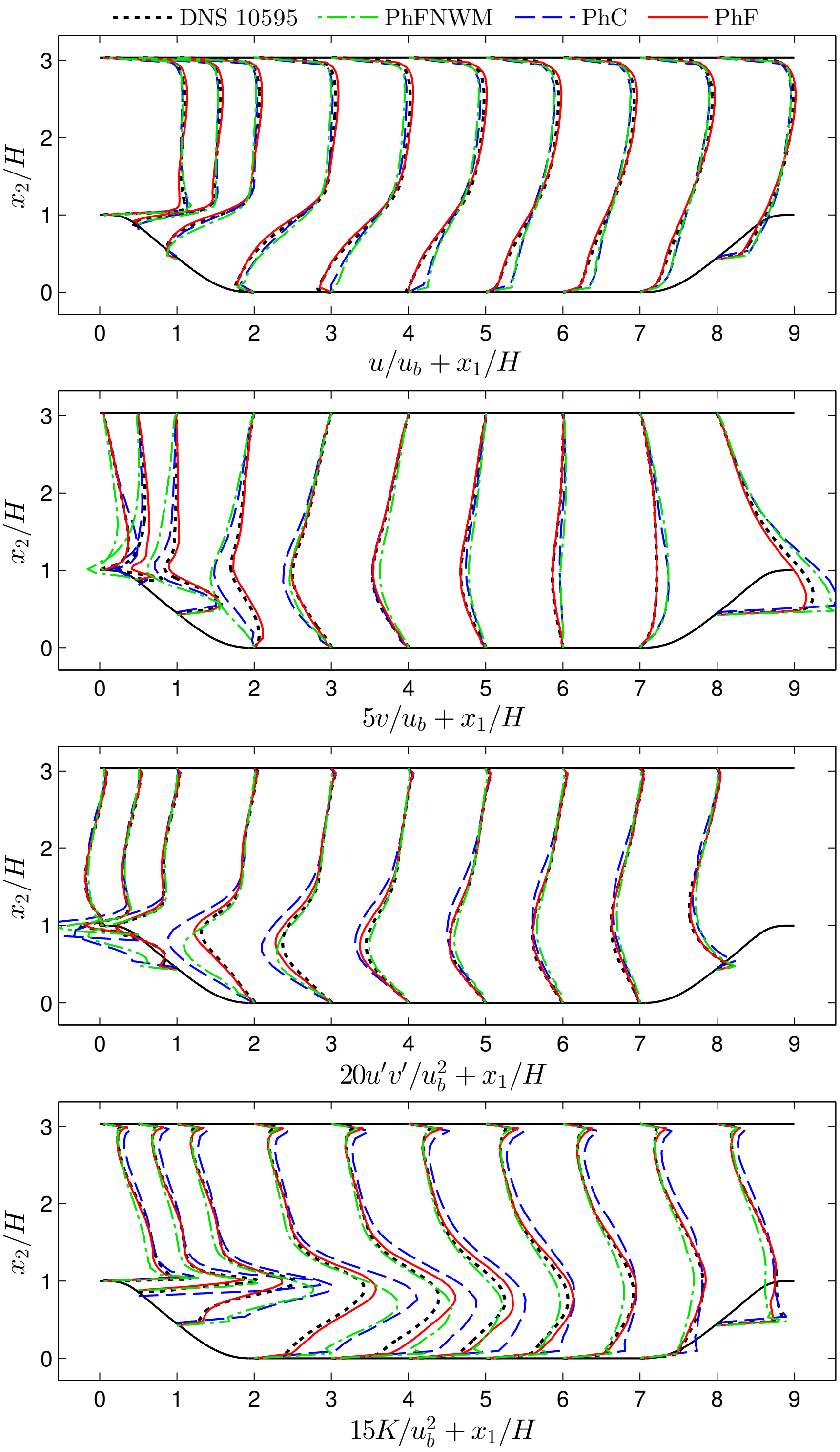


Figure 7.8.: WMLES: Mean velocity u_1 in x_1 and u_2 in x_2-direction, RSS $u_1'u_2'$ and TKE K for the periodic hill at $\mathrm{Re}_H = 10{,}595$.

From the investigations of flow over periodic hills the following conclusion may be drawn. The present enrichment-based wall model exhibits favorable characteristics with respect to sep-

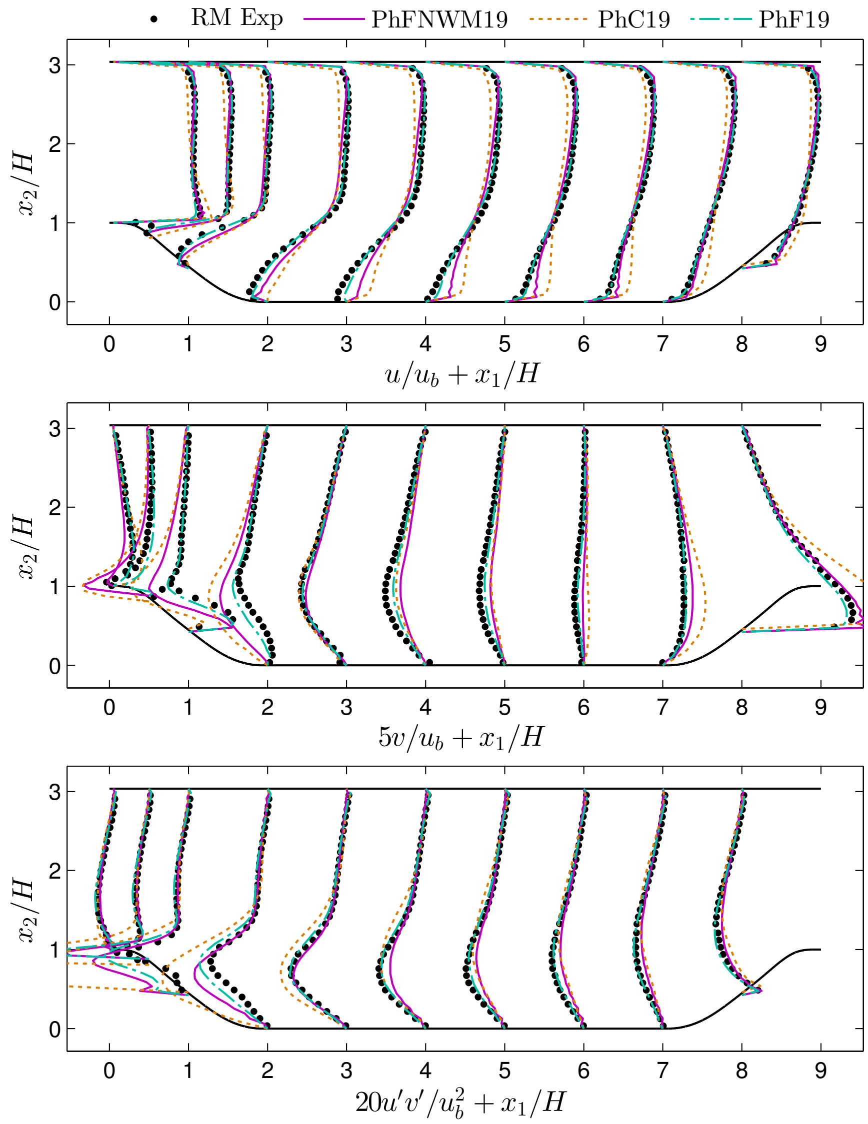


Figure 7.9.: WMLES: Mean velocity u_1 in x_1 and u_2 in x_2-direction as well as RSS $u_1'u_2'$ for the periodic hill at $\mathrm{Re}_H = 19{,}000$.

arated flows as well as under adverse pressure gradients and allows the use of comparably coarse meshes. Further simulations of the periodic hill flow using fully unstructured tetrahedral meshes confirmed the present results [111].

7.3.3. Backward Facing Step Flow

We assess the wall modeling approach further with flow over a backward facing step at $\mathrm{Re}_h = 5{,}000$ with an expansion ratio of $\mathrm{ER} = 1.2$ as studied experimentally by Jovic and Driver [130]. DNS data of a similar configuration at a Reynolds number of $\mathrm{Re}_h = 5{,}100$ is available through computations by Le et al. [170], and results for WMLES have been presented, e.g., by Chen et al. [44] mentioned earlier, who encountered difficulties predicting the correct skin friction and reattachment point for this flow as well.

Table 7.3.: WMLES: Simulation cases and resolutions of backward facing step flow. BFS_NWM without wall modeling, BFS_WM3 with wall modeling, $BFS\_J\&D\_EXP$ experiments, BFS_LMK_DNS DNS.

Case	Re_h	$x_{1,\mathrm{reatt}}/h$	N_{wm}
BFS_NWM	5,000	13.49	-
BFS_WM3	5,000	6.78	3
$BFS\_J\&D\_EXP$ [130]	5,000	6.0 ± 0.15	-
BFS_LMK_DNS [170]	5,100	6.28	-

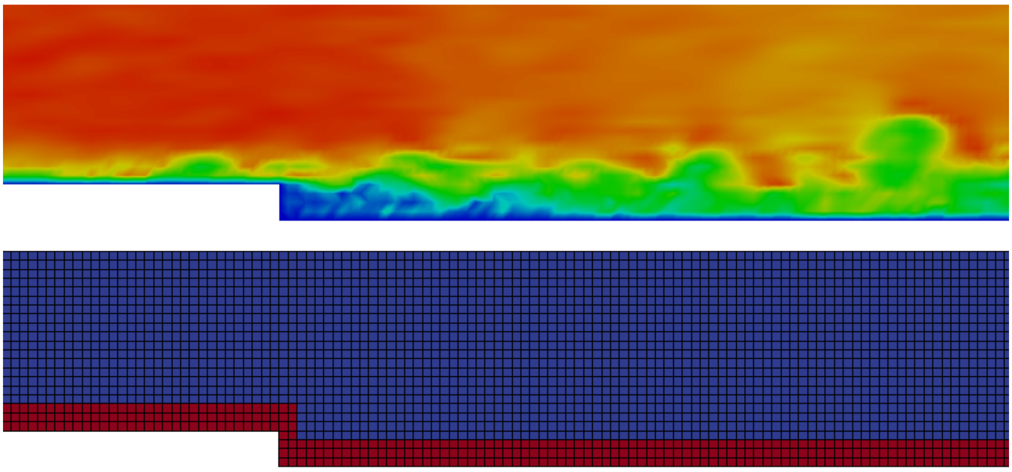


Figure 7.10.: WMLES: (Top) Instantaneous velocity magnitude over the backward facing step: red indicates high and blue low values. (Bottom) Mesh in the vicinity of the step: enriched elements are colored red and standard elements blue.

The computational domain behind the step is of the dimensions $30h \times 6h \times 3h$ in streamwise, wall-normal and spanwise direction, respectively. The domain extends $30h$ upstream of the step and the velocity is prescribed at the inflow boundary using mean DNS data of a turbulent boundary layer at a similar Reynolds number [239] with an additional random perturbation of 10% of the center line velocity u_c. The inflow data is only prescribed on the standard space $\bar{\boldsymbol{u}}_h$ whereas the enrichment $\widetilde{\boldsymbol{u}}_h$ is set to zero for simplicity. Periodic boundary conditions are applied in the spanwise direction and slip boundary conditions at the upper wall. The domain is meshed uniformly with four elements per step height h in all space dimensions. For the case with wall modeling, three rows of elements at the lower wall are enriched, including the inflow region and the step. The resulting mesh is quite coarse and displayed in Figure 7.10 along with a contour plot of the instantaneous velocity. For the statistical results presented in the following, the quantities are averaged over 5,000 time steps starting after the initial transient. As for the periodic hill benchmark, only the nodal values of the standard FE component are considered for postprocessing.

An overview of the results discussed is provided in Table 7.3. The computation including wall modeling is labeled as BFS_WM3 and compared to the same mesh where the enriched elements are replaced by standard elements, labeled BFS_NWM. We compare the results with

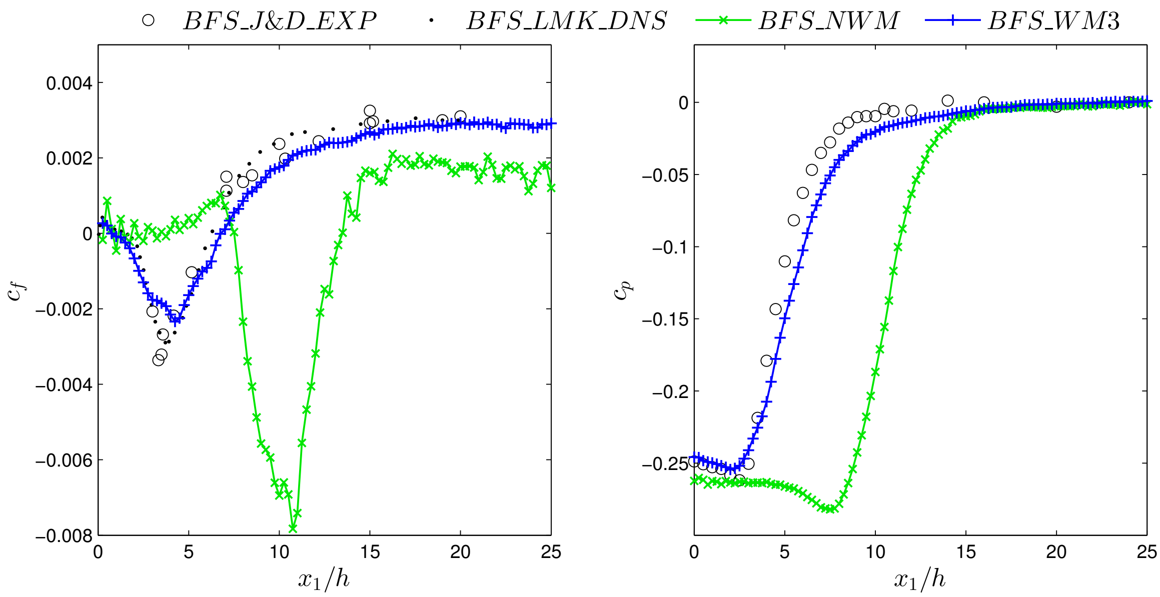


Figure 7.11.: WMLES: Skin friction (left) and pressure (right) coefficients for the flow over a backward facing step at $\mathrm{Re}_h = 5{,}000$.

the experiments by Jovic and Driver [130] labeled as $BFS\_J\&D\_EXP$ and the skin friction is additionally evaluated against the DNS data by Le et al. [170] BFS_LMK_DNS.

Again we begin with a discussion of the distribution of the skin friction and the pressure coefficients along the lower wall. They are defined analogous to Equations (4.3) and (4.4) with the reference velocity u_c and the reference pressure located at $x_1 = 24h$. The results of the case BFS_WM3 displayed in Figure 7.11 exhibit a favorable agreement with the reference data. In contrast, the computation without wall model, BFS_NWM, does not give physically reasonable results. The peak in negative skin friction is very large and shifted downstream by several step heights. Further, significant oscillations are observed. Accordingly, the reattachment length defined as the zero-crossing of the skin friction coefficient is predicted as $x_{1,\text{reatt}} = 6.78h$ for the case with wall modeling, which matches the references of $x_{1,\text{reatt}} = 6.0h$ and $x_{1,\text{reatt}} = 6.28h$ quite well. In contrast, the simulation without wall modeling predicts $x_{1,\text{reatt}} = 13.49h$. An overview with respect to reattachment lengths is also given in Table 7.3. The prediction of the pressure coefficient shows a similar quality as for the friction coefficient. Including wall modeling, the curve follows the reference data closely, while the one without wall model is delayed by several step lengths.

The mean streamwise velocity, RMS velocity fluctuations of the streamwise and wall-normal components and the RSS are displayed at six locations in Figure 7.12. It may be observed in the graphs that both the velocity and fluctuations in front of the step are in good agreement with the reference data, implying that the simple procedure of applying turbulent inflow boundary conditions gives useful results. For the case without wall modeling, the fluctuations are not reproduced correctly, however.

The mean velocity behind the step matches the reference data very well for the case BFS_WM3 including wall modeling. It is mentioned here that the velocity is only postprocessed on the element nodes, which are connected with straight lines in the graph for simplicity.

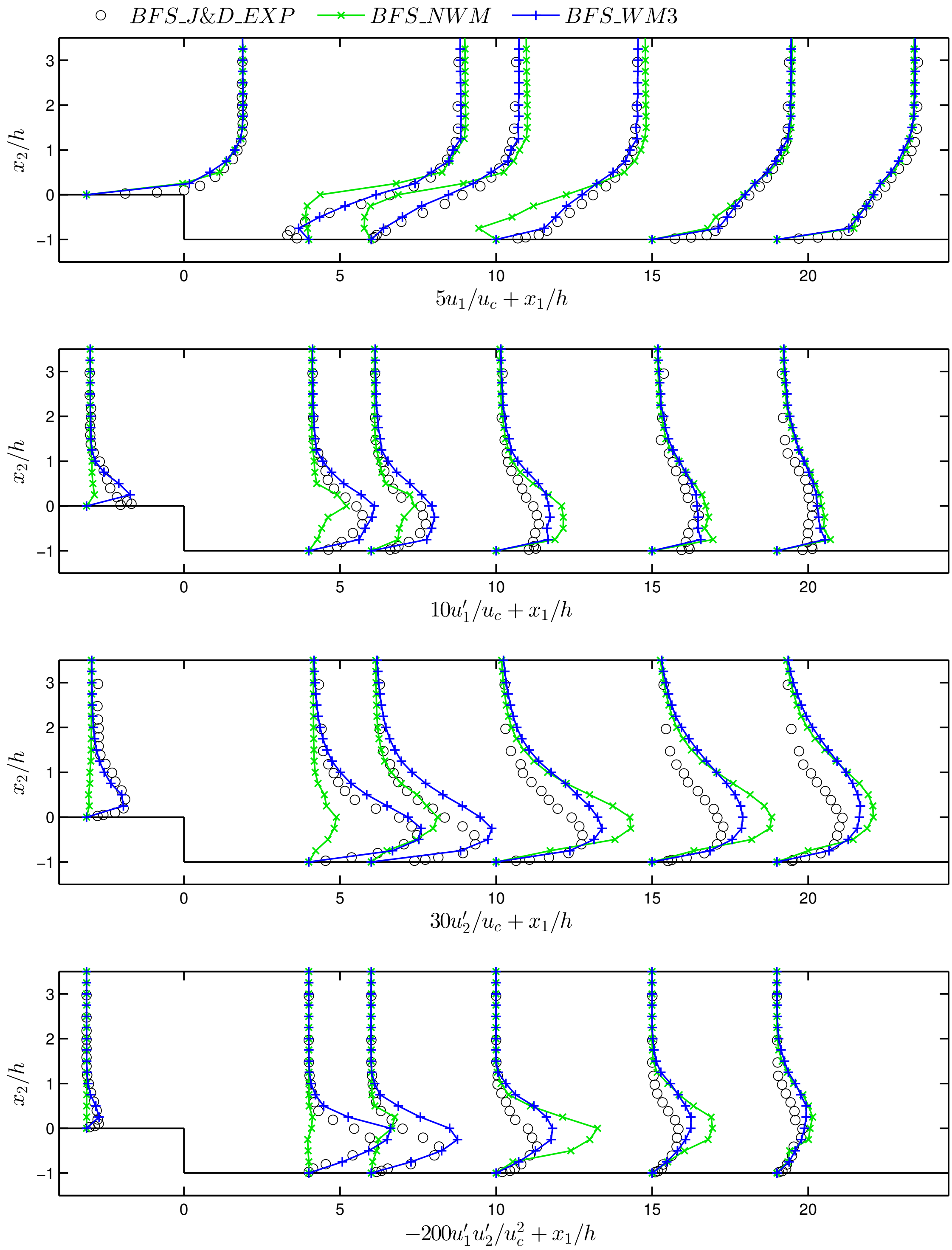


Figure 7.12.: WMLES: Mean velocity u_1 in x_1 direction, RMS values of the fluctuations u_1' and u_2' in x_2-direction, and the RSS $u_1'u_2'$ for the backward facing step flow.

Therefore, the detailed velocity distribution at the second location inside the recirculation region is not shown in the graph. Without the enrichment, the result is barely physical and the size of the recirculation is significantly overpredicted.

The RMS quantity u_1' is predicted very well for the case with wall modeling. Even without wall modeling the match is quite good in the recovery region, but lower magnitudes are observed inside the recirculation. The RMS profiles of the fluctuation in x_2-direction, u_2', are also predicted well if the wall model is included. In the recovery region, small discrepancies are visible, however. It may be assumed that this behavior is due to the coarseness of the mesh in the shear layer above the recirculation and is not directly related to the wall model. The simulation without wall model yields deficient predictions already at the first location. The RSS is predicted with acceptable accuracy at the first locations but an overestimation is observable around the fourth location. Without wall model, the RSS is neither predicted accurately at the inflow nor behind the step.

From the backward facing step flow investigated with and without wall model in this section, we find further evidence that wall modeling via function enrichment gives excellent results in separated flow regimes. The method is robust with respect to kinks in the boundary conditions and ambiguous wall-normal vectors. Its strengths are accurate predictions of the skin friction and pressure coefficients as well as mean velocity profiles with coarse meshes, but even turbulence quantities are estimated well. Additional results on this example with a different mesh were published in [154] and showed that the cell aspect ratio is uncritical for the present method.

7.4. Summary

Wall modeling via function enrichment has been applied to the continuous FEM. The wall function considered in this chapter, Spalding's law, is not prescribed as a boundary condition, but offered as an additional component of the solution in a consistent way. The first off-wall node can be placed at up to $y_1^+ \sim 500$ and the wall shear stress is still captured by the additional shape functions based on the wall function.

The method has been validated with attached and separated boundary layer flows, present in turbulent channel flow, flow over periodic hills, and backward facing step flow. Turbulent channel flow could be computed with relatively coarse meshes, and the mean velocity profiles may be computed accurately and have shown almost no dependence on the mesh. The RMS values of the velocity fluctuations do not exhibit the same level of accuracy, since the turbulent motions in the inner layer are not resolved. Flow over periodic hills and backward facing step flow exhibit the high potential of the presented method for many practical applications with high adverse pressure gradients and under separated flow conditions. The reason for the high level of accuracy in these simulations is the full consistency of the method, i.e., that the Navier–Stokes equations are satisfied in a discrete sense in the whole boundary layer, see Section 6.2.

The near-wall turbulent motions are not resolved by the present method, but modeled statistically, whereas the large turbulent motions are resolved in the outer layer on the same grid. Wall modeling via function enrichment can therefore be seen as a branch of wall-modeled DES. This similarity is developed further in Chapter 9, where the SA model is employed in RANS mode in the near-wall region and in LES mode in the outer layer. A further development of ideas discussed in the present chapter is presented in Chapter 10, where the *a priori* separation of the function space into an eddy-resolving and an averaged component is used to apply tailored turbulence models for each of these two scales.

8

Application II: Wall-Modeling for RANS in High-Order Discontinuous Galerkin

Resolving the sharp velocity gradient present in turbulent boundary layers is particularly costly in RANS simulations, as the wall-parallel resolution requirements are comparably low. Wall functions are therefore frequently used to economize computer time and storage requirements by placing the first node in the logarithmic region and specifying appropriate boundary conditions (see, e.g., the reviews in [64, 169] and the discussion in Section 6.2.5). However, wall functions are often not valid in strong nonequilibrium boundary layers and it may be difficult to define consistent boundary conditions for turbulence quantities at general off-wall locations, especially inside the buffer layer between $5 < y^+ < 30$ [64, 134]. Special wall functions have been designed in order to enhance the performance in nonequilibrium flow conditions, see [134, 143, 207, 255] and the discussion in Sections 6.2.2 and 6.3.2.3. Suitable boundary conditions for turbulence quantities have been discussed for example in [134].

In the present chapter, we transfer the idea of wall modeling via function enrichment to RANS and the high-order DG method. The RANS turbulence model takes into account all turbulent motions in the whole domain, such that no modifications have to be made to the turbulence modeling approach. The application to RANS is the most straightforward example of wall modeling via function enrichment since the whole simulation methodology reduces to taking additional shape functions into account in the spatial discretization. The SA model is considered for these investigations in the formulation as it is described and implemented in the high-order DG framework in Chapter 5. The DG method considered herein yields a simpler formulation of the enrichment as compared to the continuous FEM used in the preceding chapter. Also, the SA model offers the possibility of a straightforward extension of the approach to detached-eddy simulation, which will be presented in the subsequent chapter.

We begin with a summary of the considered function enrichment space in the first section and introduce an extension of the adaptation algorithm that only takes into account the enrichment when it is needed. Subsequently, benchmark results are computed using the two familiar examples of turbulent channel flow and flow over periodic hills. The work presented in this chapter is based on Krank et al. [152].

8.1. Enrichment Space: Enrich Only When Needed

The concept of function enrichment allows the resolution of the sharp spatial velocity gradient present at the wall with very coarse meshes up to $y_1^+ \sim 1{,}000$ in the context of RANS, while preserving consistency and flexibility in nonequilibrium boundary layers. This is achieved by inserting a wall function in the function space of the Galerkin method in addition to the standard polynomial component. Since the full Navier–Stokes equations are then solved in the boundary layer including the no slip boundary condition, good results are expected even in nonequilibrium conditions. The general framework for such a function enrichment was already introduced in Section 6.3. The methodology is further extended in the following sections by an adaptation algorithm, which automatically takes the enrichment into account when it is needed.

8.1.1. Enrichment Space

We consider wall modeling via function enrichment within the DG method according to Section 6.3.2.1. As an enrichment function, Spalding's law is used according to Equation (6.7) with the constants κ and B as defined in Section 2.1.3, since the results obtained with this function in the previous chapter are very promising. As it was discussed in Section 6.3.2.3, the analytical wall function that solves the SA equation in [7] would also be highly interesting as an enrichment function, since this function would yield an even higher degree of consistency in the approach. The enrichment function is in this chapter weighted using the polynomial degrees $l = \{0, 1, 2\}$ with an emphasis on the linear case ($l = 1$), and arguments for this choice are discussed in the context of a numerical example in Section 8.2.1. The weighting of the wall function with a linear FE space gives the method a high degree of flexibility in adapting the enrichment to separated flow conditions while the number of additional DOFs introduced by the enrichment is often small. Since we consider higher polynomial degrees in this chapter ($k = 4$) for the polynomial velocity component, the eddy viscosity working variable, and the pressure, in contrast to the low-order method employed in Chapter 7, it is sufficient to enrich a single layer of cells near the wall. As it was noted in Section 6.3.2.1, the blending of the function spaces at the position, where the enrichment ends, is not an issue within DG, since neighboring cells can have different shape functions due to the weak coupling of the cells. The composition of the resulting function space is illustrated in Figure 6.2 for an equilibrium boundary layer and for a nonequilibrium boundary layer profile. The wall shear stress is discretized using linear continuous shape functions ($m = 1$) as it was introduced in 6.3.2.4 and updated every time step in order to take into account spatial and temporal variations of the flow in the model. This adaptation algorithm is enhanced in the following.

8.1.2. Enhancement of the Adaptation Scheme

The enrichment is only necessary when the first off-wall cell spans a y^+-range up to a certain y^+_{max} at which the polynomial of degree k is no longer capable of predicting the correct velocity profile including the gradient in the viscous sublayer. In addition, if the enrichment is employed in elements spanning only a small range of y^+ units, the polynomial and enrichment shape functions may become close to linearly dependent, resulting in a degradation of conditioning and robustness. We therefore switch the enrichment on or off according to a certain criterion. This criterion is evaluated prior to every time step. The L^2-projection in Equation (6.14) allows for such nonconforming function spaces in time, i.e., a velocity solution $\boldsymbol{u}_h^{n,\text{old}}$ containing the enrichment may be projected onto a purely polynomial solution $\boldsymbol{u}_h^{n,\text{new}}$ and vice versa.

This raises the question of how to choose the criterion and a specific value of y^+_{max}. In Figure 8.1, we analyze the error occurring if a polynomial of degree $k = 4$ is used to approximate a typical boundary layer profile, given as the enrichment function ψ, for $y^+_{\text{max}} = \{10, 20, 30, 40\}$.

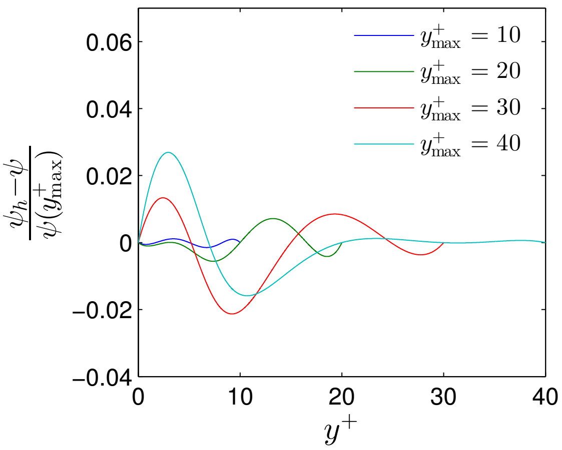


Figure 8.1.: Analysis of several values of $y^+_{\max}$. The quantity ψ_h represents an interpolation of ψ onto the Gauss–Lobatto nodes of an element with $k = 4$ with the end points located at $y^+ = 0$ and $y^+ = y^+_{\max}$, respectively.

This function is interpolated onto the Gauss–Lobatto nodes and the error is normalized by the value $\psi(y^+_{\max})$. At $y^+_{\max} = 10$, the error of the interpolated function is very small, so if the first off-wall cell spans a y^+-range up to this value, the boundary layer is fully resolved by the polynomial component. This result has lead to the recommendation of $y^+_{1e} < 10$ in wall-resolved simulations used in Section 5.4.1. The error is larger for higher values of $y^+_{\max}$ and reaches approximately 3% in the case $y^+_{\max} = 40$. The final choice of this parameter is a trade-off between robustness and accuracy and values between 20 and 30 have proven suitable in the investigations. In this chapter, we perform wall-enriched turbulent channel flow simulations with the first off-wall cell size down to $\Delta y^+_{1e} = 25$ without difficulty and set $y^+_{\max}$ to 30 for the periodic hill flow example to allow for robustness near separation and reattachment locations. A further analysis of this problem in the context of the method presented in Chapter 10 showed that $y^+_{\max} = 30$ is even suitable in eddy-resolving simulations, so this parameter is used in the remainder of this thesis if not specified otherwise. In the implementation of this criterion in two and three space dimensions, we compute the numerical value of y^+ on all quadrature points in $\widetilde{\Omega}_e$ and toggle the enrichment in that cell on if at least one quadrature point lies in $y^+ > y^+_{\max}$.

The present adaptation technique replaces the clipping of the wall shear stress introduced in Section 6.3.2.4 in order to avoid $\tau_w \rightarrow 0$ and thus $y^+ \rightarrow 0$ near separation or reattachment points.

In summary, at the beginning of each time step, we perform the following steps in our algorithm according to the present adaptation scheme and the steps discussed in Section 6.3.2.

i. Compute new wall shear stress $\tau_{w,h}$.

ii. Evaluate criterion whether element is enriched.

iii. Precompute Spalding's law and its derivatives on each quadrature point.

iv. Precompute element-wise inverse mass matrix terms of a scalar mass matrix for all enriched elements according to Section 6.3.3.2.

v. Perform element-wise L^2-projection according to Equation (6.14).

vi. Recompute vectors of the time integration scheme.

8.1.3. Implementation

The resulting function space for the velocity is inserted in the weak forms presented in Chapter 5 and integrated over all cells and faces and the enrichment cells are integrated using Gaussian quadrature rules of higher order according to the quadrature guidelines formulated in Table 6.3. The inverse mass matrix is applied in this chapter using the matrix-based approach via LU factorization as discussed in Section 6.3.3.2, since the faster matrix-free variant was not yet available at the time when the simulations were performed. With respect to the implicit steps of the numerical scheme, the standard iterative solvers according to Section 3.5.2 give similar iteration counts compared to the standard polynomial case. From this fact we conclude that conditioning is not an issue. The no-slip Dirichlet boundary condition is applied in a strong sense on the viscous term as discussed in Section 5.3.1 in order to allow very large cell sizes in the first off-wall element up to $y_{1e}^{+} = 5{,}000$, which would not be possible with the weak Dirichlet boundary conditions presented in Chapter 3. As an alternative, it would be possible to use a higher interior penalty stabilization parameter on the boundary faces in the viscous step; this approach was chosen in Chapter 10. It is noted that the latter weak Dirichlet boundary conditions are even more solver-friendly in three space dimensions, and they are recommended for future applications of the wall model. The enrichment is implemented in the INDEXA code with minimal modification of the original implementation by using the wrapper classes introduced in Section 6.3.3.4.

8.2. Numerical Examples

We investigate wall modeling via function enrichment for RANS with the same two examples as in the context of the standard RANS solver in Section 5.4, i.e., turbulent channel flow and flow over periodic hills. These two benchmark examples provide insight into the performance regarding wall-attached flow in the first case and separated flow with a high adverse pressure gradient in the second setup. All computations are carried out with a scheme of temporal accuracy of second order (BDF2) and we take the spatial polynomial degrees of $k = 4$ for the standard velocity component, pressure, and eddy viscosity, in order to be able to compare the results to the wall-resolved simulation cases presented in Section 5.4. The polynomial degree of the enrichment l is chosen as $l = 1$ for most simulation cases, but results for $l = 0$ and $l = 2$ are also presented.

8.2.1. Turbulent Channel Flow

We consider turbulent flow in a 2D plane channel with the same setup as in Section 5.4.1. The velocity solution is postprocessed at simulation time using the definition of $\boldsymbol{u}_h$ at sufficiently many y^+ levels such that the full velocity profile may be compared to the reference data. The following numerical experiments are separated into three groups investigating the independence of the Reynolds number with the same mesh, mesh refinement with a constant Reynolds number,

Table 8.1.: SA-RANS: Channel flow cases and resolutions. N_{ei} number of elements per spatial direction i, Re_τ friction Reynolds number, γ mesh stretching parameter, and Δy^+_{1e} width of first off-wall element. All computations employ $k = 4$ as the polynomial degree of the standard space as well as a single enriched element row at the wall. The polynomial degree of the enrichment space is $l = 1$ for most cases with one example for $l = 0$ and $l = 2$, respectively. The cases ch_SA_Ref are the same as in Table 5.1.

Case	$N_{e1}\times N_{e2}$	Re_τ	l	γ	Δy^+_{1e}
$ch\_N8^2\_k4l1$	8×8	180	1	-	45
	8×8	395	1	-	99
	8×8	590	1	-	118
	8×8	950	1	-	238
	8×8	2,000	1	-	500
	8×8	5,200	1	-	1,300
	8×8	10,000	1	-	2,500
	8×8	20,000	1	-	5,000
$ch395\_N8^2\_k4l1$	8×8	395	1	-	99
$ch395\_N16^2\_k4l1$	16×16	395	1	-	49
$ch395\_N32^2\_k4l1$	32×32	395	1	-	25
$ch395\_N8\times16\_k4l1$	8×16	395	1	-	49
$ch395\_N8\times32\_k4l1$	8×32	395	1	-	25
$ch395\_N8^2\_k4l0$	8×8	395	0	-	99
$ch395\_N8^2\_k4l2$	8×8	395	2	-	99
$ch50000\_N16^2\_k4l1$	16×16	50,000	1	2.0	1,175
$ch100000\_N16^2\_k4l1$	16×16	100,000	1	2.25	1,664
ch_SA_Ref	16×16	180	-	1.8	5.5
	16×16	395	-	2.0	9.3
	16×16	590	-	2.25	9.8
	32×32	950	-	2.0	9.8
	32×64	2,000	-	2.0	9.7

and the application to high Reynolds numbers. All simulation cases and resolutions are presented in Table 8.1.

The first investigations discussed here employ all the same mesh as visualized in Figure 8.2 of 8×8 equally distributed cells and including a single enriched element row with $l = 1$ at the walls. With this spatial discretization, the enrichment constitutes only 4% of the overall number of DOFs. We perform simulations using friction Reynolds numbers $\mathrm{Re}_\tau = u_\tau \delta/\nu$ in accordance with reference DNS at $\mathrm{Re}_\tau = 180$, 395, and 590 [187], $\mathrm{Re}_\tau = 950$ [6], $\mathrm{Re}_\tau = 2{,}000$ [116], and $\mathrm{Re}_\tau = 5{,}200$ [172]. Further friction Reynolds numbers of $\mathrm{Re}_\tau = 10{,}000$ and $\mathrm{Re}_\tau = 20{,}000$ are included and compared to the linear and log-laws in the respective y^+ regions. These simulation setups result in locations of the first off-wall element interface in a y^+ range between $y^+_{1e} = 45$ and 5,000 wall units, see Table 8.1. As a y^+ criterion, we consider the width of the first off-wall cell as more meaningful compared to the first off-wall Gauss–Lobatto point used in Part I, since

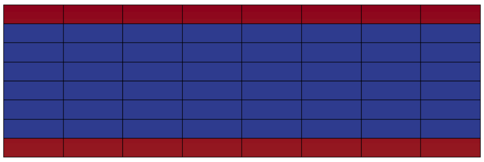


Figure 8.2.: Mesh for turbulent channel flow computations with $N = 8^2$ elements. Enriched elements are colored red and standard polynomial elements blue, i.e., a single element layer is enriched at the walls.

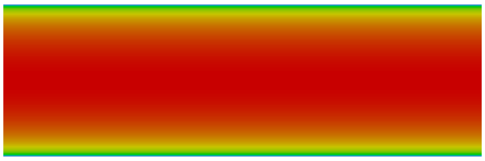


Figure 8.3.: SA-RANS: Numerical solution of the streamwise velocity of the case $ch395\_N16^2\_k4l1$ at the end of the simulation. Red indicates high and blue low velocity

the width of the first off-wall cell is equal to the thickness of the enrichment layer. The velocity solution is illustrated in Figure 8.3 for the case $ch395\_N16^2\_k4l1$. As additional reference data, we have computed wall-resolved RANS simulations, i.e., without enrichment on a fine mesh ($\Delta y^+_{1e} < 10$ according to Section 8.1), using the present numerical scheme. These simulation cases have already been presented in Section 5.4.1, where they were compared to DNS data. The results of the normalized velocity u^+ are depicted in Figure 8.4 and exhibit excellent agreement with the reference data including the inner layer and viscous sublayer despite the substantial difference in resolution respective wall units.

In the middle of the channel, the velocity is overpredicted to a minor extent for $\mathrm{Re}_\tau = 180$ in comparison to the DNS data and marginally underpredicted for $\mathrm{Re}_\tau = 20{,}000$, the error is acceptable, however. Also, the velocity is slightly overpredicted in the buffer layer between $5 < y^+ < 30$ in comparison to DNS data. The reason for this behavior is assumed to be twofold: Firstly, the enriched and the wall-resolved RANS computations show a close agreement in the buffer layer, indicating that this error is related to the SA model, see also Section 5.4.1. Secondly, the wall-resolved RANS computations show a very good agreement with DNS data in the log-layer while some of the enriched cases show a small mismatch, indicating that this error may be a discretization error. The source of the latter discretization error is presumably that Spalding's law results in a slightly different velocity distribution in the buffer layer as compared to the SA model [7, 134], see also the comparison of the wall function resulting from the SA model with Spalding's law and DNS data in Figure 6.5. Aiming at fine-tuning this approach, it would in future research certainly be valuable to investigate alternative enrichment functions, in particular the wall functions implicitly given through the SA equation as specified in Section 6.3.2.3, in order to minimize the discretization error.

Simple wall function approaches are often prone to inaccuracies during mesh refinement, especially when the coupling location moves inside the buffer layer. A refinement study is therefore performed for the case $\mathrm{Re}_\tau = 395$ using three successive refinement levels from 8×8 to 32×32

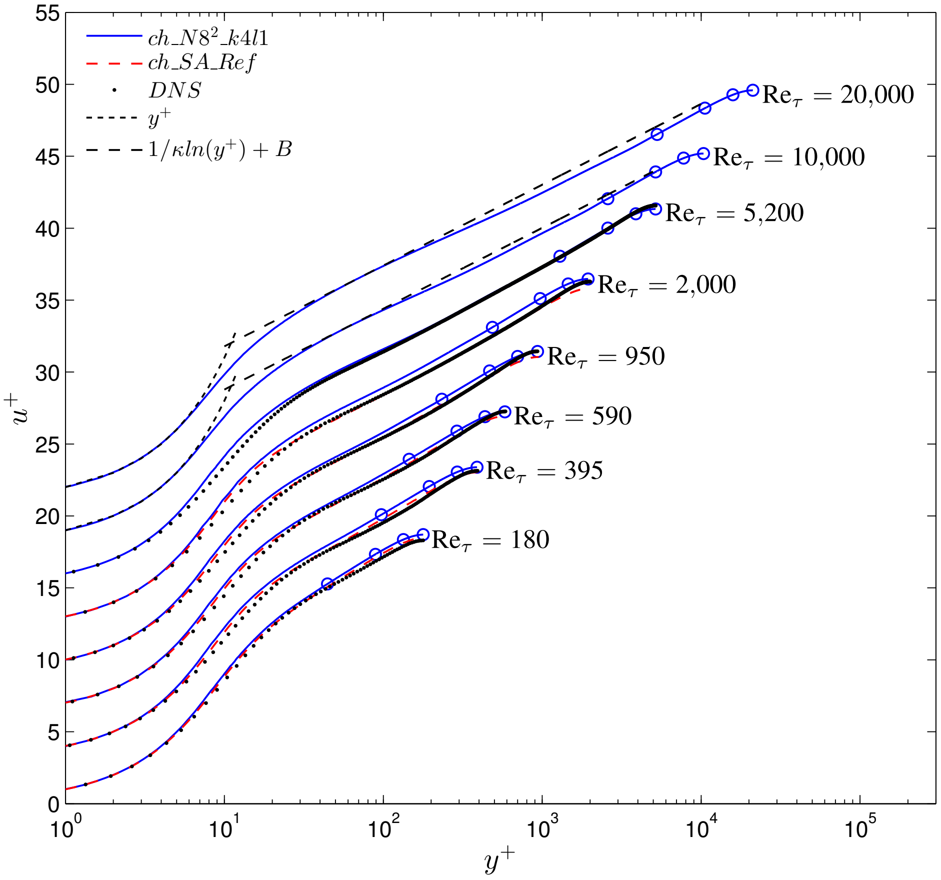


Figure 8.4.: SA-RANS: Velocity solution of turbulent channel flow $u^+ = u/u_\tau$ for $\mathrm{Re}_\tau = 180$, 395, 590, 950, 2,000, 5,200, 10,000, and 20,000, each shifted upwards by three units for clarity. All computations have been carried out with the same mesh displayed in Figure 8.2 consisting of $N = 8^2$ elements of fourth polynomial degree plus first degree for the enrichment within the first off-wall element row. Symbols indicate the location of element interfaces.

cells. The wall model is active only in the first off-wall cells, ranging up to $y^+_{1e} = 99$ in the coarsest case and $y^+_{1e} = 25$ in the finest case. The adaptation algorithm according to Section 8.1 does not have any impact on the present computations, i.e., $y^+_{\max} < 25$. Additionally, the mesh is refined unidirectionally in wall-normal direction using the meshes 8×16 and 8×32 to investigate the influence of varying element aspect ratios. Comparing the results to wall-resolved RANS in Figure 8.5 shows that the results are converged already for the 16×16-case and are not degraded if the wall model is switched off inside the buffer layer as for the finest case 32×32.

In Figure 8.5, we also demonstrate that other polynomial degrees may be used to construct the enrichment ($l = 0$ and $l = 2$) with the cases $ch395\_N8^2\_k4l0$ and $ch395\_N8^2\_k4l2$. Using $l = 0$ implies that the enrichment is multiplied with a constant shape function in each element, yielding a lower degree of flexibility in adapting to the flow configuration. The results depicted in Figure 8.5 exhibit a small underestimation of the velocity. It would in future research nevertheless be interesting to investigate this case with the enrichment function derived through the

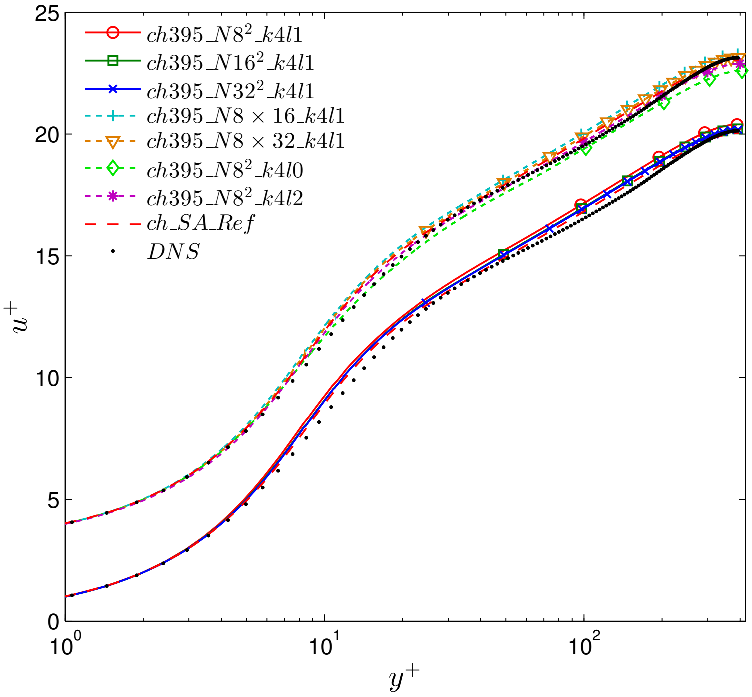


Figure 8.5.: SA-RANS: Velocity solution of turbulent channel flow $u^+ = u/u_\tau$ for $\mathrm{Re}_\tau = 395$ employing eight different discretizations. Bottom: Refinement study using $N = 8^2$, $N = 16^2$, and $N = 32^2$ elements, compared to a wall-resolved RANS reference and DNS. Top: Unidirectional refinement in wall-normal direction and investigation of the cases $l = 0$ and $l = 2$, compared to the same reference data. One single element layer closest to the wall is enriched in all cases. Symbols indicate the location of element interfaces.

SA model as this improvement in consistency is likely to enhance the results, at least in near-equilibrium flows. Regarding the case $l = 2$, the results in Figure 8.5 are in good agreement with the reference computation. The numerical investigations have shown that degrees of $l > 1$ quickly degrade the conditioning of the system, as the shape functions become too similar, and the results are not substantially improved. Therefore, we recommend using $l = 0$ or $l = 1$. In future work, it may also be interesting to consider anisotropic enrichment shape functions, which enable a higher resolution capability in streamwise and spanwise direction and constant or linear shape functions in wall-normal direction in order to circumvent this issue.

Many industrial applications, for example in the automotive, aerospace, or wind energy sector, demand for much higher friction Reynolds numbers. We demonstrate that higher Reynolds numbers may easily be computed with the present wall modeling approach. The friction Reynolds numbers $\mathrm{Re}_\tau = 50{,}000$ and $\mathrm{Re}_\tau = 100{,}000$ are investigated and compared to the linear and log-laws in the respective y^+ regions. Meshes using 16×16 cells are chosen and they are slightly refined towards the wall using the hyperbolic mapping defined in Equation (4.1) improving the resolution of the near-wall area. The values for the mesh stretching parameter γ are selected as 2.0 and 2.25 according to Table 8.1. The results displayed in Figure 8.6 confirm the accuracy of the present method and indicate the suitability for high-Reynolds-number applications.

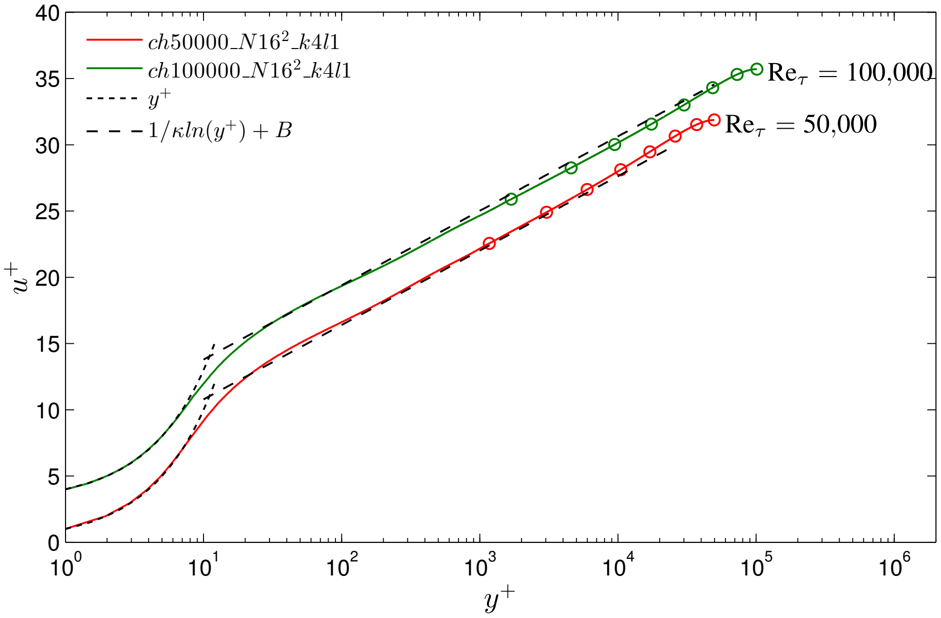


Figure 8.6.: SA-RANS: Velocity solution of turbulent channel flow $u^+ = u/u_\tau$ applied to high Reynolds numbers of $\mathrm{Re}_\tau = 50{,}000$ and $\mathrm{Re}_\tau = 100{,}000$, the latter shifted upwards by three units for clarity. One single element layer closest to the wall is enriched in both cases. Symbols indicate the location of element interfaces.

From this section we draw the conclusion that wall modeling via function enrichment in conjunction with RANS and the high-order DG method allow the prediction of wall-attached flows with coarse meshes where the first cell spans y^+ bandwidths of up to 5,000 wall units. The approach exhibits a high level of grid independence and the solution quality is retained also when increasing the spatial resolution.

8.2.2. Flow over Periodic Hills

As a second numerical example, flow over periodic hills is investigated with the same computational setup as in Section 5.4.2 at $\mathrm{Re}_H = 10{,}595$ and $\mathrm{Re}_H = 19{,}000$. As it was noted earlier, this test case is challenging for common wall modeling approaches since it includes a separation bubble as well as a high adverse pressure gradient, violating the assumptions inherent to equilibrium wall modeling approaches. Wall modeling via function enrichment on the contrary has exhibited promising characteristics in the previous chapter in the context of WMLES.

Three meshes are considered for the discretization of the 2D domain, a coarse one using 16×8 cells, a medium one using 32×16 cells, and a fine one using 64×32 cells, all employing equidistant grid spacings for simplicity. The standard Galerkin component consists of polynomials of degree $k = 4$ and the enrichment is based on polynomials of degree $l = 1$ solely included within the first element layer near the no-slip walls. The additional DOFs of the enrichment increase the overall DOF-count by no more than 4% in the coarse case, 2% in the medium case, and 1% in the fine case. The enrichment is switched off adaptively according to Section 8.1 during the initial transient. In the steady-state solution, only the finest computations are affected by the

Table 8.2.: SA-RANS: Simulation cases and resolutions of the periodic hill benchmark. $\mathrm{Re}_H = 10{,}595$: ph10595_N16×8_$k4l1$ coarse mesh with wall modeling, ph10595_N32×16_$k4l1$ medium mesh with wall modeling, ph10595_N64×32_$k4l1$ fine mesh with wall modeling, ph10595_N32×16_$k4$ medium mesh without wall modeling. $\mathrm{Re}_H = 19{,}000$: ph19000_N16×8_$k4l1$ coarse mesh with wall modeling, ph19000_N32×16_$k4l1$ medium mesh with wall modeling, ph19000_N64×32_$k4l1$ fine mesh with wall modeling, ph19000_N32×16_$k4$ medium mesh without wall modeling. Resolutions are specified in terms of elements per direction N_{ei} and grid points $N_i \approx N_{ei}(k+1)$. The wall-resolved reference cases are the same as in Table 5.2.

case	Re_H	approach	$N_{e1}\times N_{e2}$	$N_1\times N_2\times N_3$	$x_{1,\mathrm{reatt}}/H$
ph10595_N16×8_$k4l1$	10,595	RANS (SA)	16×8	80×40	7.32
ph10595_N32×16_$k4l1$	10,595	RANS (SA)	32×16	160×80	7.59
ph10595_N64×32_$k4l1$	10,595	RANS (SA)	64×32	320×160	7.64
ph10595_N32×16_$k4$	10,595	RANS (SA)	32×16	160×80	7.46
ph10595_SA_Ref	10,595	RANS (SA)	64×32	320×160	7.68
ph19000_N16×8_$k4l1$	19,000	RANS (SA)	16×8	80×40	7.35
ph19000_N32×16_$k4l1$	19,000	RANS (SA)	32×16	160×80	7.57
ph19000_N64×32_$k4l1$	19,000	RANS (SA)	64×32	320×160	7.62
ph19000_N32×16_$k4$	19,000	RANS (SA)	32×16	160×80	7.46
ph19000_SA_Ref	19,000	RANS (SA)	64×32	320×160	7.67

adaptation as the enrichment is switched off in a few cells near the separation and reattachment points at the lower boundary as follows: ph10595_N64×32_$k4l1$: $0.28H - 1.97H$ (separation region), $5.06H - 7.88H$ (reattachment region), ph19000_N64×32_$k4l1$: $0.70H - 1.13H$ (separation region), $7.31H - 7.73H$ (reattachment region). Furthermore, the results are compared to the wall-resolved RANS cases presented in Section 5.4.2 and the medium mesh without wall modeling are considered to assess the impact of the enrichment. All simulation cases including the labels used in the following plots are listed in Table 8.2. The mesh is mapped onto the curved boundary using an isogeometric approach as in Section 4.3. The resulting coarse mesh is displayed in Figure 8.7. With these grids, the first off-wall elements including the enrichment span a range in y^+-units up to approximately $y^+_{1\mathrm{e}} \approx 420$ at the top wall and $y^+_{1\mathrm{e}} \approx 340$ at the hill crest for the coarser mesh as well as the higher Reynolds number. The detailed distribution of $y^+_{1\mathrm{e}}$ for all cases is included in Figure 8.8. The velocity solution is visualized in Figure 8.9. All quantities are postprocessed at simulation time using the full definition of $\boldsymbol{u}_h$ and p_h and the probe locations may also be placed inside cells.

We commence the discussion of the results with the friction and pressure coefficients c_f and c_p at the walls. They are defined as in Equations (4.3) and (4.4) and the reference pressure p_{ref} is taken at $x_1 = 0$ on the upper wall. The results are plotted in Figure 8.10 for $\mathrm{Re}_H = 10{,}595$. Therein, the skin friction computed by the wall-modeled simulation cases converges well to the reference computation. Remaining discrepancies are observed in the vicinity of the hill crest, where the first off-wall cell spans up to 73 wall units in the finest case, which still seems to be slightly too coarse to capture all details accurately. Yet, the tendency to convergence is clear. The

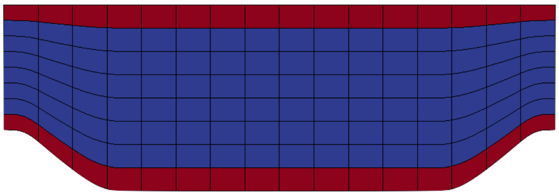


Figure 8.7.: Grid of coarse refinement level. Enriched elements are colored red and standard polynomial elements blue.

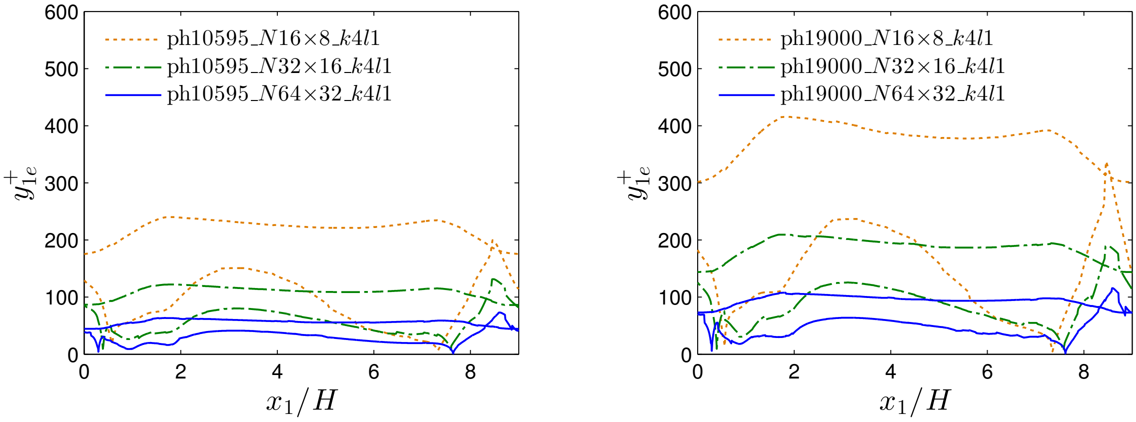


Figure 8.8.: SA-RANS: Location of first off-wall element interface y_{1e}^{+} for flow past periodic hills at $\mathrm{Re}_H = 10{,}595$ (left) and $\mathrm{Re}_H = 19{,}000$ (right). The shallower curves correspond to the upper wall.

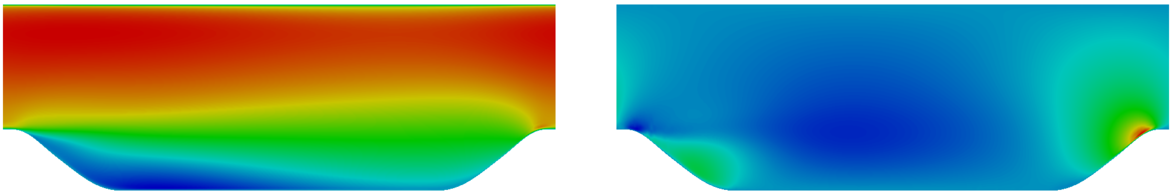


Figure 8.9.: SA-RANS: Streamwise velocity (left) and vertical velocity (right) of the case ph10595_$N32{\times}16$_$k4l1$ at the final simulation time. Red indicates high and blue low values.

largest deviation from the reference is observed for the medium mesh without wall model, for example in the recirculation region and near the hill top. The overall performance of all cases may be summarized by the reattachment lengths $x_{1,\text{reatt}}/H$ included in Table 8.2, which confirm these observations quantitatively. The pressure curves according to Figure 8.10 are less sensitive regarding the wall model and refinement yields almost an ideal match with the reference data. Also, the medium case without wall modeling is of very similar quality to the corresponding case with wall modeling.

Next, the velocity solutions in streamwise and vertical direction u and v are plotted in Figure 8.11 at ten streamwise locations for the same Reynolds number $\mathrm{Re}_H = 10{,}595$. The three computations carried out with the present approach display overall a very good agreement with

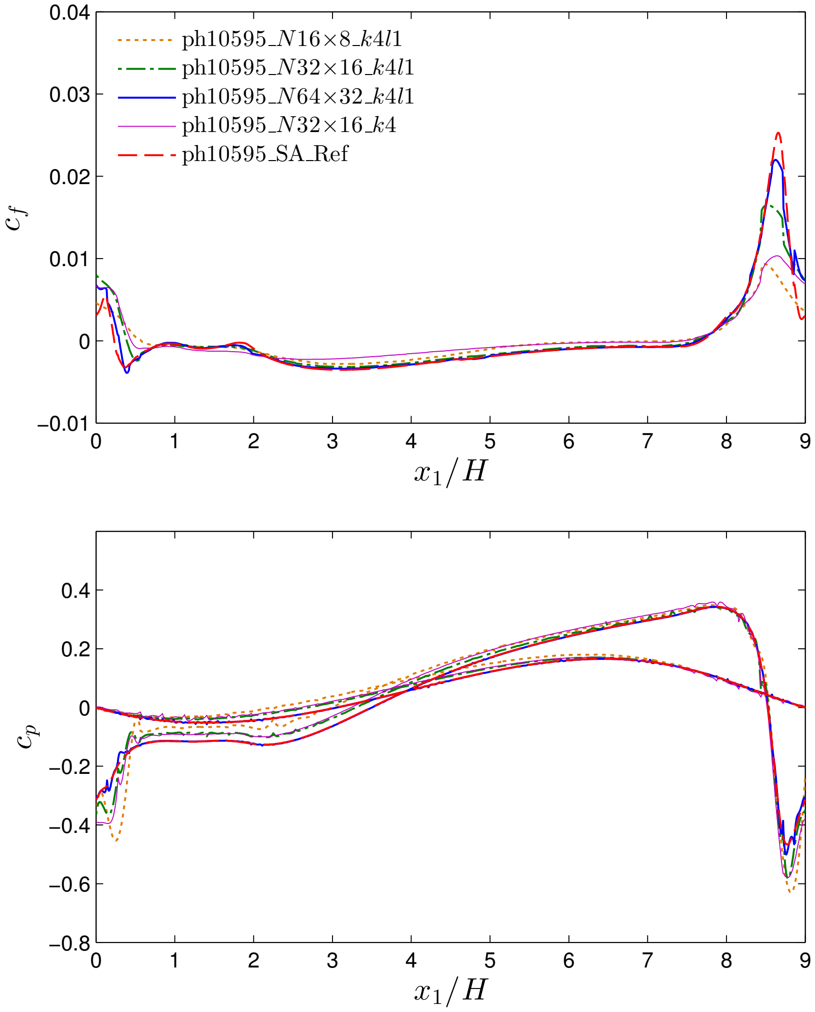


Figure 8.10.: SA-RANS: Friction coefficient (top) and pressure coefficients (bottom) for $\mathrm{Re}_H = 10{,}595$ comparing the wall-modeled simulations with wall-resolved RANS cases. The shallower pressure curves correspond to the upper wall.

the wall-resolved reference. This fact indicates that the wall model is well capable of providing an appropriate function space that yields results almost equal to considerably finer meshes without wall model. Noticeable differences are solely observed in the vertical velocity between $x_1/H = 0$ and $x_1/H = 1$ for the coarse and minor differences for the medium grid. The poorest results are obtained with the medium grid without wall model, where the boundary layer at the top wall is not sufficiently resolved.

The application of the same meshes to the higher Reynolds number of $\mathrm{Re}_H = 19{,}000$ confirms the observations of the lower Reynolds number. The reattachment lengths predicted with the wall-modeled cases included in Table 8.2 show a clear convergence behavior to the reference computation. The velocity solution displayed in Figure 8.12 shows very similar results to the lower Reynolds number case. Refinement leads to small differences between $x_1/H = 0$ and $x_1/H = 1$ and the finest wall-modeled computation is identical to the reference using the SA

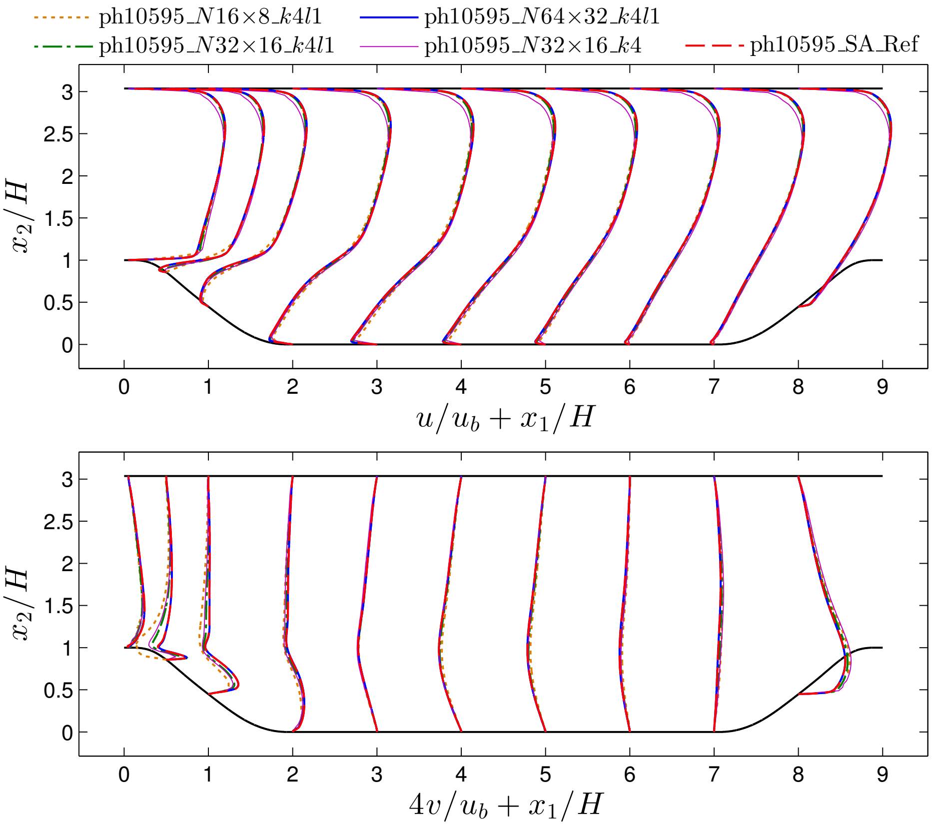


Figure 8.11.: SA-RANS: Streamwise (top) and vertical (bottom) velocity components u and v of flow over periodic hills at $\mathrm{Re}_H = 10{,}595$. Comparison of wall-modeled, wall-resolved, and underresolved simulations.

model. If the wall model is not included as for the case with the medium mesh, the boundary layer at the top wall is not predicted accurately.

In summary, it has been found that the present wall modeling approach provides an appropriate function space even in separated flows, as we have been able to reproduce reference results using the SA model without wall model. A major benefit of the wall model is that refinement does not lead to a degradation of the result quality as opposed to standard wall functions, since the polynomial component of the elements is retained and automatically jumps in if necessary.

8.3. Summary

In this chapter, we have applied wall modeling via function enrichment to RANS and the high-order DG method. This wall modeling approach overcomes the limitations of common wall function models regarding grid dependence and poor performance in separated flows. This is achieved by solving the full Navier–Stokes equations including the SA model with exact wall boundary conditions. The method is capable of employing grids where the first off-wall cell spans a y^+-range of up to $5{,}000$ wall units.

Finally, researchers may be interested in employing the present function-enrichment approach in the context of other RANS turbulence models or flow physics. Such an extension is in gen-

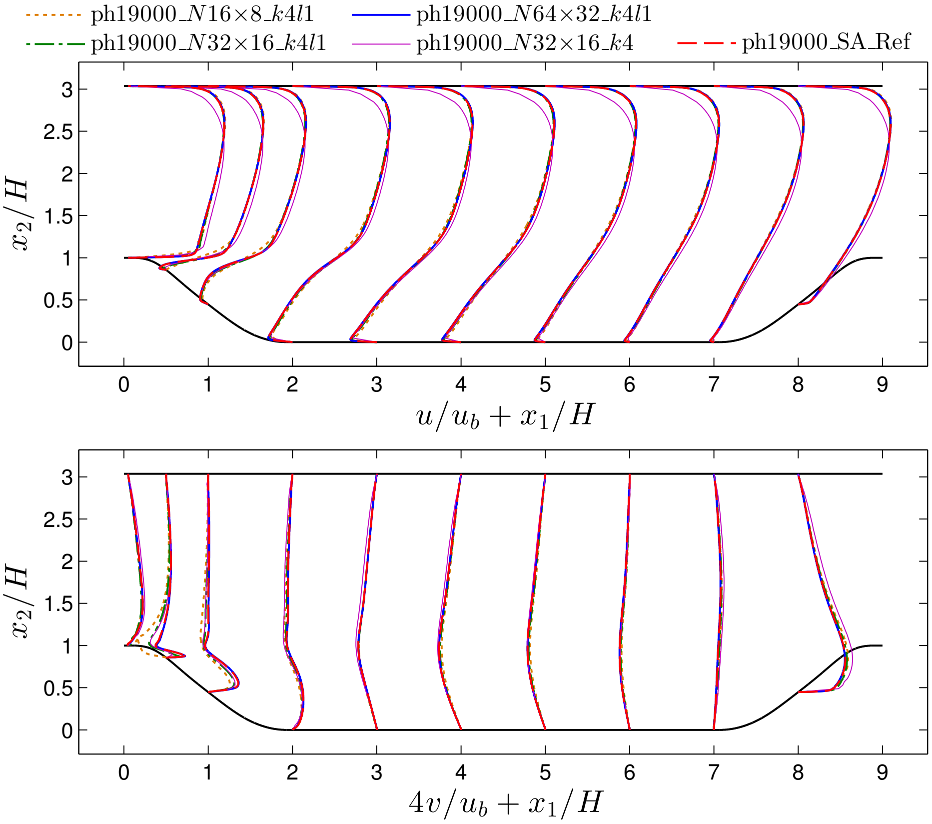


Figure 8.12.: SA-RANS: Streamwise (top) and vertical (bottom) velocity components u and v of flow over periodic hills at $\mathrm{Re}_H = 19{,}000$.

eral possible, if the additional variables vary smoothly within the boundary layer and may be discretized using a polynomial degree of k in the first off-wall cell, see Section 6.3.2.1.

9

Application III: Extension to DES

The first results of wall modeling via function enrichment were presented in Chapter 7 within the continuous FEM as a wall modeling technique for LES. This method allowed the use of coarse cells with a wall-normal width of several hundred wall units in the inner layer. Such a coarse spatial discretization necessitates a statistical turbulence model of RANS-type in that area, since the energetic scales are not resolved. In Chapter 7, a residual-based turbulence model was used, supported by a structural LES model in the outer layer, a model that was originally not intended for underresolved boundary-layer simulations. The numerical examples still showed promising results in separated flows, but the limiting factor in terms of accuracy was the turbulence model.

In this chapter, we show that the widely used DDES methodology [243] (see Section 5.5) may be used to model the unresolved turbulence in the near-wall region in wall modeling via function enrichment. The near-wall region is computed in RANS mode, where the grid filter size is too coarse to resolve the turbulent motions. Further away from the wall, it is affordable to resolve the dominant turbulent scales, so the SA model is modified in such a way that it acts as a one-equation LES subgrid model (see Equations (5.33) to (5.37)). Such a turbulence model is physically motivated and widely recognized in the literature for its accuracy in separated flows. As the DDES model is based on a modification of the SA model, the implementation of the enrichment presented in the previous chapter can be extended in a straightforward way.

Such an implementation of wall modeling via function enrichment has the potential of substantially reducing the computational cost of (D)DES. The grid saving of the standard (D)DES in comparison to LES is achieved by using relatively coarse meshes in the wall-parallel directions of up to 0.1δ (WMLES) and δ (classical DES) with the boundary layer thickness δ (see Section 2.3.4). The wall-normal direction necessitates many grid points in order to resolve the laminar sublayer due to the requirement of placing the first off-wall node at $y_1^+ \sim 1$, however. For example, if a boundary layer of a thickness of 10,000 wall units is computed with a constant grid stretching factor of 1.15 [195], a total of 53 grid layers would be required. This is a quite high cost compared to the relatively low engineering interest in that region. Wall modeling via function enrichment within the approach presented in this chapter allows the first grid point to be located in the range $y_1^+ \sim$ 10 to 100, saving 17–33 grid layers for that example, without noteworthy loss in accuracy, in addition to much better conditioned equation systems through the lower grid anisotropy.

In the next section, we summarize the enrichment formulation used in this chapter. In Section 9.2, the approach is validated and assessed by use of the two familiar benchmark examples of turbulent channel flow and flow over periodic hills. The research presented in this chapter was previously published in Krank et al. [151].

Table 9.1.: SA-DDES: Overview of simulation cases for the turbulent channel flow. The number of polynomial grid points per direction i is $N_i = (k+1)N_{ie}$ with the number of cells per direction N_{ie} and the polynomial degree $k = 4$, Δy^+_{1e} is the thickness of the first off-wall cell, in which the enrichment is active, $y = C_{\text{DES}} h$ is the RANS–LES switching location in terms of channel half-height δ, and err(τ_w) is the relative error of the computed wall shear stress.

Re_τ	$N_{1e} \times N_{2e} \times N_{3e}$	γ	Δy^+_{1e}	$y = C_{\text{DES}} h$	err(τ_w)
395	16×8×8	0.8	76	0.05δ	0.6%
950	16×8×8	1.6	91	0.05δ	4.3%
2,000	16×8×8	1.9	137	0.05δ	−3.3%
	16×16×8	1.9	54	0.05δ	0.1%
	32×16×16	1.9	54	0.025δ	0.4%
5,200	16×16×8	2.2	93	0.05δ	−4.8%
10,000	16×16×8	2.5	116	0.05δ	−2.3%
20,000	16×24×8	2.5	139	0.05δ	0.2%
50,000	16×40×8	2.5	191	0.05δ	0.8%

9.1. Enrichment Space

Analogous to the previous chapters, the function space for the velocity variable is composed of a polynomial and an enrichment component according to Equations (6.1) to (6.5). In this chapter, we employ a weighting of the enrichment function with a constant shape function in each cell ($l = 0$), i.e., with one DOF per space dimension in $\widetilde{\boldsymbol{u}}_0$. This particular choice allows a substantial simplification of the enrichment space as

$$\widetilde{\boldsymbol{u}}_h(\boldsymbol{x}, t) = \psi(\boldsymbol{x}, t)\widetilde{\boldsymbol{u}}_0(t). \tag{9.1}$$

A comparison of such a constant weighting with a linear weighting in each cell ($l = 1$) in the context of WMLES is presented in the subsequent chapter. This analysis shows that the algorithm using linear weighting requires three times more solver iterations in the viscous step than the constant weighting, at the same time as the results are not substantially improved. This result renders the constant weighting most relevant in three space dimensions. Regarding the enrichment function, we use Spalding's law (Equation (6.7)) in this chapter as well, but note that the wall function given through the SA model (Equation (6.9)) would also be very well suited.

In the algorithm, Spalding's law is adapted in each time step according to the local wall shear stress and its temporal evolution is taken into account. To this end, we employ the adaptation algorithm presented in Section 6.3.2.4 with the enhancements discussed in Section 8.1. The latter modification 'switches off' the enrichment in cells where the wall-normal width in y^+-units becomes small, such that the enrichment is not necessary to resolve the gradient and the turbulent length scales. With the WMLES approach presented in Chapter 10, this adaptation method was assessed in detail, and the previously suggested value of $y^+_{\text{max}} = 30$ showed to be suitable for eddy-resolving simulations as well.

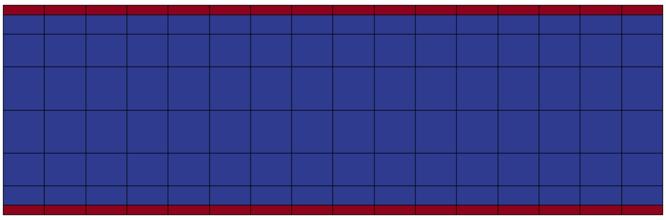


Figure 9.1.: Mesh for turbulent channel flow at $\mathrm{Re}_\tau = 950$. Red indicates enriched cells and blue standard polynomial cells, i.e., a single layer of cells at the wall is enriched. In each cell, the solution consists of a polynomial of 4th degree plus one enrichment shape function in the enriched cells.

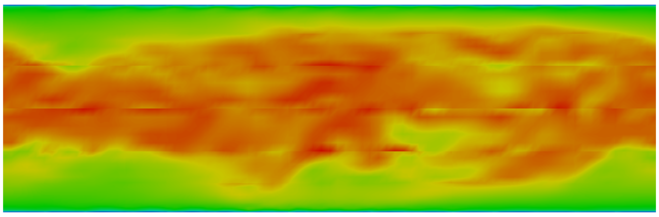


Figure 9.2.: SA-DDES: Instantaneous numerical solution of turbulent channel flow at $\mathrm{Re}_\tau = 950$ via velocity magnitude. Red indicates high and blue low values.

9.2. Numerical Examples

Wall modeling via function enrichment is assessed by considering DDES in the WMLES branch. We employ again the examples of turbulent channel flow and flow over periodic hills as benchmark flows. Analogous to the previous chapter, the polynomial degree of $k = 4$ is used for all simulation cases presented. All simulation cases further use the adaptive time stepping method according to Section 5.3.2 with a temporal accuracy of second order (BDF2), a Courant number of $\mathrm{Cr} = 0.14$, and a diffusion number of $D = 0.02$. The weak forms including the enrichment shape functions are integrated in space using the higher order quadrature rules listed in Table 6.3. The matrix-free inverse mass operator via Schur complement according to Section 6.3.3.2 is used. Finally, for the examples presented in this chapter, we note that we apply no-slip boundary conditions weakly according to Section 3.3.2 in all steps of the scheme. These boundary conditions limit the width of the first off-wall cell to a few hundred wall units, as the no-slip condition would otherwise be violated severely.

9.2.1. Turbulent Channel Flow

We consider flow in a stream- and spanwise periodic channel of the dimensions $2\pi\delta \times 2\delta \times \pi\delta$ in streamwise, wall-normal, and spanwise direction, respectively, with the same setup as in Section 4.2. We investigate this flow in a wide range of friction Reynolds numbers, which are chosen according to the available DNS data at $\mathrm{Re}_\tau = 395$ [187], $\mathrm{Re}_\tau = 950$ [6], $\mathrm{Re}_\tau = 2{,}000$ [116], $\mathrm{Re}_\tau = 5{,}200$ [172], and additionally $\mathrm{Re}_\tau = 10{,}000$, $\mathrm{Re}_\tau = 20{,}000$, and $\mathrm{Re}_\tau = 50{,}000$. All simulation cases, meshes, and resolution criteria are presented in Table 9.1.

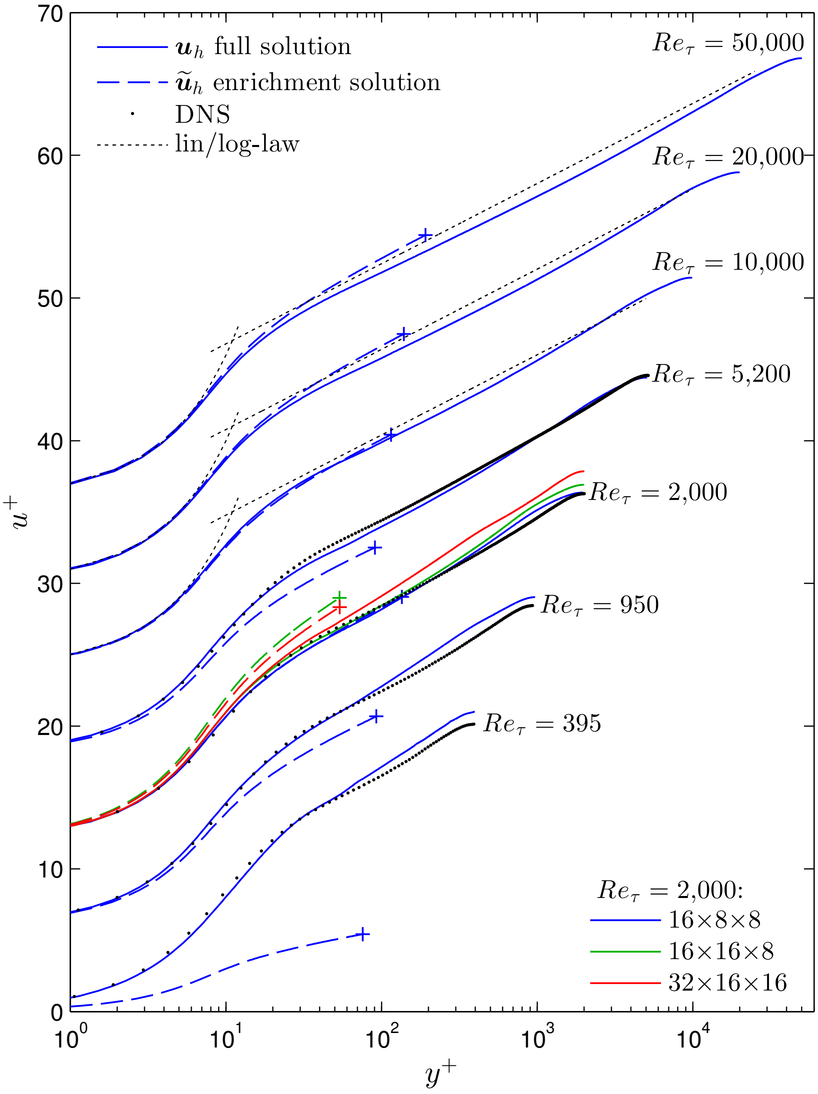


Figure 9.3.: SA-DDES (WMLES) of turbulent channel flow at several Reynolds numbers: Mean velocity normalized according to $u^+ = \langle u_1 \rangle / u_\tau$.

The meshes considered are chosen such that the wall-parallel grid length scale yields approximately $h = 0.08\delta$ for most cases, so the RANS–LES switching point is located at $C_{\text{DES}} h \approx 0.05\delta$. One simulation case uses twice the number of grid cells in streamwise and spanwise direction, resulting in a RANS–LES switching point at $C_{\text{DES}}\Delta \approx 0.025\delta$. As for the wall-normal resolution, the enrichment is taken into account in the wall-nearest cell layer in all simulation cases, see Figure 9.1. As it was discussed earlier, the enrichment shape functions allow the resolution of the averaged near-wall flow with very coarse cell sizes. The width of the first off-wall cell lies in the range of 51 to 191 wall units in this section. In order to enable an application to high Reynolds numbers, a hyperbolic grid stretching is additionally considered, according to Equation (4.1), with the mesh stretching parameter γ. The values of γ for all simulation cases are included in Table 9.1. In the numerical method, the velocity solution is

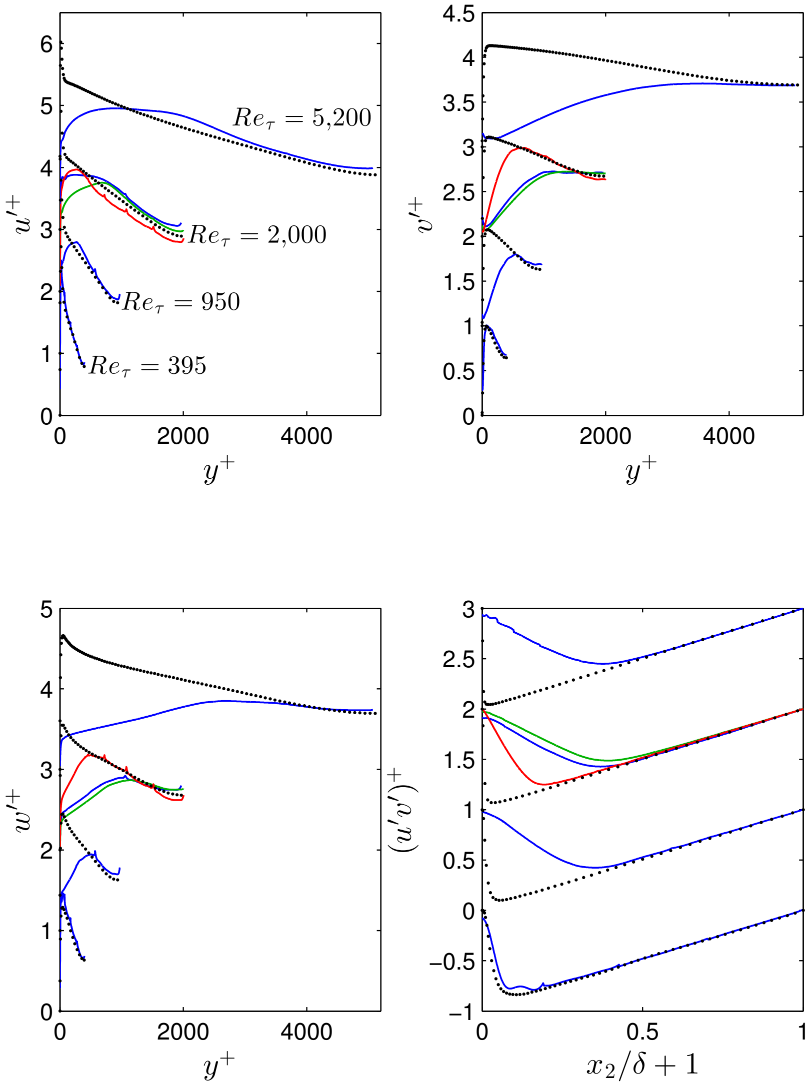


Figure 9.4.: SA-DDES (WMLES) of turbulent channel flow at several Reynolds numbers: RMS-velocities and RSS. All quantities are normalized according to $u'^+ = \sqrt{\langle u_1'^2 \rangle}/u_\tau$, $v'^+ = \sqrt{\langle u_2^2 \rangle}/u_\tau$, $w'^+ = \sqrt{\langle u_3^2 \rangle}/u_\tau$, and $(u'v')^+ = \langle u_1 u_2 \rangle / u_\tau^2$.

postprocessed at sufficient wall-parallel layers inside each cell using the definition of the velocity variable according to Equation (6.1) such that the behavior of the enrichment may be analyzed.

The turbulent flow is visualized at one time instant in Figure 9.2. Time-averaged results are presented in Figures 9.3 and 9.4. The mean velocity is generally predicted very accurately in the laminar sublayer and the log-layer, where the enrichment shape functions are active. In order to get a better impression of the role of the enrichment, the numerical enrichment solution is plotted in Figure 9.3 alongside the full mean velocity solution. The enrichment solution represents the largest part of the near-wall solution in most cases, including the high velocity gradient. In particular in cases, where the first off-wall cell spans a range of more than 100 wall units, the enrichment represents most of the mean velocity. Solely at the lowest Reynolds number, the en-

Table 9.2.: SA-DDES: Simulation cases and resolutions of the periodic hill flow. The cases use a coarse mesh with $32\times16\times16$ grid cells and a fine mesh with $64\times32\times32$ elements. The polynomial degree is $k = 4$ for all simulation cases, and the number of grid points per direction is $k + 1$ in each cell. The separation and reattachment lengths $x_{1,\mathrm{sep}}$ and $x_{1,\mathrm{reatt}}$ correspond to the zero-crossings of the skin friction.

Case	$N_{e1}\times N_{e2}\times N_{e3}$	$N_1\times N_2\times N_3$	Re_H	$\max(\Delta y^+_{1e})$	$x_{1,\mathrm{sep}}/H$	$x_{1,\mathrm{reatt}}/H$
ph10595_coarse	$32\times16\times16$	$160\times80\times80$	10,595	75	0.25	4.58
ph10595_fine	$64\times32\times32$	$320\times160\times160$	10,595	36	0.17	4.62
DNS 10595	-	$896\times448\times448$	10,595	-	0.19	4.51
ph37000_coarse	$32\times16\times16$	$160\times80\times80$	37,000	143	0.40	3.34
ph37000_fine	$64\times32\times32$	$320\times160\times160$	37,000	78	0.26	4.56
RM_Exp [211]	-	-	37,000	-	-	3.76
CM_WMLES_coarse	-	$128\times64\times64$	37,000	-	-	2.3
CM_WMLES_fine [265]	-	$256\times128\times128$	37,000	-	-	2.8

richment solution plays a minor role, which essentially means that the polynomial component is capable of resolving most of the flow. Further away from the wall we observe the characteristic log-layer mismatch, that we expect in wall-attached simulations using DDES [195, 268] (see Section 6.2.1). The log-layer mismatch is especially visible for the lower Reynolds numbers. We note that there are several techniques available in the literature that reduce this effect, for example [237]. In the framework of the present enrichment methodology, it is possible to construct an alternative hybrid RANS/LES turbulence model, which does not show a log-layer mismatch by definition. This approach is the topic of the subsequent chapter.

The RMS velocities and the RSS are presented in Figure 9.4 up to $\mathrm{Re}_\tau = 5{,}200$ and compared with the DNS data. These quantities show that the RANS–LES transition extends up to approximately 0.4δ and the flow is in full LES mode further away from the wall. This means that we do not expect agreement with the DNS below 0.4δ, and the curves match the DNS above this value very well. Only in the refined case at $\mathrm{Re}_\tau = 2{,}000$ and at the lowest Reynolds number, the RANS–LES transition takes place closer to the wall.

Finally, a major advantage of the present method is the accurate prediction of the wall shear stress despite of the coarse grids. In Table 9.1, we list the relative error of the computed wall shear stress compared to the nominal simulation parameters for each simulation case. The error lies within a few percent for all cases. Comparing the values with the errors in the skin friction coefficient presented in [195] of up to 22%, this is an excellent result.

We conclude from this section that wall modeling via function enrichment allows an accurate computation of the near-wall region in turbulent boundary layers with very coarse cells, while still satisfying the full incompressible Navier–Stokes equations in the whole boundary layer. DDES is a suitable turbulence modeling approach for wall modeling via function enrichment.

9.2.2. Flow over Periodic Hills

As a second benchmark example, we consider flow over periodic hills at the Reynolds numbers of $\mathrm{Re}_H = 10{,}595$ and $\mathrm{Re}_H = 37{,}000$. Several hybrid RANS/LES methods were assessed us-

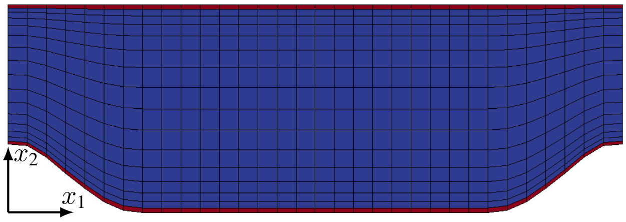


Figure 9.5.: Mesh for flow over periodic hills of the case ph37000_coarse. Red indicates enriched cells and blue standard polynomial cells, i.e., a single layer of cells at the wall is enriched. In each cell, the solution consists of a polynomial of 4th degree plus one enrichment shape function in the enriched cells.

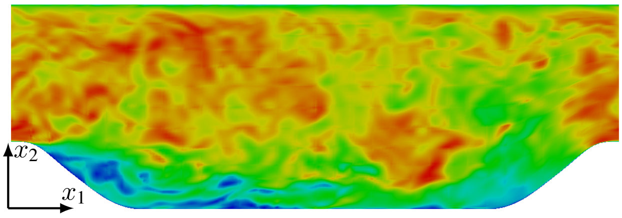


Figure 9.6.: SA-DDES: Instantaneous numerical solution of flow over periodic hills of the case ph37000_coarse via velocity magnitude. Red indicates high and blue low values.

ing this flow configuration within the European ATAAC initiative [232], including DDES (see the final report by Jakirlić for cross-comparison of results). A strong adverse pressure gradient and flow separation from the curved boundary are challenging for many statistical modeling approaches, but DDES yielded very good agreement with a reference LES in that study.

The computational setup is analogous to Section 4.3, except for the details specified in the following. Two meshes are considered at each Reynolds number, a coarser mesh with 32×16×16 cells, and a finer one with 64×32×32 cells. As for the previous example, the solution is represented by a polynomial of degree $k = 4$ in each cell, plus one enrichment shape function in the wall-nearest cell layer. The mesh is moderately stretched towards the no slip walls to yield a better resolution of the near-wall area. One representative mesh is displayed in Figure 9.5. The wall-normal width of the enrichment layer is plotted in Figure 9.7 in wall coordinates. An overview of all simulation cases and resolution parameters is given in Table 9.2. One snapshot of the instantaneous velocity field is visualized in Figure 9.6.

We begin the discussion of the results with the skin friction and pressure coefficients c_f and c_p. They are defined according to Equations (4.3) and (4.4) where the reference pressure p_{ref} is taken at $x_1 = 0$ at the upper wall. The results of the lower Reynolds number are compared to the DNS in Figure 9.8. All profiles yield very good agreement with the DNS. Solely the skin friction coefficient predicted by the coarse mesh shows an overprediction of the magnitude between $x_1/H = 2$ and $x_1/H = 4$. Even the characteristic peak in the skin friction on the windward side of the hill crest is predicted very well for both cases. The overall excellent agreement is also observed in the estimation of the length of the reattachment zone of $x_{1,\text{reatt}}/H = 4.58$ and 4.62 (see Table 9.2) in comparison to the DNS result of $x_{1,\text{reatt}}/H = 4.51 \pm 0.06$. We also note that the 'waviness' of the skin friction coefficient is much less pronounced than in Chapter 4.3.

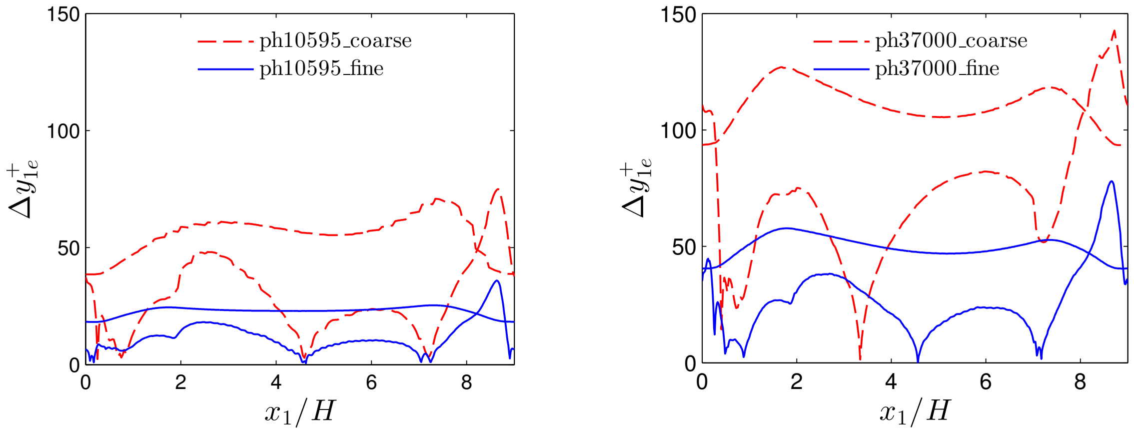


Figure 9.7.: SA-DDES: Width of wall-layer (width of first off-wall cell) for $\mathrm{Re}_H = 10{,}595$ (top) and $\mathrm{Re}_H = 37{,}000$ (bottom). The shallower curves correspond to the upper wall.

The velocity profiles of the same Reynolds number are compared to the DNS data at ten streamwise locations in Figure 9.9. The streamwise velocity agrees exceptionally well with the reference DNS. The vertical velocity shows a minor difference at $x_1/H = 2$ for the coarser simulation case, but the remaining profiles essentially lie on the DNS curves. A similar level of accuracy is observed in the RSS distribution. The TKE computed with the coarse simulation case shows an underprediction of the magnitude in the shear layer. These results also exhibit unphysical peaks in the shear layer, which are typical for high-order DG, since the discontinuity present in the velocity yields higher fluctuations near the element boundaries, see also [149].

The excellent results obtained at the lower Reynolds number motivate an application of the wall model to a substantially higher Reynolds number. The velocity statistics are compared to the available experimental reference data at $\mathrm{Re}_H = 37{,}000$ in Figure 9.10. In order to allow for a critical assessment of the present wall modeling approach, we additionally compare the results of the mean streamwise velocity with a recent implementation of an equilibrium wall model within the high-order DG [265] (cases baseline and fine in that publication). These simulations employ grids comparable to the respective coarse and fine case presented in this work and are also included in the overview if Table 9.2. Regarding the mean velocity, all wall-modeled cases yield larger errors as compared to the lower Reynolds number. The equilibrium wall model overpredicts the velocity in the recirculation zone, yielding a shorter reattachment length of $x_{1,\mathrm{reatt}}/H = 2.3$ and $x_{1,\mathrm{reatt}}/H = 2.8$ in comparison to the experiments ($x_{1,\mathrm{reatt}}/H = 3.76$, see Table 9.2). The present wall-enriched DDES simulations overpredict the mean streamwise velocity in that region with the coarse mesh and underpredict the velocity in the fine case. Yet, the DDES cases are closer to the reference than the equilibrium model, both for the coarse and fine mesh. The reattachment lengths are computed as $x_{1,\mathrm{reatt}}/H = 3.34$ and 4.56 and confirm the observations of the mean velocity. The profiles of the vertical velocity yield differences with the reference on the lee side of the hill as a result of the different length of the separation bubble. The magnitude of the RSS is overpredicted by the coarse case and is accurately estimated by the fine case.

We conclude from the results of the periodic hill flow that wall modeling via function enrichment with DDES as turbulence model is well capable of computing nonequilibrium flows.

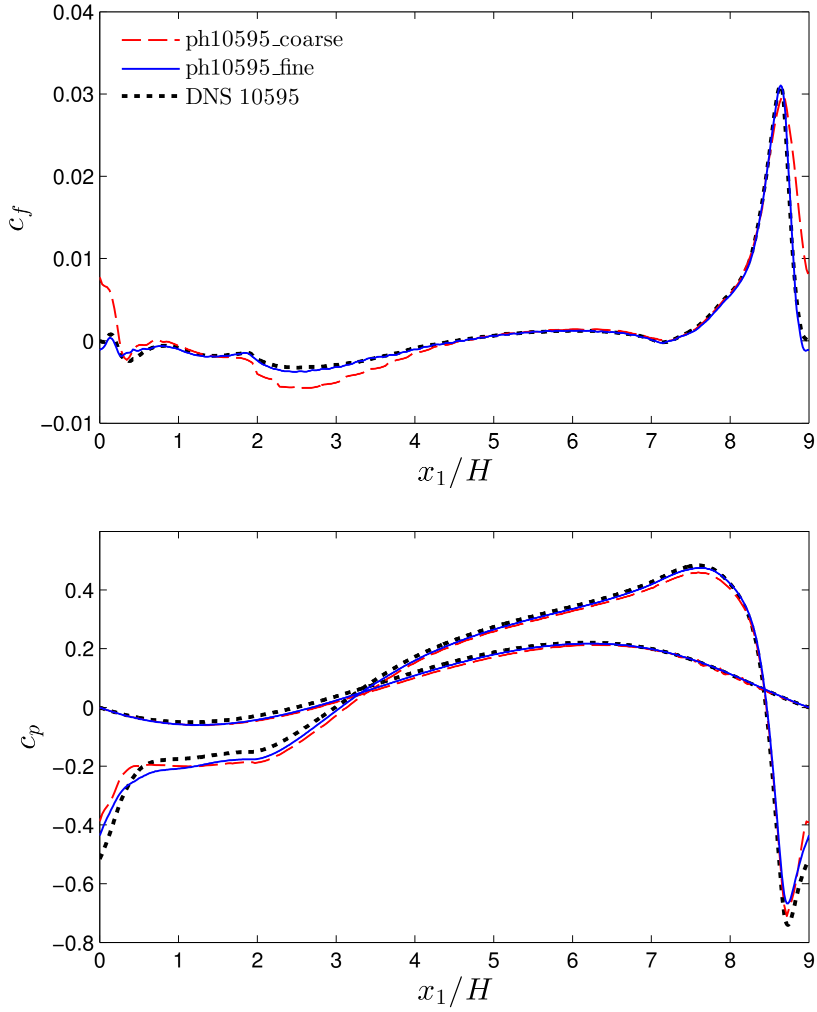


Figure 9.8.: SA-DDES: Skin friction coefficient at the lower wall (left) and pressure coefficient at the lower and upper boundary (right). The shallower pressure coefficient curves correspond to the upper wall.

This is due to the full consistency of the method, as all terms of the Navier–Stokes equations are satisfied discretely.

9.3. Summary

In this chapter, we have used the DDES methodology to model the unresolved turbulent motions in wall modeling via function enrichment. The results of turbulent channel flow and flow over periodic hills are in very good agreement with the reference data.

Wall modeling via function enrichment with the DDES turbulence model does not provide a solution for the problems in the hybrid RANS–LES transition region in attached boundary layers. However, an alternative hybrid RANS/LES turbulence modeling approach can be constructed

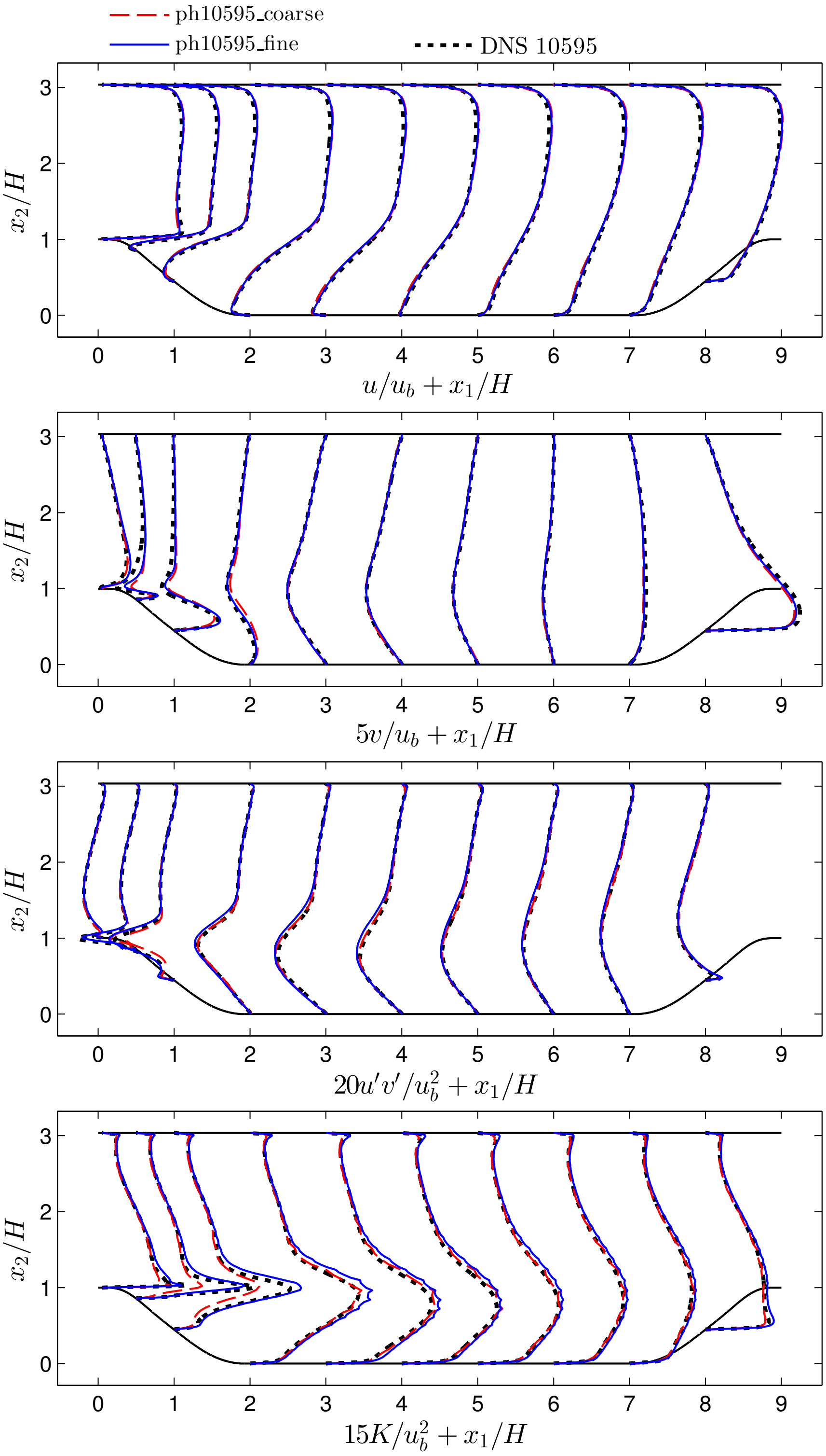


Figure 9.9.: SA-DDES: Streamwise $u = \langle u_1 \rangle$ and vertical $v = \langle u_2 \rangle$ mean velocity, RSS $u'v' = \langle u_1 u_2 \rangle - \langle u_1 \rangle \langle u_2 \rangle$, and TKE $K = 1/2(u'u' + v'v' + w'w')$ of the periodic hill flow at $\mathrm{Re}_H = 10{,}595$.

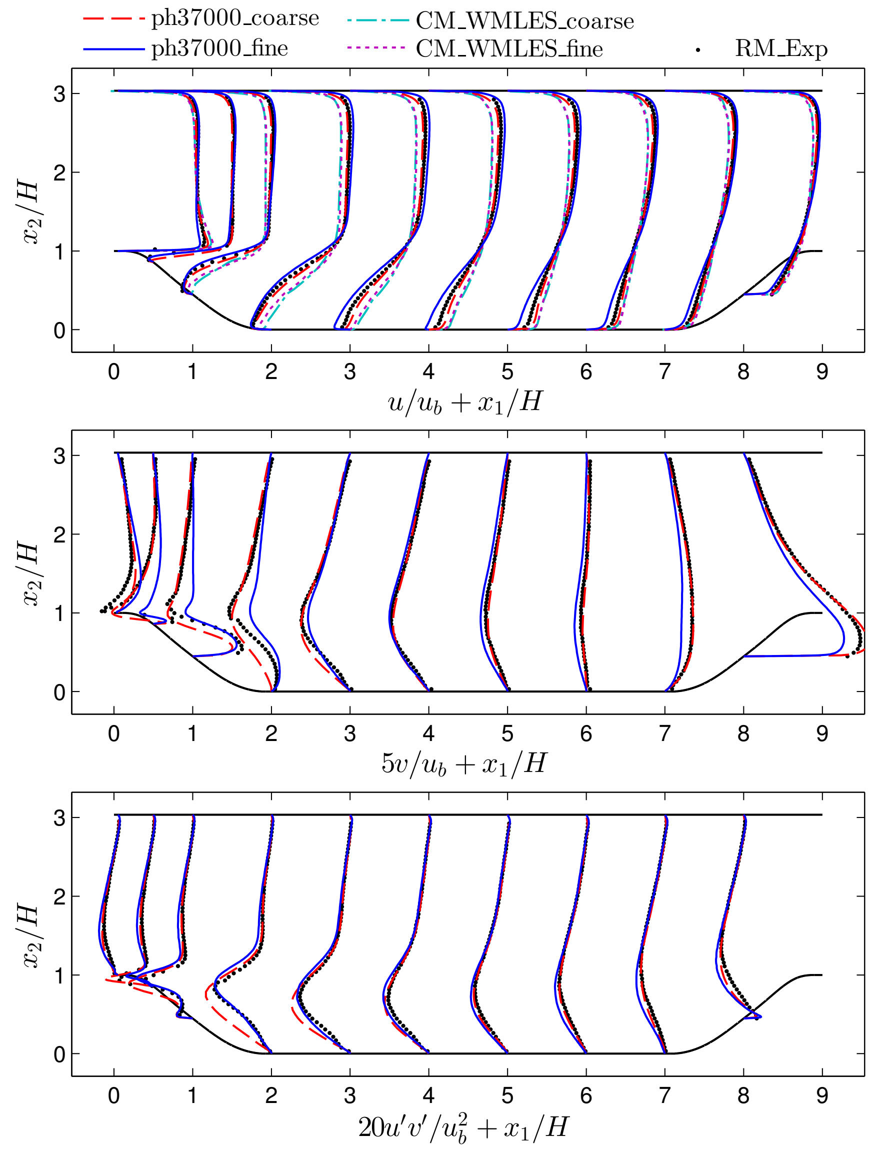


Figure 9.10.: SA-DDES: Streamwise $u = \langle u_1 \rangle$ and vertical $v = \langle u_2 \rangle$ mean velocity as well as RSS $u'v' = \langle u_1 u_2 \rangle - \langle u_1 \rangle \langle u_2 \rangle$ of the periodic hill flow at $\mathrm{Re}_H = 37{,}000$. The results of the cases CM_WMLES_coarse and CM_WMLES_fine are only available for the streamwise velocity.

based on the enrichment, which *a priori* circumvents these problems and the associated log-layer mismatch. This turbulence model is described in the next chapter.

10

Application IV: A Multiscale Approach to Hybrid RANS/LES Wall Modeling

The previous chapter showed that a hybrid RANS/LES turbulence model is a consistent approach to model the unresolved turbulent motions in the context of wall modeling via function enrichment. However, the DES model used in that chapter did not address the issue of the hybrid RANS–LES transition. In DES, all turbulent motions are modeled in the near-wall region, where the eddy viscosity model operates in RANS mode. Further away from the wall, the energetic turbulent motions are resolved and they are responsible for most of the momentum transfer. In such an approach, the problem is therefore the transition region from the RANS to the LES zone, where the turbulence is not sufficiently developed to account for the momentum transfer but the model does not operate in RANS mode either. This problem results in the log-layer mismatch discussed earlier in Section 6.2.1, which is also visible in the DES results in Chapters 5 and 8.

In this chapter, a new and fully consistent wall modeling approach is developed, which makes use of the structure of the shape functions in wall modeling via function enrichment and resolves the problem of the RANS–LES transition.

The remainder of this chapter is organized as follows. In the first subsection, we outline the basic idea of the wall modeling approach and discuss similarities and differences in comparison to other hybrid RANS/LES methods. In Section 10.2, we derive consistent governing equations for the RANS and LES components and show how these equations may be solved in the framework of the variational multiscale method. In Section 10.3, the function space used for the velocity variables is revisited. Section 10.4 describes how the multiscale wall model can be implemented in the present high-order DG scheme. In Section 10.5, numerical examples demonstrate the excellent mesh independence of the present wall model and show how the full consistency of the model enables accurate results in separated flow conditions. The multiscale wall model was previously described in Krank et al. [150] and the present chapter is closely related to that article.

10.1. Basic Idea and Comparison to Existing Hybrid RANS/LES Methods

10.1.1. Basic Idea

The two challenges in computing turbulent boundary layers were introduced in Section 6.1 as (i) the high spatial gradient and (ii) the multiscale nature of the turbulent motions. In this chapter, we propose a novel approach to hybrid RANS/LES wall modeling, which resolves the velocity gradient at the wall with relatively coarse meshes (cf. (i)), models the near-wall turbulence (cf. (ii)), and additionally solves the problem of the RANS–LES transition. The basic idea is that, in the vicinity of the wall, the solution is decomposed into a RANS and an LES component. The two components overlap in this near-wall layer, and together they represent the velocity solution. The decoupling is achieved by applying the RANS eddy viscosity on the RANS DOFs only which allows the energy-carrying turbulent structures to fully develop inside the near-wall layer, which entirely circumvents the aforementioned problem of transition at the RANS/LES interface. This structure of the eddy viscosity term is a direct result of applying a new three-level hybrid RANS/LES filter, which is defined based on Germano's framework of additive filters [97]. Despite the fact that the two components are expressed and modeled separately, the full Navier–Stokes equations are computed in the whole boundary layer in a single equation, with exact boundary conditions. This ensures robustness in challenging flow conditions such as separated flows featuring a high adverse pressure gradient. The velocity gradient is resolved, although the first off-wall node for the LES may be placed in a range $y_1^+ \sim 1 \ldots 100$, by using wall modeling via function enrichment.

Once the solution is decomposed into a RANS and an LES component, each with different shape functions, we employ a feature of the Galerkin method: The weighting functions have the same structure as the solution functions, allowing an explicit decomposition of the governing equations into separate equations for the RANS and the LES components – although the problem is still solved in one single equation – according to the variational multiscale methodology [117, 118]. The scale separation allows appropriate turbulence modeling for each of the scales, a RANS turbulence model for the RANS equation and an LES subgrid model for the LES scale, exactly as they are derived through the three-level hybrid RANS/LES filter. We note that this is a fundamental difference to the application of function enrichment in DES according to the previous chapter, where the domain is clearly separated into the inner RANS region and the outer LES region, with a 'grey area' in between, since the RANS eddy viscosity acts on the polynomial velocity component as well in that study; these differences are detailed in the subsequent section.

10.1.2. Comparison to Other Hybrid RANS/LES Methodologies

The composition of the solution of a RANS and an LES component relates the present method to the nonlinear disturbance equations (NLDE) [164, 165, 186], which have been proposed in order to reconstruct the energy-carrying turbulent structures (LES) around an averaged mean flow (RANS), with the primary application in the field of aeroacoustics. In the NLDE, the RANS and LES components overlap in a part of the domain, as they do in the present method. Several features of our approach distinguish between the methods:

- In NLDE, the RANS equations are first solved for the mean flow and the turbulent fluctuations are reconstructed in a subsequent step, whereas the RANS and LES solutions are computed simultaneously in a single equation in the present model.
- In NLDE, the LES represents fluctuations around a mean flow, while the LES and RANS are each a variable fraction of one solution in the present methodology.
- The NLDE usually require the LES to be wall-resolved, whereas in our model the RANS provides the LES with the necessary dissipation if the LES is underresolved, i.e., not all energetic scales have to be computed in the LES.

The multiscale wall model is further compared to two of the classical hybrid RANS/LES wall models, the original DES approach by Spalart et al. [244], which was applied in the DDES variant in the previous chapter, and the method of blending subgrid models through a weighted sum by Baggett [11]. Both models follow the idea of two spatially separated zones, a RANS layer and an LES bulk flow, with a transition region in between. They compare to the proposed method as follows:

- DES achieves the blending by limiting the wall distance function in the SA model by a characteristic grid length scale such that the RANS model acts as a one-equation LES model in the bulk flow, see, e.g., [267] for an application with DG. The present method applies the idea of limiting the wall distance of the RANS model by a characteristic cell length scale in order to account for the resolved Reynolds stresses.

- The method of weighted sum blends the subgrid stress tensor from a RANS model at the wall into an LES model in the bulk flow via an explicitly defined blending function. This approach has been refined by Germano [97] and Rajamani and Kim [208], who suggested to blend LES and RANS filtering operators instead of subgrid models, and it is this filter that will be extended to a three-level filter in the present work.

- The major difference of the proposed approach to both the DES and the weighted sum method is, however, that a marginally/underresolved LES, which is not directly affected by the RANS eddy viscosity, extends to the no-slip boundary. It is this difference that avoids nonphysical velocity fluctuations, such as described by Baggett [11], within the present method. A direct qualitative and quantitative comparison of DES using function enrichment according th the preceding chapter and the present multiscale wall model will be presented in Section 10.5.1.

10.2. A Multiscale Approach to Wall Modeling

The primary contribution of this chapter is the derivation of a consistent multiscale RANS/LES wall modeling framework for wall modeling via function enrichment. In the context of wall modeling via function enrichment, the Galerkin method enables a much more elegant turbulence model than the DDES model used in the previous chapter; the derivation of this model is the topic of this section.

In the following, the solution of the incompressible Navier–Stokes equations in a layer near the wall is split into two parts, an eddy-resolving component (LES) and an ensemble-averaged component (RANS). The equations are derived by applying a new three-level filter to the Navier–Stokes equations, which is defined based on Germano's framework for additive filtering. The implications of this filter are that both the RANS eddy viscosity term and the definition of the eddy viscosity variable depend only on the RANS component, thus the LES component can evolve independently and is limited solely by its own resolution. Further, the RANS and the LES components are not spatially separated, but overlap in the near-wall layer. This construction yields a method that does not show the typical problems in the RANS–LES transition region and the associated log-layer mismatch.

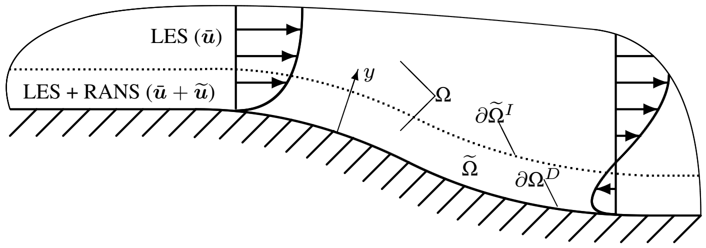


Figure 10.1.: Zoom-in on the near-wall area. The full domain is denoted Ω and a subset in a layer near the wall $\widetilde{\Omega} \subset \Omega$. In $\widetilde{\Omega}$, the solution is composed of a RANS and an LES component, both in variable composition, while the solution outside this area is represented by an LES component only. The interface in between is denoted $\partial\widetilde{\Omega}^I$.

In this section, we begin with the introduction of the scale separation by a three-level filter and subsequently discuss the differences to the classical hybrid RANS/LES filter in Section 10.2.1. Section 10.2.2 shows how the variational multiscale method is employed to solve for the RANS and LES solutions using weak forms and appropriate solution as well as weighting functions.

10.2.1. Three-Level Hybrid RANS/LES Filter

We propose a novel hybrid RANS/LES filter based on Germano's framework of additive filtering [97]. The filter is constructed such that an underresolved LES solution overlaps with the RANS solution in the near-wall area. In this subsection, the classical two-level additive hybrid RANS/LES filter [97] is first recited and that framework is subsequently used to define a new three-level filter. In applying the new filter to the incompressible Navier–Stokes equations, we obtain a set of equations for an eddy-resolving velocity variable and a statistical velocity variable, and their sum represents the final velocity solution. These equations have formally seven unknowns, three LES velocity components, three RANS velocity components, and one pressure variable. In practice, the LES and RANS components will be solved by taking a different set of shape functions tailored for each of the two scales (see Sections 10.2.2 and 10.3).

The derivations address the near-wall area of a computational domain. Figure 10.1 gives an overview of the general concept and the variables defining the wall model. The near-wall region is denoted $\widetilde{\Omega} \subset \Omega$ analogous to the preceding chapters; the baseline LES solver is used outside this area. Coupling conditions defining the transition from the near-wall region to the outer region at the interface $\partial\widetilde{\Omega}^I$ are also discussed in this section.

10.2.1.1. The Classical Two-Level Additive Hybrid RANS/LES Filter

Germano's hybrid RANS/LES filter [97] defines the velocity and pressure as a weighted sum of the LES-filtered and RANS-averaged quantities, yielding

$$\boldsymbol{u}_H = a_1 \bar{\boldsymbol{u}} + a_2 \langle \boldsymbol{u} \rangle, \tag{10.1}$$

$$p_H = a_1 \bar{p} + a_2 \langle p \rangle. \tag{10.2}$$

Herein, $\overline{(\cdot)}$ denotes an LES-filtering operator and $\langle \cdot \rangle$ a statistical (ensemble-averaging) operator. As an LES filter, implicit grid filtering is commonly considered, which may also be interpreted as a variational projection in the context of the Galerkin method [120]. The continuous blending functions $0 \leq a_1 \leq 1$ and $a_2 = 1 - a_1$ may be spatially dependent and are predefined. The filter fulfills two basic requirements, namely

$$a_1 + a_2 = 1, \tag{10.3}$$

$$\langle \boldsymbol{u}_H \rangle = \langle \boldsymbol{u} \rangle, \tag{10.4}$$

and equivalently for the pressure. Applying this filter to the governing equations (2.2) and (2.1) and using the rule of permutation of the derivative and the filtering operator [97], we obtain

$$\nabla \cdot \boldsymbol{u}_H = 0, \tag{10.5}$$

$$\frac{\partial \boldsymbol{u}_H}{\partial t} + \nabla \cdot \left(\boldsymbol{\mathcal{F}}^c \left(\boldsymbol{u}_H \right) + p_H \boldsymbol{\mathcal{I}} - \boldsymbol{\mathcal{F}}^\nu \left(\boldsymbol{u}_H \right) \right) = \boldsymbol{f} - \nabla \cdot \boldsymbol{\tau}_H, \tag{10.6}$$

where $\boldsymbol{\tau}_H$ is the corresponding subgrid stress tensor, which is defined in [97]. It is noted that the chain rule applies in all spatial derivatives if a_1 is spatially dependent.

It is argued that the construction of this filter promotes some of the issues associated to hybrid RANS/LES methods, such as in the transition region. For any blending factor $a_1 < 1$, the resolved LES scales in $\boldsymbol{u}_H$ are not allowed to evolve with their full magnitude as they are multiplied by a factor smaller than one, which induces a damping of the resolved turbulence. Therefore, the smallest resolved scales would be larger than the given grid filter size and the method does not exhibit optimal efficiency in resolving turbulent scales. It is one of the primary incentives of this work to construct a numerical method which allows the LES scales to evolve independently, only limited by the coarseness of the LES filter size. This is achieved by proposing a new formulation of the hybrid RANS/LES filter in the following, which allows the use of $a_1 = 1$ throughout.

10.2.1.2. A New Three-Level Additive Hybrid RANS/LES Filter

We modify the classical additive hybrid RANS/LES filter by adding a third term, yielding

$$\boldsymbol{u}_H = a_1 \bar{\boldsymbol{u}} + a_2 \langle \boldsymbol{u} \rangle + a_3 \langle \bar{\boldsymbol{u}} \rangle, \tag{10.7}$$

$$p_H = a_1 \bar{p} + a_2 \langle p \rangle + a_3 \langle \bar{p} \rangle, \tag{10.8}$$

for the filtered velocity and pressure. Herein, the third filtering operator $\langle \overline{(\cdot)} \rangle$ defines a hierarchical filter by successively applying an LES filter and a statistical filter. This particular construction enables the choice of the constant factors $a_1 = 1$, $a_2 = 1$, and $a_3 = -1$ throughout the near-wall region. The third component subtracts the statistical average of the resolved variables from the balance. Due to $a_1 = 1$, resolved turbulent structures are not damped artificially by the method. The eddy resolving component $\bar{\boldsymbol{u}}$ may be significantly underresolved as we place the first off-wall node in the region of $y_1^+ \sim 1 \ldots 100$, so the energy-carrying scales do not have to be resolved in the near-wall area. The resulting filtering operator fulfills the requirements:

$$a_1 + a_2 + a_3 = 1, \tag{10.9}$$

$$\langle \boldsymbol{u}_H \rangle = \langle \boldsymbol{u} \rangle, \tag{10.10}$$

and equivalently for the pressure.

The present hybrid filter may be related to the NLDE, where full LES and RANS solutions overlap in a part of the domain [164, 165, 186]. The NLDE is formally derived using a hierarchical scale separation operator, applying both an LES filter and a RANS operator, such that $\langle \bar{\boldsymbol{u}} \rangle = \langle \boldsymbol{u} \rangle$ holds by definition. This characteristic stands in contrast to the present approach, since the LES filter size is usually so coarse that the mean flow is not captured sufficiently by the LES scale and we have $\langle \bar{\boldsymbol{u}} \rangle \neq \langle \boldsymbol{u} \rangle$.

When this approach is applied to the incompressible Navier–Stokes equations, the derivatives of a_i with respect to the spatial coordinates vanish and the subgrid tensor becomes [97]:

$$\begin{aligned} \boldsymbol{\tau}_H =& a_1 \boldsymbol{\tau}_{\text{LES}} + a_2 \boldsymbol{\tau}_{\text{RANS}} + a_3 \boldsymbol{\tau}_3 + a_1 a_2 \left(\bar{\boldsymbol{u}} - \langle \boldsymbol{u} \rangle \right) \otimes \left(\bar{\boldsymbol{u}} - \langle \boldsymbol{u} \rangle \right) \\ &+ a_1 a_3 \left(\bar{\boldsymbol{u}} - \langle \bar{\boldsymbol{u}} \rangle \right) \otimes \left(\bar{\boldsymbol{u}} - \langle \bar{\boldsymbol{u}} \rangle \right) + a_2 a_3 \left(\langle \boldsymbol{u} \rangle - \langle \bar{\boldsymbol{u}} \rangle \right) \otimes \left(\langle \boldsymbol{u} \rangle - \langle \bar{\boldsymbol{u}} \rangle \right), \end{aligned} \tag{10.11}$$

with the LES and RANS subgrid tensors

$$\boldsymbol{\tau}_{\text{LES}} = \overline{\boldsymbol{u} \otimes \boldsymbol{u}} - \bar{\boldsymbol{u}} \otimes \bar{\boldsymbol{u}}, \tag{10.12}$$

$$\boldsymbol{\tau}_{\text{RANS}} = \langle \boldsymbol{u} \otimes \boldsymbol{u} \rangle - \langle \boldsymbol{u} \rangle \otimes \langle \boldsymbol{u} \rangle = \langle \boldsymbol{u}' \otimes \boldsymbol{u}' \rangle, \tag{10.13}$$

$$\boldsymbol{\tau}_3 = \langle \overline{\boldsymbol{u} \otimes \boldsymbol{u}} \rangle - \langle \bar{\boldsymbol{u}} \rangle \otimes \langle \bar{\boldsymbol{u}} \rangle = \langle \boldsymbol{\tau}_{\text{LES}} \rangle + \langle \bar{\boldsymbol{u}}' \otimes \bar{\boldsymbol{u}}' \rangle, \tag{10.14}$$

where $(\cdot)'$ denotes the fluctuating component of the respective quantity, i.e., $\boldsymbol{u} = \langle \boldsymbol{u} \rangle + \boldsymbol{u}'$. In order to simplify the subgrid terms in the following, we define a new velocity variable summarizing the statistical velocity contributions, $\widetilde{\boldsymbol{u}} = \langle \boldsymbol{u} \rangle - \langle \bar{\boldsymbol{u}} \rangle$.

Remark: The quantity $\bar{\boldsymbol{u}}$ is of course related to the standard polynomial solution and the variable $\widetilde{\boldsymbol{u}}$ to the enrichment solution as introduced in Chapter 6. This means, that we will explicitly solve for these components in the final numerical method and the velocity solution is considered in the form $\boldsymbol{u}_H = \bar{\boldsymbol{u}} + \widetilde{\boldsymbol{u}}$, as a sum of an LES and a RANS component.

Inserting the constants a_i in Equation (10.11) and application of the relation (10.10), $\langle \boldsymbol{u} \rangle = \langle \bar{\boldsymbol{u}} \rangle + \widetilde{\boldsymbol{u}}$, yields

$$\begin{aligned} \boldsymbol{\tau}_H =& \boldsymbol{\tau}_{\text{LES}} + \boldsymbol{\tau}_{\text{RANS}} - \boldsymbol{\tau}_3 + \underbrace{\left(\bar{\boldsymbol{u}} - \langle \boldsymbol{u} \rangle \right)}_{\bar{\boldsymbol{u}}' - \widetilde{\boldsymbol{u}}} \otimes \underbrace{\left(\bar{\boldsymbol{u}} - \langle \boldsymbol{u} \rangle \right)}_{\bar{\boldsymbol{u}}' - \widetilde{\boldsymbol{u}}} - \underbrace{\left(\bar{\boldsymbol{u}} - \langle \bar{\boldsymbol{u}} \rangle \right)}_{\bar{\boldsymbol{u}}'} \otimes \underbrace{\left(\bar{\boldsymbol{u}} - \langle \bar{\boldsymbol{u}} \rangle \right)}_{\bar{\boldsymbol{u}}'} \\ &- \underbrace{\left(\langle \boldsymbol{u} \rangle - \langle \bar{\boldsymbol{u}} \rangle \right)}_{\widetilde{\boldsymbol{u}}} \otimes \underbrace{\left(\langle \boldsymbol{u} \rangle - \langle \bar{\boldsymbol{u}} \rangle \right)}_{\widetilde{\boldsymbol{u}}} \qquad (10.15) \\ =& \boldsymbol{\tau}_{\text{LES}} - \langle \boldsymbol{\tau}_{\text{LES}} \rangle + \langle \boldsymbol{u}' \otimes \boldsymbol{u}' \rangle - \langle \bar{\boldsymbol{u}}' \otimes \bar{\boldsymbol{u}}' \rangle - \bar{\boldsymbol{u}}' \otimes \widetilde{\boldsymbol{u}} - \widetilde{\boldsymbol{u}} \otimes \bar{\boldsymbol{u}}'. \qquad (10.16) \end{aligned}$$

In the near-wall area, we assume that Reynolds stresses dominate, allowing the simplification

$$\boldsymbol{\tau}_H \approx \langle \boldsymbol{u}' \otimes \boldsymbol{u}' \rangle - \langle \bar{\boldsymbol{u}}' \otimes \bar{\boldsymbol{u}}' \rangle. \tag{10.17}$$

Herein, the LES subgrid stresses play a minor role in the vicinity of the wall, as barely any turbulence is resolved due to an overly coarse LES mesh and usually almost the entire solution is represented by $\widetilde{\boldsymbol{u}}$. In the present chapter, we consider the implicit LES capabilities of the numerical method (see Chapter 4), where the numerical upwind fluxes introduce an appropriate amount of dissipation in the high-frequency range, so any LES subgrid tensor is omitted in the following. Further remarks on how to include an explicit LES model are given in Section 10.2.2.

As a model for the remaining subgrid terms, we employ an eddy viscosity closure

$$\begin{aligned}\boldsymbol{\tau}_H &\approx \langle \boldsymbol{u}' \otimes \boldsymbol{u}' \rangle - \langle \bar{\boldsymbol{u}}' \otimes \bar{\boldsymbol{u}}' \rangle \\ &\approx -2\nu_t \left(\boldsymbol{\epsilon}\left(\langle \boldsymbol{u} \rangle\right) - \boldsymbol{\epsilon}\left(\langle \bar{\boldsymbol{u}} \rangle\right)\right) \\ &= -2\nu_t \boldsymbol{\epsilon}\left(\widetilde{\boldsymbol{u}}\right)\end{aligned} \tag{10.18}$$

with the RANS eddy viscosity ν_t. The relevant velocity component, on which the eddy viscosity model acts, is the RANS component $\widetilde{\boldsymbol{u}}$, which is a direct result of the formal definition of the three-level filter, as indicated in Equation (10.18).

Remark: The earlier elaborations on using a constant blending factor of $a_1 = 1$ manifest themselves in the numerical method: the LES DOFs are not directly impacted by the RANS eddy viscosity, so they are not damped.

For simplicity and since we only apply it to the inner layer in the present work, we employ Prandtl's algebraic mixing length eddy viscosity model including van Driest's damping function [63]

$$\nu_t = l_{\text{mix}}^2 |\boldsymbol{\epsilon}\left(\widetilde{\boldsymbol{u}}\right)|, \tag{10.19}$$

with the mixing length

$$l_{\text{mix}} = \kappa y_{\text{DES}} \left(1 - e^{-y^+/A^+}\right), \tag{10.20}$$

where $A^+ = 26$, the norm $|\boldsymbol{\epsilon}\left(\widetilde{\boldsymbol{u}}\right)| = \sqrt{2\boldsymbol{\epsilon}\left(\widetilde{\boldsymbol{u}}\right) : \boldsymbol{\epsilon}\left(\widetilde{\boldsymbol{u}}\right)}$, and a modified wall distance y_{DES}. An algebraic RANS model is particularly attractive in this application, since such a model does not require the solution of an additional transport equation, the definition of boundary conditions at the outer edge of the RANS/LES zone would be challenging, and the scheme is not subject to the diffusion number as the SA implementation in Chapter 5, if the viscous Navier–Stokes term is implemented in an implicit step. We further note that the Reynolds stresses in Equation (10.17) include the difference between the full and resolved stress tensor $\langle \boldsymbol{u}' \otimes \boldsymbol{u}' \rangle - \langle \bar{\boldsymbol{u}}' \otimes \bar{\boldsymbol{u}}' \rangle$. It would be possible to explicitly compute all terms involving the resolved scales in Equation (10.17) $(-\langle \bar{\boldsymbol{u}}' \otimes \bar{\boldsymbol{u}}' \rangle)$ by averaging over homogeneous directions and subtracting the result from the eddy viscosity tensor in the philosophy of Medic et al. [182]. In this work we account for the resolved Reynolds stress tensor by considering the classical DES approach of Spalart et al. [244], which limits the wall distance y with the characteristic grid length scale h times a model constant C_{DES}:

$$y_{\text{DES}} = \min\left(y, C_{\text{DES}} h\right) \approx \left(\frac{1}{y^\beta} + \frac{1}{\left(C_{\text{DES}} h\right)^\beta}\right)^{-1/\beta}, \tag{10.21}$$

and we use the latter modification in order to avoid kinks in the residual. For C_{DES}, Spalart et al. [244] specify a value of the order of unity and a careful calibration using turbulent channel flow yields $C_{\text{DES}} = 0.85$ with the solver presented in Section 10.4. We further employ the constant $\beta = 6$ and take h as the wall-normal width of the element Δy_e divided by $k + 1$, $h = \Delta y_e/(k+1)$, see also Section 5.5. Finally, the y^+-variable in the mixing length eddy viscosity model in Equation (10.20) is computed based on the definitions of $\tau_{w,h}$ and y_h in Equations (6.12) and (6.13).

Remark: Taking the wall-normal cell width in Equation (10.21) stands in contrast to the recommendation in DES of using $\max(\Delta x_1, \Delta x_2, \Delta x_3)$ [236] as a characteristic cell length (see

Section 5.5). The wall-normal grid size is chosen in this work as we require the flow to be fully turbulent at the interface $\partial\widetilde{\Omega}^I$ where the RANS component ends, and the resolved LES stresses are assumed to behave approximately universal in wall-normal direction. The requirement that turbulent eddies have to be resolved by the first off-wall cell limit the cell aspect ratio $\max(\Delta x_1, \Delta x_3)/\Delta y_1$ in the same way as for wall-stress models. The specific cell aspect ratio requirements are analyzed in detail in Section 10.5.1.

As the final result we obtain for the hybrid-filtered Navier–Stokes equations

$$\nabla\cdot(\bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}}) = 0, \tag{10.22}$$

$$\frac{\partial(\bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}})}{\partial t}+\nabla\cdot\left(\boldsymbol{\mathcal{F}}^c(\bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}})+p_H\boldsymbol{\mathcal{I}}-\boldsymbol{\mathcal{F}}^\nu(\bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}})-\boldsymbol{\mathcal{F}}^{\nu_t}(\widetilde{\boldsymbol{u}})\right) = \boldsymbol{f}, \tag{10.23}$$

where the components of the velocity solution have been expanded in $\boldsymbol{u}_H = \bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}}$. We solve for these two velocity components, an eddy-resolving velocity component $\bar{\boldsymbol{u}}$ and a statistical velocity component $\widetilde{\boldsymbol{u}}$, explicitly in our numerical method. The idea is that the method computes turbulent eddies where the LES mesh is sufficiently fine. If the mesh is too coarse to resolve the near-wall flow, the method automatically promotes the RANS modes and computes the flow, or parts of it, in a statistical sense. Since the resolution of the pressure is of much less relevance in turbulent boundary layers, the pressure is kept as a single variable filtered with the hybrid RANS/LES filter.

Remark: At first glance, a straightforward discretization of Equations (10.22) and (10.23) with the same function space for $\bar{\boldsymbol{u}}$ and $\widetilde{\boldsymbol{u}}$ does not seem to be possible, as there are formally seven unknowns (three components each in $\bar{\boldsymbol{u}}$ and $\widetilde{\boldsymbol{u}}$ as well as p_H) whereas solely four equations are available. As it will become clear in the subsequent section, this seemingly underdetermined problem statement can be solved in an elegant and fully consistent way in the context of the Galerkin method by taking phyiscally meaningful shape functions for the two velocity components, which are linearly independent. To this end, the obvious choice for such a basis is given by the concept of wall modeling via function enrichment. A more general framework of weak forms in a multiscale context, where different shape functions are used for particular scales, is discussed in the subsequent section and the particular function space used in this chapter is described in Section 10.3.

Outside of the near-wall layer $\widetilde{\Omega}$, only the LES scale $\bar{\boldsymbol{u}}$ is considered, and the coupling condition on the interface $\partial\widetilde{\Omega}^I$ in Figure 10.1 is $\boldsymbol{u}_H = \bar{\boldsymbol{u}}$. Additionally, we require that the viscous flux by the eddy viscosity term $\boldsymbol{\mathcal{F}}^{\nu_t}(\widetilde{\boldsymbol{u}})$ becomes zero in normal direction to avoid a kink in the solution, giving the interface condition

$$\boldsymbol{\mathcal{F}}^{\nu_t}(\widetilde{\boldsymbol{u}})\cdot\boldsymbol{n} = \boldsymbol{0} \quad \text{on } \partial\widetilde{\Omega}^I\times[0,\mathcal{T}]. \tag{10.24}$$

Since ν_t is in general not zero, this condition is equivalent to $\boldsymbol{\epsilon}(\widetilde{\boldsymbol{u}})\cdot\boldsymbol{n} = \boldsymbol{0}$.

10.2.2. Variational Multiscale Formulation

The goal of the present subsection is twofold: We explain, on an abstract level, how the solvability of Equations (10.22) and (10.23) is enabled despite the seemingly underdetermination of the equation system. This is done by choosing weighting functions for the weak form which are of the same structure as the velocity components and by taking linearly independent functions for the two scales $\bar{\boldsymbol{u}}$ and $\widetilde{\boldsymbol{u}}$ with finite dimension each. An efficient set of discrete linearly independent function spaces via function enrichment is presented in Section 10.3. Furthermore, we explain how the viscous operator in Equation (10.23) has to be modified in order to enable a physically suitable RANS solution to develop.

For this purpose, we derive an abstract weak form of Equations (10.22) and (10.23) in the standard procedure by multiplication of these equations with appropriate weighting functions $\boldsymbol{v} \in \mathcal{V}^{\boldsymbol{u}}$ as well as $q \in \mathcal{V}^{p}$ and integration over the whole spatial domain Ω. As a result we obtain the variational form of the incompressible Navier–Stokes equations as

$$\mathscr{C}\left(q, \bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}}\right) = 0, \qquad (10.25)$$

$$\mathscr{M}\Big(\boldsymbol{v}, \frac{\partial\left(\bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}}\right)}{\partial t}\Big) + \mathscr{F}^{c}\left(\boldsymbol{v}, \bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}}\right) + \mathscr{P}\left(\boldsymbol{v}, p_H\right) - \mathscr{F}^{\nu}\left(\boldsymbol{v}, \bar{\boldsymbol{u}}\right) - \mathscr{F}^{\nu+\nu_t}\left(\boldsymbol{v}, \widetilde{\boldsymbol{u}}\right) = \ell\left(\boldsymbol{v}\right). \qquad (10.26)$$

Herein, the terms correspond to the transient (mass) term $\mathscr{M}$, convective term $\mathscr{F}^{c}$, pressure term $\mathscr{P}$, the rearranged viscous terms $\mathscr{F}^{\nu}$ and $\mathscr{F}^{\nu+\nu_t}$, and the right-hand-side term ℓ, as well as the velocity-divergence term of the continuity equation $\mathscr{C}$. All terms are bilinear regarding $\boldsymbol{v}$ as well as $\bar{\boldsymbol{u}}$ and $\widetilde{\boldsymbol{u}}$ except for the convective term, which is nonlinear in the second slot. The detailed definitions of these operators will be elaborated in the context of the solver description in Section 10.4.

The classical Bubnov–Galerkin method suggests to take weighting functions of the same space as the solution functions. We choose appropriate solution functions $\bar{\boldsymbol{u}} \in \mathcal{V}^{\bar{u}}$ and $\widetilde{\boldsymbol{u}} \in \mathcal{V}^{\widetilde{u}}$, assume direct sum decomposition of the underlying spaces $\mathcal{V}^{\boldsymbol{u}} = \mathcal{V}^{\bar{u}} \oplus \mathcal{V}^{\widetilde{u}}$, i.e., $(\bar{\boldsymbol{u}} + \widetilde{\boldsymbol{u}}) \in \mathcal{V}^{\boldsymbol{u}}$, and we require $\mathcal{V}^{\bar{u}} \cap \mathcal{V}^{\widetilde{u}} = \{\boldsymbol{0}\}$ for linear independence. Employing the same basis for the weighting functions, we obtain two components for $\boldsymbol{v}$, $\bar{\boldsymbol{v}} = \mathcal{V}^{\bar{u}}$ and $\widetilde{\boldsymbol{v}} = \mathcal{V}^{\widetilde{u}}$, and we have $\boldsymbol{v} = \bar{\boldsymbol{v}} + \widetilde{\boldsymbol{v}}$. By inserting this ansatz into the weak forms (10.25) and (10.26), the momentum equation may be decomposed into two equations resembling the two scales of the solution according to the classical variational multiscale paradigm [117, 118]: an LES scale, weighted with $\bar{\boldsymbol{v}}$, and a RANS scale, weighted with $\widetilde{\boldsymbol{v}}$, yielding

$$\mathscr{C}\left(q, \bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}}\right) = 0, \qquad (10.27)$$

$$\mathscr{M}\Big(\bar{\boldsymbol{v}}, \frac{\partial\left(\bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}}\right)}{\partial t}\Big) + \mathscr{F}^{c}\left(\bar{\boldsymbol{v}}, \bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}}\right) + \mathscr{P}\left(\bar{\boldsymbol{v}}, p_H\right) - \mathscr{F}^{\nu}\left(\bar{\boldsymbol{v}}, \bar{\boldsymbol{u}}\right) - \mathscr{F}^{\nu+\nu_t}\left(\bar{\boldsymbol{v}}, \widetilde{\boldsymbol{u}}\right) = \ell\left(\bar{\boldsymbol{v}}\right), \qquad (10.28)$$

$$\mathscr{M}\Big(\widetilde{\boldsymbol{v}}, \frac{\partial\left(\bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}}\right)}{\partial t}\Big) + \mathscr{F}^{c}\left(\widetilde{\boldsymbol{v}}, \bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}}\right) + \mathscr{P}\left(\widetilde{\boldsymbol{v}}, p_H\right) - \mathscr{F}^{\nu}\left(\widetilde{\boldsymbol{v}}, \bar{\boldsymbol{u}}\right) - \mathscr{F}^{\nu+\nu_t}\left(\widetilde{\boldsymbol{v}}, \widetilde{\boldsymbol{u}}\right) = \ell\left(\widetilde{\boldsymbol{v}}\right). \qquad (10.29)$$

The equation for the continuity equation remains unchanged. Through the weighting of Equations (10.28) and (10.29) with linearly independent function spaces, the equation system is well posed and resolves the underdetermination discussed in the previous section.

The equation for the LES scale, Equation (10.28), is essentially a standard LES approach on a background convective velocity $\widetilde{\boldsymbol{u}}$, similar to NLDE [164], except for an additional eddy viscosity term based on $\widetilde{\boldsymbol{u}}$, providing the LES scale with physical dissipation in regions where the energy-carrying scales are not sufficiently resolved. Potential LES subgrid models contained in $\boldsymbol{\tau}_{\mathrm{LES}}$ would also be added to this equation.

Equation (10.29) represents the RANS scale, which resolves a variable fraction of the averaged velocity. It may be observed that the LES solution couples into the RANS equation in (10.29) and this coupling should be investigated in more detail. Considering again the NLDE methodology as a reference, the RANS equations are well defined without the additional LES source terms. However, the definition of the eddy viscosity in this work (Equation (10.19)) takes the resolved Reynolds stresses of the LES arising in the convective term, such that the full convective flux should be included, and thus the transient term accordingly. In contrast, there is no physical justification for the viscous LES part. Indeed, our numerical tests revealed that the method tends to promote an unphysical RANS mode, such as a RANS solution directed adverse to the primary flow direction even in attached boundary layers, if the viscous LES term is included in the RANS equation. Further arguments to be considered are that we expect the viscous RANS flux to be dominant in the underresolved LES region $\left(|\boldsymbol{\mathcal{F}}^{\nu}(\bar{\boldsymbol{u}})| \ll |\boldsymbol{\mathcal{F}}^{\nu+\nu_t}(\widetilde{\boldsymbol{u}})|\right)$. Additionally, numerical stability issues according to a coercivity analysis (in Section 10.4.2) suggest that this term would significantly restrict the range of application if taken into account. We conclude that the viscous LES term should be canceled from the RANS equation according to

$$\mathscr{M}\left(\widetilde{\boldsymbol{v}}, \frac{\partial\left(\bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}}\right)}{\partial t}\right) + \mathscr{F}^{c}\left(\widetilde{\boldsymbol{v}}, \bar{\boldsymbol{u}}+\widetilde{\boldsymbol{u}}\right) + \mathscr{P}\left(\widetilde{\boldsymbol{v}}, p_H\right) - \mathscr{F}^{\nu+\nu_t}\left(\widetilde{\boldsymbol{v}}, \widetilde{\boldsymbol{u}}\right) = \ell\left(\widetilde{\boldsymbol{v}}\right) \qquad (10.30)$$

in replacement of Equation (10.29). It is noted that this modification solely has an impact on one component of the equations and therefore primarily represents a measure for helping the method to find a physically suitable composition of $\bar{\boldsymbol{u}}$ and $\widetilde{\boldsymbol{u}}$ within the solution space. Similar modifications in the framework of the variational multiscale method are frequently considered regarding scale separation into large and small resolved/unresolved scales in LES turbulence modeling, see, e.g., the review article [100].

In this section, a consistent set of governing equations has been derived, which decomposes the incompressible Navier–Stokes equations into a RANS scale and an LES scale. We have argued that a linearly independent function space is required for these two components in order to guarantee solvability of the equation system. In the next section, we discuss how wall modeling via function enrichment may be used to construct a highly efficient discrete function space for $\bar{\boldsymbol{u}}$ and $\widetilde{\boldsymbol{u}}$.

10.3. RANS and LES Velocity Components Using Wall Modeling via Function Enrichment

A key ingredient of the present multiscale wall model is that wall modeling via function enrichment as presented in the previous chapters provides appropriate solution spaces for the LES and RANS components in the same solution vector. This is done by constructing additional shape shape functions using a wall-law as enrichment function, as it was described in Chapter 6 and applied in Chapters 7 to 9. As a consequence, the function space is *a priori* capable of representing the averaged (RANS) boundary layer profile of the inner layer with very few DOFs. In addition, the standard high-order polynomial is used to resolve eddies where the mesh is sufficiently fine. Since these basis functions are linearly independent, they may be used to approximate the RANS and LES scales in the variational multiscale method described in Section 10.2.2. This function space enables the use of very coarse meshes in the near-wall region along with exact boundary conditions since the function space is capable of resolving the sharp boundary layer gradient present in the mean velocity. The ingredients necessary to construct such a function space were already described in detail in Chapter 6 and the enrichment formulation considered in this chapter is summarized in the following.

According to Equation (6.1), the discrete velocity solution consists of two parts, the high-order polynomial component $\bar{\boldsymbol{u}}_h\left(\boldsymbol{x},t\right)$ and the enrichment component $\widetilde{\boldsymbol{u}}_h$, yielding

$$\boldsymbol{u}_h\left(\boldsymbol{x},t\right)=\bar{\boldsymbol{u}}_h\left(\boldsymbol{x},t\right)+\widetilde{\boldsymbol{u}}_h\left(\boldsymbol{x},t\right). \tag{10.31}$$

These two components are of course related to the respective RANS and LES contributions of the solution as introduced in the previous section. In each element, we explicitly solve for an LES component in terms of the high-order polynomial contribution $\bar{\boldsymbol{u}}_h$. The RANS velocity component is approximated by the enrichment shape functions. As indicated in Figure 10.1, the subdomain $\widetilde{\Omega}_h$ corresponds to the RANS/LES zone and this layer is chosen equivalent to the width of the first off-wall cell in the present high-order context (e.g. $k = 4$). If $k \leq 2$, two cell layers may be considered. The enrichment function is weighted with a polynomial of degree $l = \{0, 1\}$ with an emphasis on the constant case ($l = 0$) analogous to the preceding chapter. Arguments for this choice are discussed in the numerical analysis of this parameter in Section 10.5.1. The resulting decomposition of $\boldsymbol{u}_h$ into the two components is illustrated in Figure 6.2.

Regarding the choice of the enrichment function, we employed Spalding's law in the previous Chapters 7 to 9, which has proven to be suitable both for RANS, DES, and LES. Preliminary investigations of the present wall model have indicated the substantial potential of using the particular wall function which is consistent with the eddy viscosity model presented in Section 10.2.1 in the viscous sublayer and the buffer layer ($y^+ < 30$). For example, the predictions of the wall shear stress could be enhanced in accuracy and robustness. This wall function has been proposed by van Driest [63] and is defined in Equation (6.8); see also the numerical evaluation routines in Appendix A.2.

As the enrichment function is universal with respect to the wall coordinate $y^+ = y\sqrt{\tau_w/\rho}/\nu$, the wall coordinate scales the enrichment function in wall-normal direction to match the local wall shear stress. To this end, the adaptation algorithm as introduced in Chapter 6.3.2.4 with the modifications described in Chapter 8 is used. As in the previous chapters, the wall shear stress present in the wall y^+ variable is recomputed from the velocity solution of the previous time step. Due to the transient character of the simulation, this choice introduced a certain 'lag' in $\tau_{w,h}$. Due to the semi-explicit time integration scheme considered, this error is small.

Finally, the present adaptation algorithm also allows to switch off the wall model when not needed, i.e., when the standard polynomial component is sufficient to resolve the necessary scales, see Section 8.1. In particular, one observes problems in conditioning when the enriched element spans a y^+-range smaller than approximately 15 wall units, depending on the respective choice of k and l, as the standard and enrichment shape functions become close to linearly dependent when $\tau_w \to 0$, yielding $\psi \to 0$. This issue is circumvented by switching off the enrichment and the complete multiscale wall model if all quadrature points of the wall-nearest cell lie in $y^+ < 30$. A parameter study regarding the latter quantity, using turbulent channel flow with the setup applied in Figure 10.7, yielded almost indistinguishable results for values of y^+_{max} in the range of 20 to 30. In case y^+ becomes larger than 30 at any quadrature point of this cell in a subsequent time step, the wall model is simply switched on again by taking the enrichment into account in the solution of the new time step and an appropriate RANS solution develops automatically within a few time steps.

10.4. Galerkin Formulation and Implementation

The multiscale wall model is implemented in the high-order DG solver presented in Chapter 3. The modifications of the viscous term introduced in Equation (10.30) require major modifications of the viscous term. In addition, the viscous term is reformulated to take into account the spatially varying material parameter given through the eddy viscosity. In order to allow an accurate description of the formulation, the important aspects of the Galerkin formulation are repeated in the following (Section 10.4.1). The present section also presents a numerical stability analysis of the viscous multiscale term (Section 10.4.2) and discusses the implementation of the enriched elements (Section 10.4.3).

10.4.1. Galerkin Formulation

Variational formulations are derived for all steps of the time integration scheme (Equations (3.2) to (3.6)) and the weak forms are related to the symbolic variational formulations discussed in Section 10.2.2. These weak forms are adapted from Chapters 3 and 5 with major modifications of the viscous term as summarized in the following. To this end, we consider the notation used in Section 3.3.1. Regarding the imposition of boundary conditions, the present chapter is limited to no-slip and periodic conditions analogous to Section 5.2. These boundary conditions are imposed by prescribing external values for ϕ^+ and $\nabla\phi^+$ on $\partial\Omega^D$ and interface conditions on $\partial\widetilde{\Omega}^I$. The weak form of the multiscale model additionally makes use of the definitions $\widetilde{\Omega}_e = \Omega_e \cap \widetilde{\Omega}$ (enriched cell interior), $\partial\widetilde{\Omega}^I_e = \partial\Omega_e \cap \partial\widetilde{\Omega}^I$ (face on $\partial\widetilde{\Omega}^I$, exactly one face enriched), and $\partial\widetilde{\Omega}_e = \partial\Omega_e \cap \widetilde{\Omega} \backslash \partial\widetilde{\Omega}^I_e$ (both adjacent faces enriched).

We derive element-wise weak forms for each term of the Navier–Stokes equations by multiplying the respective term with a weighting function and integrating over one element volume:

Mass Term. The mass term is obtained without further modification as

$$\mathscr{M}_e(\boldsymbol{v},\boldsymbol{u}) = (\boldsymbol{v},\boldsymbol{u})_{\Omega_e}\,. \tag{10.32}$$

Neighboring cells are not connected due to the absence of face terms.

Convective Term. The weak form is integrated by parts, yielding

$$\mathscr{F}^c_e(\boldsymbol{v},\boldsymbol{u}) = -\left(\nabla\boldsymbol{v},\boldsymbol{\mathcal{F}}^c\left(\boldsymbol{u}\right)\right)_{\Omega_e} + \left(\boldsymbol{v},\boldsymbol{\mathcal{F}}^{c*}\left(\boldsymbol{u}\right)\cdot\boldsymbol{n}_\Gamma\right)_{\partial\Omega_e} \tag{10.33}$$

for the convective term. The convective flux is defined via the local Lax–Friedrichs flux

$$\boldsymbol{\mathcal{F}}^{c*}\left(\boldsymbol{u}\right) = \{\{\boldsymbol{\mathcal{F}}^c\left(\boldsymbol{u}\right)\}\} + \Lambda/2[\![\boldsymbol{u}]\!], \tag{10.34}$$

and we take the maximum of $\Lambda = \max(\lambda^-,\lambda^+)$ across the interface of the largest eigenvalue of the flux Jacobian (see Equation (3.10)).

Pressure Gradient. The pressure gradient is integrated by parts as suggested in Section 3.4, yielding

$$\mathscr{P}_e(\boldsymbol{v}, p) = -\left(\nabla\cdot \boldsymbol{v}, p\right)_{\Omega_e} + \left(\boldsymbol{v}, \{\{p\}\}\boldsymbol{n}_\Gamma\right)_{\partial\Omega_e} . \tag{10.35}$$

Velocity Divergence. The velocity divergence is integrated by parts according to Section 3.4 as well and results in

$$\mathscr{C}_e(q, \boldsymbol{u}) = -\left(\nabla q, \boldsymbol{u}\right)_{\Omega_e} + \left(q, \{\{\boldsymbol{u}\}\}\cdot \boldsymbol{n}_\Gamma\right)_{\partial\Omega_e} . \tag{10.36}$$

Viscous Multiscale Term. We discuss a suitable viscous term in view of the modifications introduced in Section 10.2.2. The baseline viscous implementation of our solver is an interior penalty method in symmetric form in the standard flow solver (Chapter 3) and in nonsymmetric form in the enriched solver (Chapter 5). The nonsymmetric variant of the interior penalty method has the advantage of being stable with very low requirements on the penalty parameter [220], which is beneficial in the case of nonpolynomial shape functions, such as the present boundary layer enrichment, where the derivation of inverse estimates on the fly is not economical.

The viscous term considered in the multiscale model follows Chapter 5 and the impact of the LES solution $\bar{\boldsymbol{u}}$ on the equations for the RANS scale weighted with $\widetilde{\boldsymbol{v}}$ are canceled as suggested in Section 10.2.2. We propose the following formulation of the viscous multiscale term:

$$\begin{array}{lllll}
 & \text{volume terms :} & \text{adjoint terms :} & \text{std. consist. terms :} & \text{penalty terms :} \\
 & \multicolumn{4}{l}{\mathscr{F}_e^{\nu}(\bar{\boldsymbol{v}}, \widetilde{\boldsymbol{v}}, \bar{\boldsymbol{u}}, \widetilde{\boldsymbol{u}}) =} \\
\widetilde{\boldsymbol{v}},\widetilde{\boldsymbol{u}} : & -\left(\boldsymbol{\epsilon}\left(\widetilde{\boldsymbol{v}}\right), \boldsymbol{\mathcal{F}}^{\nu+\nu_t}\left(\widetilde{\boldsymbol{u}}\right)\right)_{\widetilde{\Omega}_e} & -\left(w\boldsymbol{\mathcal{F}}^{\nu+\nu_t}\left(\widetilde{\boldsymbol{v}}\right), [\![\widetilde{\boldsymbol{u}}]\!]\right)_{\partial\widetilde{\Omega}_e} & +\left(\widetilde{\boldsymbol{v}}, \{\{\boldsymbol{\mathcal{F}}^{\nu+\nu_t}\left(\widetilde{\boldsymbol{u}}\right)\}\}\cdot\boldsymbol{n}_\Gamma\right)_{\partial\widetilde{\Omega}_e} & -\left(\widetilde{\boldsymbol{v}}, \tau_{\mathrm{IP}}\nu[\![\widetilde{\boldsymbol{u}}]\!]\cdot\boldsymbol{n}_\Gamma\right)_{\partial\Omega_e} \\
\widetilde{\boldsymbol{v}},\bar{\boldsymbol{u}} : & & -\left(w\boldsymbol{\mathcal{F}}^{\nu+\nu_t}\left(\widetilde{\boldsymbol{v}}\right), [\![\bar{\boldsymbol{u}}]\!]\right)_{\partial\widetilde{\Omega}_e} & +\left(\widetilde{\boldsymbol{v}}, \{\{\boldsymbol{\mathcal{F}}^{\nu}\left(\bar{\boldsymbol{u}}\right)\}\}\cdot\boldsymbol{n}_\Gamma\right)_{\partial\widetilde{\Omega}_e^I} & -\left(\widetilde{\boldsymbol{v}}, \tau_{\mathrm{IP}}\nu[\![\bar{\boldsymbol{u}}]\!]\cdot\boldsymbol{n}_\Gamma\right)_{\partial\Omega_e} \\
\bar{\boldsymbol{v}},\widetilde{\boldsymbol{u}} : & -\left(\boldsymbol{\epsilon}\left(\bar{\boldsymbol{v}}\right), \boldsymbol{\mathcal{F}}^{\nu+\nu_t}\left(\widetilde{\boldsymbol{u}}\right)\right)_{\widetilde{\Omega}_e} & -\left(\dfrac{1}{2}\boldsymbol{\mathcal{F}}^{\nu}\left(\bar{\boldsymbol{v}}\right), [\![\widetilde{\boldsymbol{u}}]\!]\right)_{\partial\widetilde{\Omega}_e^I} & +\left(\bar{\boldsymbol{v}}, \{\{\boldsymbol{\mathcal{F}}^{\nu+\nu_t}\left(\widetilde{\boldsymbol{u}}\right)\}\}\cdot\boldsymbol{n}_\Gamma\right)_{\partial\widetilde{\Omega}_e} & -\left(\bar{\boldsymbol{v}}, \tau_{\mathrm{IP}}\nu[\![\widetilde{\boldsymbol{u}}]\!]\cdot\boldsymbol{n}_\Gamma\right)_{\partial\Omega_e} \\
\bar{\boldsymbol{v}},\bar{\boldsymbol{u}} : & -\left(\boldsymbol{\epsilon}\left(\bar{\boldsymbol{v}}\right), \boldsymbol{\mathcal{F}}^{\nu}\left(\bar{\boldsymbol{u}}\right)\right)_{\Omega_e} & -\left(\dfrac{1}{2}\boldsymbol{\mathcal{F}}^{\nu}\left(\bar{\boldsymbol{v}}\right), [\![\bar{\boldsymbol{u}}]\!]\right)_{\partial\Omega_e} & +\left(\bar{\boldsymbol{v}}, \{\{\boldsymbol{\mathcal{F}}^{\nu}\left(\bar{\boldsymbol{u}}\right)\}\}\cdot\boldsymbol{n}_\Gamma\right)_{\partial\Omega_e} & -\left(\bar{\boldsymbol{v}}, \tau_{\mathrm{IP}}\nu[\![\bar{\boldsymbol{u}}]\!]\cdot\boldsymbol{n}_\Gamma\right)_{\partial\Omega_e}
\end{array} \tag{10.37}$$

where all terms are expanded into the individual contributions from the RANS and LES scale. Herein, the row $(\widetilde{\boldsymbol{v}}, \widetilde{\boldsymbol{u}})$ represents the RANS contribution to the RANS equation, the row $(\widetilde{\boldsymbol{v}}, \bar{\boldsymbol{u}})$ the LES contribution to the RANS equation, the row $(\bar{\boldsymbol{v}}, \widetilde{\boldsymbol{u}})$ the RANS contribution to the LES equations and finally the last row the LES contributions to the LES scale. The columns correspond to the respective volume terms, the adjoint face terms, the standard consistency face terms, and the penalty terms. We note the skew-symmetry of the standard consistency and adjoint face terms as well as the symmetry of the penalty terms.

In Equation (10.37) we do not entirely cancel the LES impact on the RANS equations but shift the adjoint term at the inner enriched faces $\partial\widetilde{\Omega}_e$ from the row $(\bar{\boldsymbol{v}}, \widetilde{\boldsymbol{u}})$ to the row $(\widetilde{\boldsymbol{v}}, \bar{\boldsymbol{u}})$. This modification is consistent, since it relies on the discontinuity of the velocity solution at element boundaries, and is required in order to guarantee a stable scheme through skew-symmetric face terms. The skew-symmetry is beneficial in the context of the coercivity analysis in the subsequent section. In contrast, on the interface between enriched and nonenriched cells $\partial\widetilde{\Omega}_e^I$, the opposite face terms are considered, namely the adjoint term in row $(\bar{\boldsymbol{v}}, \widetilde{\boldsymbol{u}})$ and the skew-symmetric standard consistency term in row $(\widetilde{\boldsymbol{v}}, \bar{\boldsymbol{u}})$ again in order to allow the fulfillment of the coercivity argument. The resulting Galerkin formulation is in agreement with the interface conditions $\bar{\boldsymbol{u}}^+ = \bar{\boldsymbol{u}}^- + \widetilde{\boldsymbol{u}}^-$ and $\widetilde{\boldsymbol{u}}^+ = \boldsymbol{0}$ on $\partial\widetilde{\Omega}_e^I$ (i.e., $[\![\boldsymbol{u}]\!] = (\bar{\boldsymbol{u}}^- + \widetilde{\boldsymbol{u}}^- - \bar{\boldsymbol{u}}^+) \otimes \boldsymbol{n}_\Gamma$ and $[\![\widetilde{\boldsymbol{u}}]\!] = \widetilde{\boldsymbol{u}}^- \otimes \boldsymbol{n}_\Gamma$), where the left face $(\cdot)^-$ is enriched and the right face $(\cdot)^+$ is not enriched, without loss of generality. However, the condition $\boldsymbol{\epsilon}(\widetilde{\boldsymbol{u}}^-) \cdot \boldsymbol{n} = \boldsymbol{0}$ is not exactly fulfilled as we have $\boldsymbol{\mathcal{F}}^{\nu+\nu_t}(\widetilde{\boldsymbol{u}}^-) \cdot \boldsymbol{n}_\Gamma^- = \{\{\boldsymbol{\mathcal{F}}^{\nu}(\bar{\boldsymbol{u}})\}\} \cdot \boldsymbol{n}_\Gamma$ on $\partial\widetilde{\Omega}_e^I$, but the error is considered to be small since kinks have not been observed in the solution at that boundary. As an alternative, numerical tests have shown that the standard consistency term in row $(\widetilde{\boldsymbol{v}}, \bar{\boldsymbol{u}})$ may be neglected, which yields a better fulfillment of the latter interface condition while problems in stability have not been observed with this modified formulation. The numerical examples shown in Section 10.5 yet employ the variant with proven stability.

We further note that the material parameter ν is applied to the interior penalty term in the whole domain. As interior penalty stabilization parameter τ_{IP} we use the definition by Hillewaert [113] (see also Equations (3.15) and (3.14)), since this choice has yielded favorable results in underresolved turbulent flows (ILES) in Chapter 4. On the Dirichlet boundary, we have taken measures to enhance the accuracy of the weakly enforced no-slip condition in Chapter 5 applied with the enrichment in Chapter 8. In the present chapter, we increase the interior penalty parameter τ_{IP} on all Dirichlet boundary faces by a factor of 10, which has a similar effect while being more solver-friendly in 3D. The weights of the averaging operators with spatially varying material parameter ν_t included in the $\boldsymbol{\mathcal{F}}^{\nu+\nu_t}$-terms are given through harmonic weighting [36] similar to Equation (5.20):

$$w^- = \frac{\nu + \nu_t^+}{2\nu + \nu_t^- + \nu_t^+}, \qquad w^+ = \frac{\nu + \nu_t^-}{2\nu + \nu_t^- + \nu_t^+}. \tag{10.38}$$

On $\partial\widetilde{\Omega}^I$, all terms including the eddy viscosity vanish, making a consideration of the varying material law unnecessary.

Velocity Div-Div Penalty. The present scheme requires a div-div penalty operator for stabilization of mass conservation, see Section 3.4 for a detailed discussion. The corresponding operator reads

$$\mathscr{D}_e(\boldsymbol{v}, \boldsymbol{u}) = (\nabla\cdot \boldsymbol{v}, \tau_{\mathrm{D}} \nabla\cdot \boldsymbol{u})_{\Omega_e}. \tag{10.39}$$

We note that the contribution of this term vanishes if the velocity field is exactly divergence-free, as it relies on the continuity residual $\nabla\cdot \boldsymbol{u}$. The stabilization parameter τ_{D} is given in Equation (3.30).

Pressure Laplace Term. The pressure Poisson equation further requires the discretization of a Laplace term. We use the symmetric interior penalty method [10], yielding

$$\mathscr{L}_e\left(q, p^{n+1}\right) = -\left(\nabla q, \nabla p^{n+1}\right)_{\Omega_e} + \frac{1}{2}\left(\nabla q, [\![p^{n+1}]\!]\right)_{\partial\Omega_e} + \left(q, \boldsymbol{\mathcal{P}}^* \cdot \boldsymbol{n}_\Gamma\right)_{\partial\Omega_e}. \tag{10.40}$$

For the numerical flux $\boldsymbol{\mathcal{P}}^*$, we include an interior penalty term on internal faces and use the pressure boundary conditions on the Dirichlet boundary (5.12) as given in Equation (5.23).

Finally, these weak operators allow a compact definition of the dual-splitting scheme in element-wise Galerkin form. The explicit convective step reads

$$\frac{\gamma_0 \mathscr{M}_e(\boldsymbol{v}_h, \hat{\boldsymbol{u}}_h) - \sum_{i=0}^{J-1} \left(\alpha_i \mathscr{M}_e(\boldsymbol{v}_h, \boldsymbol{u}_h^{n-i})\right)}{\Delta t} \\ = -\sum_{i=0}^{J-1} \beta_i \mathscr{F}_e^c \left(\boldsymbol{v}_h, \boldsymbol{u}_h^{n-i}\right) + \mathscr{M}_e \left(\boldsymbol{v}_h, \boldsymbol{f}_h^{n+1}\right). \quad (10.41)$$

The pressure Poisson equation is

$$-\mathscr{L}_e \left(q_h, p_h^{n+1}\right) = -\frac{\gamma_0}{\Delta t} \mathscr{C}_e \left(q_h, \hat{\boldsymbol{u}}_h\right) \quad (10.42)$$

and as boundary condition for $\hat{\boldsymbol{u}}_h$ on $\partial\Omega^D$ we employ the interior value $\{\{\hat{\boldsymbol{u}}_h\}\} = \hat{\boldsymbol{u}}_h^-$. The element-wise projection step is obtained by adding the div-div penalty term (10.39) to the left-hand side, resulting in

$$\mathscr{M}_e \left(\boldsymbol{v}_h, \hat{\hat{\boldsymbol{u}}}_h\right) + \mathscr{D}_e \left(\boldsymbol{v}_h, \hat{\hat{\boldsymbol{u}}}_h\right) = \mathscr{M}_e \left(\boldsymbol{v}_h, \hat{\boldsymbol{u}}_h\right) - \mathscr{P}_e \left(\boldsymbol{v}_h, p_h^{n+1}\right) \quad (10.43)$$

where boundary conditions on the pressure variable are again applied using the internal value $\{\{p_h^{n+1}\}\} = p_h^{n+1,-}$ on $\partial\Omega^D$. The viscous step may be written as

$$\frac{\gamma_0}{\Delta t} \left(\mathscr{M}_e \left(\boldsymbol{v}_h, \boldsymbol{u}_h^{n+1}\right) - \mathscr{M}_e \left(\boldsymbol{v}_h, \hat{\hat{\boldsymbol{u}}}_h\right)\right) = \mathscr{F}_e^\nu \left(\bar{\boldsymbol{v}}_h, \widetilde{\boldsymbol{v}}_h, \bar{\boldsymbol{u}}_h^{n+1}, \widetilde{\boldsymbol{u}}_h^{n+1}\right). \quad (10.44)$$

Boundary conditions for all remaining quantities on no-slip walls are specified using the mirror principle [112], defining the exterior velocity contribution with $\boldsymbol{u}_h^+ = -\boldsymbol{u}_h^- + 2\boldsymbol{g}_{\boldsymbol{u}} = -\boldsymbol{u}_h^-$ on $\partial\Omega^D$.

10.4.2. Coercivity Analysis

We sketch a coercivity analysis of the viscous term in order to investigate the stability of the numerical scheme. The Lax–Milgram theorem ensures solvability of the variational Helmholtz problem (10.44) if there is a constant $C \geq 0$ such that

$$-\mathscr{F}_e^\nu \left(\bar{\boldsymbol{v}}, \widetilde{\boldsymbol{v}}, \bar{\boldsymbol{v}}, \widetilde{\boldsymbol{v}}\right) \geq C \left(\|\bar{\boldsymbol{v}}\|^2 + \|\widetilde{\boldsymbol{v}}\|^2\right), \text{ for all } \bar{\boldsymbol{v}} \in \mathcal{V}^{\bar{\boldsymbol{u}}}, \widetilde{\boldsymbol{v}} \in \mathcal{V}^{\widetilde{\boldsymbol{u}}} \quad (10.45)$$

holds. Herein, we disregard the mass terms, as they always yield a positive contribution on the left-hand side of the inequality and the Helmholtz equation degenerates to an L^2 projection if $-\mathscr{F}_e^\nu \to 0$.

Due to their skew-symmetric construction, the face terms on $\partial\widetilde{\Omega}_e$ in Equation (10.37) cancel each other when the solution and weighting functions are equal. Solely the interior penalty face terms remain, which have an entirely positive contribution. Therefore, the following inequality holds for enriched cells,

$$- \mathscr{F}_e^\nu \left(\bar{\boldsymbol{v}}, \widetilde{\boldsymbol{v}}, \bar{\boldsymbol{v}}, \widetilde{\boldsymbol{v}}\right) \\ \geq \left(\boldsymbol{\epsilon}\left(\bar{\boldsymbol{v}}\right), 2\nu \boldsymbol{\epsilon}\left(\bar{\boldsymbol{v}}\right)\right)_{\widetilde{\Omega}_e} + \left(\boldsymbol{\epsilon}\left(\bar{\boldsymbol{v}}\right), 2(\nu + \nu_t) \boldsymbol{\epsilon}\left(\widetilde{\boldsymbol{v}}\right)\right)_{\widetilde{\Omega}_e} + \left(\boldsymbol{\epsilon}\left(\widetilde{\boldsymbol{v}}\right), 2(\nu + \nu_t) \boldsymbol{\epsilon}\left(\widetilde{\boldsymbol{v}}\right)\right)_{\widetilde{\Omega}_e} \quad (10.46)$$

leaving only the volume terms for further consideration. The first and the last volume term are each symmetric and thus positive while the nonsymmetric second term is problematic as it can yield a negative contribution. In the following, we derive an estimate for this term in order to guarantee that the sum of all terms in the bilinear form is bounded from below by $C\left(\|\bar{\boldsymbol{v}}\|^2 + \|\widetilde{\boldsymbol{v}}\|^2\right)$.

Young's inequality

$$2ab \geq -\frac{a^2}{\varepsilon} - \varepsilon b^2 \tag{10.47}$$

for any $\varepsilon > 0$ applied to the second volume term in (10.46) results in:

$$\begin{aligned}&\left(\boldsymbol{\epsilon}(\bar{\boldsymbol{v}}), 2\widehat{(\nu+\nu_t)}\boldsymbol{\epsilon}(\widetilde{\boldsymbol{v}})\right)_{\Omega_e}\\ &\qquad \geq \left(\boldsymbol{\epsilon}(\bar{\boldsymbol{v}}), -\frac{\widehat{(\nu+\nu_t)}}{\varepsilon}\boldsymbol{\epsilon}(\bar{\boldsymbol{v}})\right)_{\Omega_e} + \left(\boldsymbol{\epsilon}\left(\widetilde{\boldsymbol{v}}\right), -\varepsilon\widehat{(\nu+\nu_t)}\boldsymbol{\epsilon}\left(\widetilde{\boldsymbol{v}}\right)\right)_{\Omega_e}.\end{aligned} \tag{10.48}$$

Herein, we have introduced the modified viscosity term $\widehat{(\nu+\nu_t)}$. Inserting this relation in inequality (10.46) yields an estimate for $-\mathscr{F}_e^{\nu}$ based on the two symmetric terms,

$$\begin{aligned}&-\mathscr{F}_e^{\nu}\left(\bar{\boldsymbol{v}}, \widetilde{\boldsymbol{v}}, \bar{\boldsymbol{v}}, \widetilde{\boldsymbol{v}}\right)\\ &\geq \left(\boldsymbol{\epsilon}(\bar{\boldsymbol{v}}), (2\nu - \frac{\widehat{(\nu+\nu_t)}}{\varepsilon})\boldsymbol{\epsilon}(\bar{\boldsymbol{v}})\right)_{\Omega_e} + \left(\boldsymbol{\epsilon}(\widetilde{\boldsymbol{v}}), \left(2(\nu+\nu_t) - \varepsilon\widehat{(\nu+\nu_t)}\right)\boldsymbol{\epsilon}(\widetilde{\boldsymbol{v}})\right)_{\Omega_e},\end{aligned} \tag{10.49}$$

allowing the conclusion that the Lax–Milgram theorem is fulfilled under the following conditions:

$$2\nu - \frac{\widehat{(\nu+\nu_t)}}{\varepsilon} \geq 0, \qquad 2(\nu+\nu_t) - \varepsilon\widehat{(\nu+\nu_t)} \geq 0. \tag{10.50}$$

Upon rearrangement, these inequalities directly result in the condition for coercivity of the viscous multiscale term:

$$\widehat{(\nu+\nu_t)} \leq 2\sqrt{\nu(\nu+\nu_t)}. \tag{10.51}$$

This means that the amount of viscous dissipation introduced in the LES scale by the RANS solution has to be limited if $\nu + \nu_t > 4\nu$ for reasons of stability. We apply the modified eddy viscosity

$$\widehat{\nu+\nu_t} = \min(\nu+\nu_t, 2\sqrt{\nu(\nu+\nu_t)}) \tag{10.52}$$

in the volume term of the row $(\bar{\boldsymbol{v}}, \widetilde{\boldsymbol{u}})$ in Equation (10.37), which comes along with a minor limitation of the application range of the wall model. This aspect will be investigated in detail in Section 10.5.1. We anticipate at this point that the width of the first off-wall cell should not exceed a y^+-range of approximately 120 wall units in the statistical data. The relation (10.52) guarantees the stability of the scheme in outliers due to turbulent fluctuations. We emphasize that this behavior is related to the particular discrete method used for the viscous multiscale term in the present work and we encourage mathematicians to develop alternative formulations in order to weaken or remove this limitation.

For completeness, we remark on the relations for the case with the viscous LES term according to Equation (10.29), i.e., without the modification introduced in Equation (10.30) and therefore including all terms in the rows $(\widetilde{\boldsymbol{v}}, \bar{\boldsymbol{u}})$ and $(\bar{\boldsymbol{v}}, \widetilde{\boldsymbol{u}})$ in $\widetilde{\Omega}_e$ as well as on $\partial\widetilde{\Omega}_e$ (in Equation (10.37)). The nonsymmetric term on the right hand side of inequality (10.46) would get a material factor of $2(2\nu + \nu_t)$ instead of $2(\nu + \nu_t)$, resulting in the condition $\widehat{(\nu + \nu_t)} \leq 2\sqrt{\nu(\nu + \nu_t)} - \nu$ for the material parameter of the volume term in the row $(\bar{\boldsymbol{v}}, \widetilde{\boldsymbol{u}})$. This condition limits the growth of the eddy viscosity variable if $\nu + \nu_t > \nu$, which would represent a too strong restriction in practical applications.

10.4.3. Implementation

The Galerkin formulations of the dual splitting scheme, Equations (10.41) to (10.44), are integrated by numerical quadrature, yielding a matrix formulation for each sub-step. The matrix forms are similar to the ones described in Sections 3.4.4 and 5.3.3. The numerical integration of the weak forms is performed using Gaussian quadrature as described in Section 6.3.3.1. In this chapter, we consider the matrix-free version of the inverse mass operator according to Section 6.3.3.2 and the mixed matrix-free evaluation of the residual of the matrix-free projection solver discussed in Section 6.3.3.3. All other steps of the scheme are evaluated as described in Section 3.5.

10.5. Numerical Examples

The present multiscale wall model is validated with turbulent channel flow (Section 10.5.1) as well as flow past periodic hills (Section 10.5.2). These two benchmark examples provide insight into the performance of the wall model both in attached and separated boundary layers. We follow the earlier recommendations of using a polynomial degree of $k = 4$ for the discretization of the LES scale, since this choice yields a good compromise between accuracy and time-to-solution within the present solver. In addition, a polynomial of 4th degree inside the wall layer is capable of resolving sufficient turbulence within a single layer of cells. The influence of the degree of the polynomial weighting of the enrichment function, l, is investigated and guidelines for the use of the wall model are formulated.

10.5.1. Turbulent Channel Flow

As a first validation example, we consider turbulent flow in a rectangular channel of the dimensions $2\pi\delta \times 2\delta \times \pi\delta$ in streamwise, vertical, and spanwise direction, respectively, with the channel half-width δ using the setup as considered in Section 4.2. The grid is graded towards the walls by the hyperbolic mapping given in Equation (4.1) with the mesh stretching parameter γ. The wall model is active in the first off-wall element layer and a typical mesh is shown in Figure 10.2. The initial conditions are solely applied on the polynomial DOFs of the solution and the RANS component develops in the course of the simulation once $y^+ > 30$ at any quadrature point of a wall-layer cell according to Section 8.1. The flow is driven by a constant pressure gradient derived from the nominal flow quantities and the results are normalized using the numerical wall shear stress in the friction velocity $u_\tau = \sqrt{\tau_w/\rho}$. We consider the friction Reynolds numbers $\mathrm{Re}_\tau = u_\tau\delta/\nu$ in accordance to DNS reference data at $\mathrm{Re}_\tau = 395$ [187], $\mathrm{Re}_\tau = 950$ [6], $\mathrm{Re}_\tau = 2{,}000$ [116], and $\mathrm{Re}_\tau = 5{,}200$ [172]. One snapshot of the turbulent flow is visualized

Table 10.1.: Multiscale-WMLES (MS-WMLES): Overview of simulation cases for the turbulent channel flow example: Investigation of several Reynolds numbers (Figure 10.4), comparison of the new approach to DDES including function enrichment (see Chapter 9) (Figure 10.5), grid stretching factors and mesh aspect ratios (Figure 10.6), transition from wall-resolved to WMLES (Figure 10.7), weighting of the enrichment function with a constant or linear basis (Figure 10.8), and the performance comparison of the wall model with wall-resolved LES (Figure 10.9). The number of cells per direction i is denoted N_{ie} and the normalized dimensions of the first off-wall cell is defined as $\Delta y^+_{1e} = \Delta y_{1e} u_\tau / \nu$ in wall-normal direction and $\Delta x^+_{ie} = \Delta x_{ie} u_\tau / \nu$ in direction i and $\Delta x^+_{e\,\max} = \max(\Delta x^+_{1e}, \Delta x^+_{2e}, \Delta x^+_{3e})$. The number of LES grid points per direction i is $N_i = (k+1) N_{ie}$ with $k = 4$ in all cases.

fig.	case	$N_{1e}\times N_{2e}\times N_{3e}$	Re$_\tau$	l	γ	Δy^+_{1e}	$\Delta x^+_{e\,\max}$	$\frac{\Delta y^+_{1e}(k+1)}{\Delta x^+_{e\,\max}}$
10.4	$ch395\_N12\times 8\times 12\_k4l0\_\gamma 0.8$	12×8×12	395	0	0.8	76	207	1.84
	$ch950\_N12\times 8\times 12\_k4l0\_\gamma 1.6$	12×8×12	950	0	1.6	91	497	0.92
	$ch2000\_N24\times 8\times 24\_k4l0\_\gamma 2.2$	24×8×24	2,000	0	2.2	96	524	0.92
	$ch5200\_N48\times 16\times 48\_k4l0\_\gamma 2.05$	48×16×48	5,200	0	2.05	114	681	0.84
10.5	$ch950\_N16^3\_k4l0\_\gamma 0.001$	16×16×16	950	0	0.001	119	373	1.60
	$ch950\_N16^3\_k4l0\_\gamma 0.001\_DDES$	16×16×16	950	0	0.001	119	373	-
10.6	$ch950\_N16^3\_k4l0\_\gamma 0.001$	16×16×16	950	0	0.001	119	373	1.60
	$ch950\_N16^3\_k4l0\_\gamma 0.8$	16×16×16	950	0	0.8	85	373	1.14
	$ch950\_N16^3\_k4l0\_\gamma 1.2$	16×16×16	950	0	1.2	59	373	0.79
	$ch950\_N16^3\_k4l0\_\gamma 1.5$	16×16×16	950	0	1.5	42	373	0.56
	$ch950\_N16^3\_k4l0\_\gamma 1.0$	16×16×16	950	0	1.0	72	373	0.97
	$ch950\_N16\times 16\times 32\_k4l0\_\gamma 1.0$	16×16×32	950	0	1.0	72	373	0.97
10.7	$ch950\_N24^3\_k4\_\gamma 2.25$	24×24×24	950	-	2.25	10	249	0.20
	$ch950\_N24^3\_k4l0\_\gamma 1.75$	24×24×24	950	0	1.75	19	249	0.38
	$ch950\_N24^3\_k4l0\_\gamma 1.5$	24×24×24	950	0	1.5	27	249	0.54
	$ch950\_N24^3\_k4l0\_\gamma 1.25$	24×24×24	950	0	1.25	36	249	0.72
	$ch950\_N24^3\_k4l0\_\gamma 1.0$	24×24×24	950	0	1.0	46	249	0.92
10.8	$ch950\_N16^3\_k4l0\_\gamma 1.2$	16×16×16	950	0	1.2	59	373	0.79
	$ch950\_N16^3\_k4l1\_\gamma 1.2$	16×16×16	950	1	1.2	59	373	0.79
10.9	$ch950\_N12\times 8\times 12\_k4l0\_\gamma 1.6$	12×8×12	950	0	1.6	91	497	0.92
	$ch950\_N24^3\_k4\_\gamma 2.25$	24×24×24	950	-	2.25	10	249	0.20
	$ch950\_N32^3\_k4\_\gamma 2.0$	32×32×32	950	-	2.0	10	187	0.27

in Figure 10.3 via contours of the velocity magnitude and turbulent vortex structures are made visible using iso-surfaces of the Q-criterion. All simulation parameters and discretization cases are summarized in Table 10.1.

Reynolds Number Independence. The results of a first application of the wall model including all four Reynolds numbers are depicted in Figure 10.4. Overall, excellent agreement is ob-

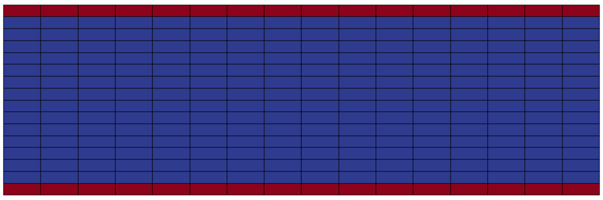


Figure 10.2.: Grid of the case $ch950\_N16^3\_k4l0\_\gamma0.001$. The solution is represented by a polynomial of degree four in each cell plus the enrichment shape functions in the wall-layer (red cells).

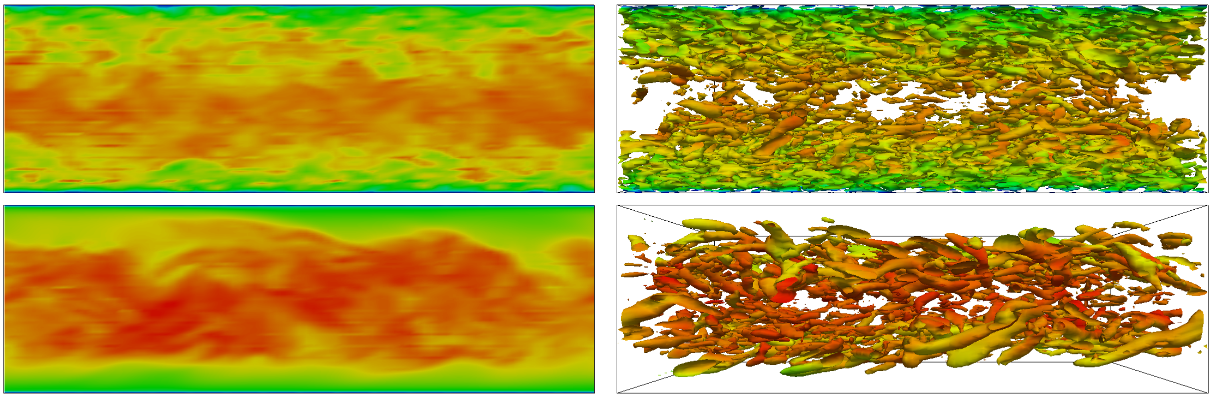


Figure 10.3.: MS-WMLES: Qualitative comparison of the present multiscale wall model including enrichment (top) with DDES using enrichment (see Chapter 9) (bottom) on the same mesh. Velocity magnitude (left) and visualization of turbulent eddies via the Q-criterion colored by velocity magnitude (right) of the cases $ch950\_N16^3\_k4l0\_\gamma0.001$ and $ch950\_N16^3\_k4l0\_\gamma0.001\_DDES$. All cases use the same color scale and the Q-criterion shows the same iso-value. Red indicates high and blue low values.

served both for the mean velocity, the RMS velocities and the RSS. Furthermore, it is apparent that the results do not depend on the Reynolds number. Regarding the mean velocity, the enrichment represents the whole mean velocity and the time-averaged LES solution is almost zero in the wall layer. Therefore, the LES result is only visualized for the lowest Reynolds number and not considered in the remainder of this section. The RSS is computed based on the resolved fluctuations, which explains the small gap to the reference data in the near-wall zone. In particular the u'^+ and w'^+ curves exhibit small unphysical peaks at the element interfaces, which is a typical result for coarse meshes in DG and has been observed in the previous chapters as well. These unphysical peaks vanish with increasing resolution.

Comparison to DDES. Since the primary incentive for the development of the present multiscale wall model was the deficiencies of DES in the hybrid RANS/LES transition region, we compare the two simulation methodologies directly. To this end, we consider DDES including wall modeling via function enrichment presented in the previous chapter for the spatial discretization, such that the exact same meshes can be used for both models. Figure 10.3 shows a

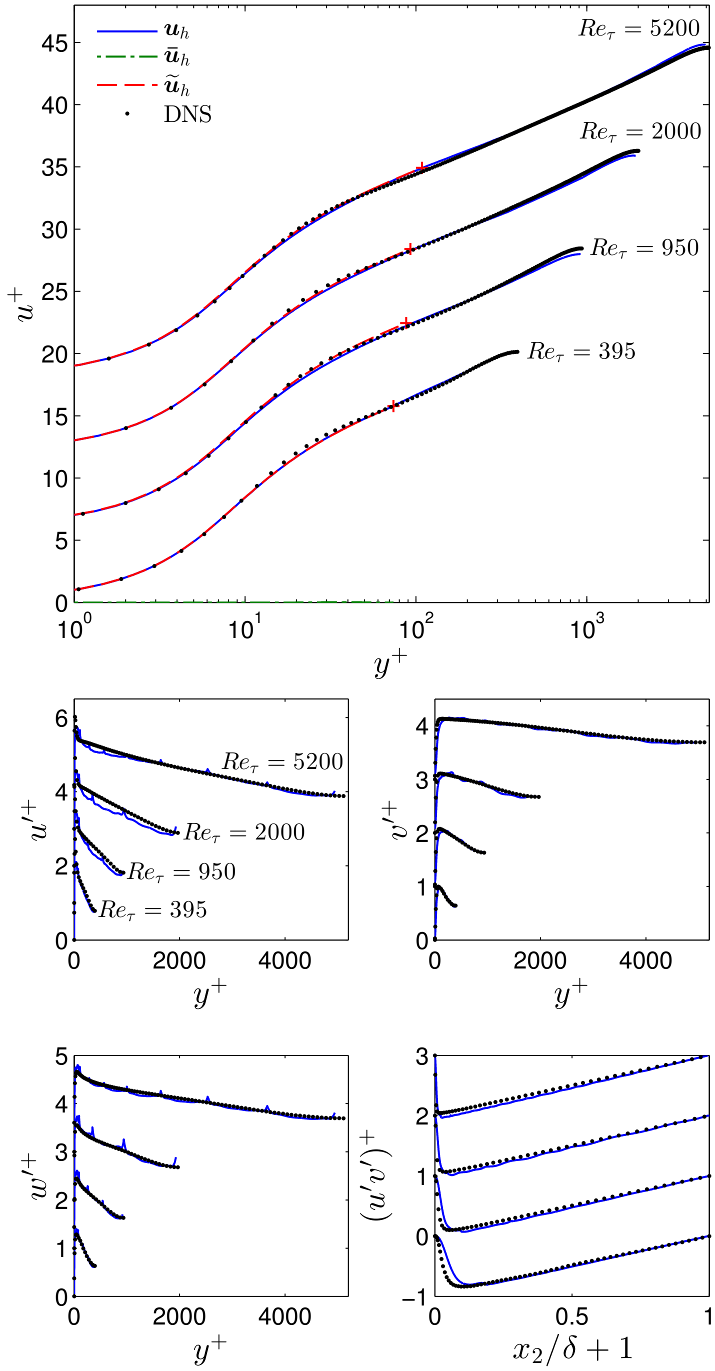


Figure 10.4.: MS-WMLES: WMLES of turbulent channel flow at several Reynolds numbers: Mean velocity (top) and RMS-velocities as well as RSS (bottom). All quantities are normalized according to $u^+ = \langle u_1 \rangle / u_\tau$, $u'^+ = \sqrt{\langle u_1'^2 \rangle}/u_\tau$, $v'^+ = \sqrt{\langle u_2^2 \rangle}/u_\tau$, $w'^+ = \sqrt{\langle u_3^2 \rangle}/u_\tau$, and $(u'v')^+ = \langle u_1 u_2 \rangle / u_\tau^2$. The polynomial part is inside the wall-layer only displayed for $\mathrm{Re}_\tau = 395$.

qualitative comparison of two simulations at $\mathrm{Re}_\tau = 950$; they deviate drastically. DDES operates in RANS mode in the inner layer up to $y/\delta \approx 0.05$ and the eddy viscosity acts on the polynomial

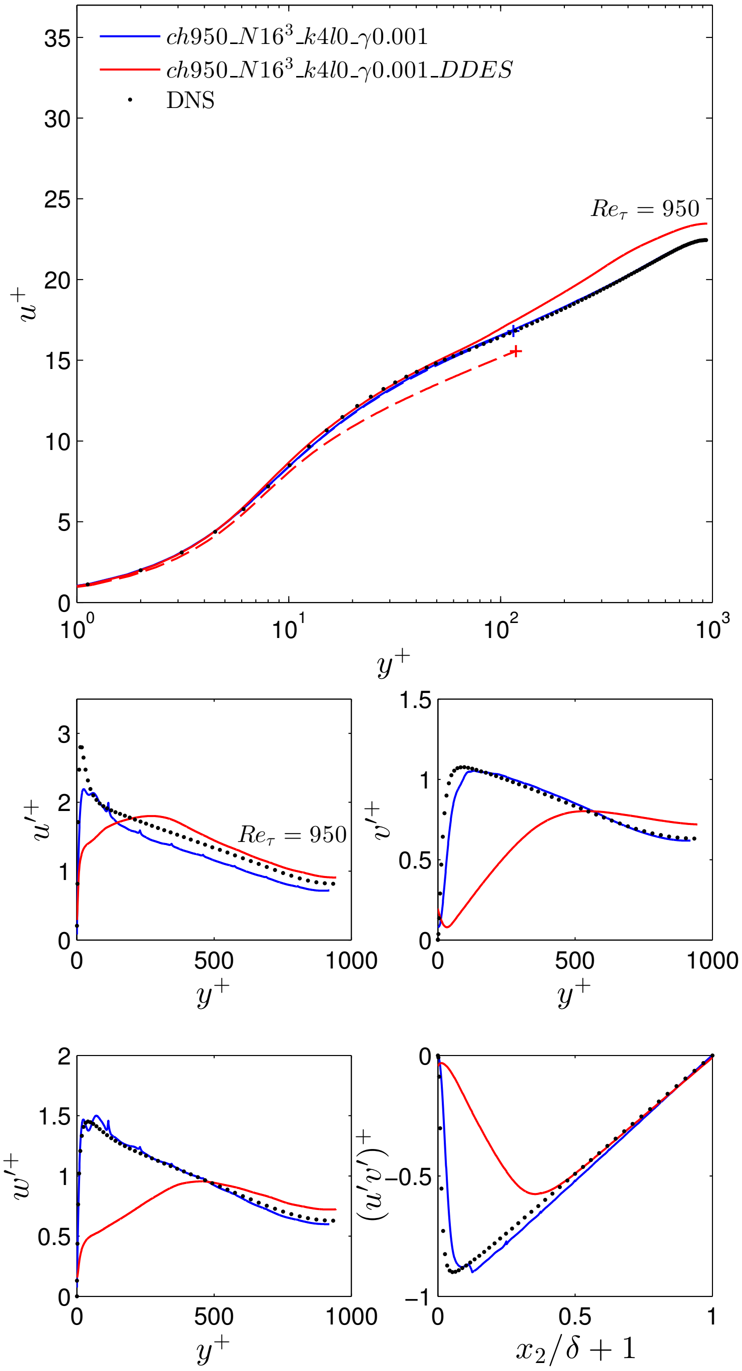


Figure 10.5.: MS-WMLES: Quantitative comparison of the present multiscale wall model with DDES using function enrichment (see Chapter 9) employing the same grid: Mean velocity (top) and RMS-velocities as well as RSS (bottom). All quantities are normalized according to $u^+ = \langle u_1 \rangle / u_\tau$, $u'^+ = \sqrt{\langle u_1'^2 \rangle} / u_\tau$, $v'^+ = \sqrt{\langle u_2^2 \rangle} / u_\tau$, $w'^+ = \sqrt{\langle u_3^2 \rangle} / u_\tau$, and $(u'v')^+ = \langle u_1 u_2 \rangle / u_\tau^2$. The full solution is displayed as solid line and the enrichment component as dashed line.

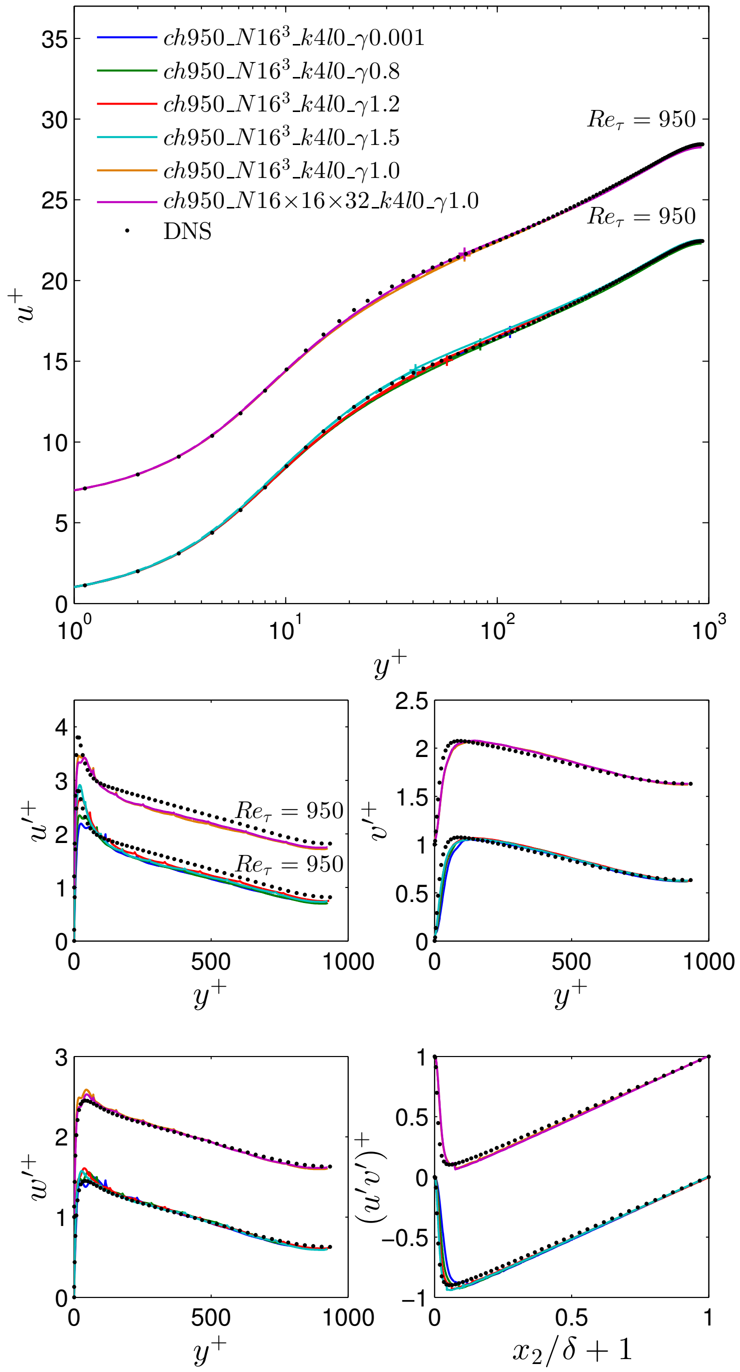


Figure 10.6.: MS-WMLES: Grid stretching and element aspect ratios: Mean velocity (top) and RMS-velocities as well as RSS (bottom). All quantities are normalized according to $u^+ = \langle u_1 \rangle / u_\tau$, $u'^+ = \sqrt{\langle u_1'^2 \rangle} / u_\tau$, $v'^+ = \sqrt{\langle u_2^2 \rangle} / u_\tau$, $w'^+ = \sqrt{\langle u_3^2 \rangle} / u_\tau$, and $(u'v')^+ = \langle u_1 u_2 \rangle / u_\tau^2$. The full solution is displayed as solid line and the enrichment component as dashed line.

velocity component as well, such that no vortices are resolved in the inner layer. This stands in contrast to the present multiscale wall model, which computes turbulent motions in the inner boundary layer as well. In DDES, turbulent eddies evolve outside of the inner layer, but the flow generally behaves differently due to the use of the LES eddy viscosity subgrid model given through the SA model in LES mode. These two simulations are further compared quantitatively through velocity statistics in Figure 10.5. As it is expected, DDES overpredicts the mean velocity in the outer layer as a result of a log-layer mismatch caused by the RANS–LES transition. Such a log-layer mismatch is not observed with the present model. The turbulent stresses show that the transition in the DDES model extends until approximately $y/\delta = 0.4$, whereas the present multiscale wall model makes much better use of the eddy-resolving capability of the mesh and resolves the important turbulent motions beyond $y/\delta = 0.05$.

Grid Independence. The wall model including its application range and robustness with regard to grid dependence is analyzed in a systematic manner in the following. In Figure 10.6, five grid stretching factors are investigated using the same number of cells in each spatial direction. In addition, the number of cells is doubled in the x_3-direction in one case. The results show that the wall model generally exhibits an excellent robustness regarding the choice of the mesh. The dependence of the cell aspect ratio may be analyzed by considering the measure $\Delta y_{1e}^+(k+1)/\max(\Delta x_{1e}^+, \Delta x_{2e}^+, \Delta x_{3e}^+)$, which quantifies the available number of grid points in the wall-parallel direction to resolve a turbulent motion of the size of Δy_{1e}^+; this quantity is included in Table 10.1 for all simulation cases. Remember that we require sufficient turbulence to be resolved at a distance from the wall Δy_{1e}^+ such that the RANS model can be 'switched off', so the cell aspect ratio should not exceed a certain limit. It is noted that other wall modeling approaches, such as wall-stress models, in principle have the same requirement if not even more stringent. The factor $\Delta y_{1e}^+(k+1)/\max(\Delta x_{1e}^+, \Delta x_{2e}^+, \Delta x_{3e}^+)$ varies in the range 0.56 to 1.60. The simulation case with the highest aspect ratio, $ch950\_N16^3\_k4l0\_\gamma1.5$, shows a minor overprediction of the mean velocity where the wall model ends, allowing the conclusion that $\Delta y_{1e}^+(k+1)/\max(\Delta x_{1e}^+, \Delta x_{2e}^+, \Delta x_{3e}^+)$ should not go below the limit ~ 0.5. According to Jiménez [127], the streamwise size of the energetic eddies in the log-layer is approximately $5y$ so they are well-resolved with 2.5 grid points at the height Δy_{1e}^+, where the RANS layer ends.

Application Rage. The coercivity analysis in Section 10.4.2 requires the clipping of the eddy viscosity beyond $\widehat{\nu + \nu_t} > 4\nu$ in the nonsymmetric volume term of the viscous operator. A normalization of this relation may be obtained by division with the viscosity $\widehat{\nu_t^+} = \widehat{\nu + \nu_t}/\nu > 4$. This relation is universal in y^+-units; compare the normalized eddy viscosity profiles in Figure 5.4. It is thus possible to investigate the clipping of $\widehat{\nu_t^+}$ for one flow configuration and to transfer the conclusions to other simulations. Given that the maximum values of $\widehat{\nu_t^+}$ grow with the width of the wall layer, it is sufficient to find the upper limit of the application range (cf. Equation (10.52)). In Figure 10.6, $\widehat{\nu_t^+}$ is clipped on approximately 11% of the quadrature points of all wall-layer cells for the case $ch950\_N16^3\_k4l0\_\gamma0.001$. Although the results are excellent for this case, we do not recommend the application of the wall model far beyond this y^+-range in order to guarantee the reliability of the results. In conclusion, the wall model should not extend beyond approximately 120 wall units in the statistical data. We stress at this point that the stability of the numerical method is guaranteed irrespective of the application of the wall model and this y^+-limit is only due to the result quality, which may degrade for thicker wall layers.

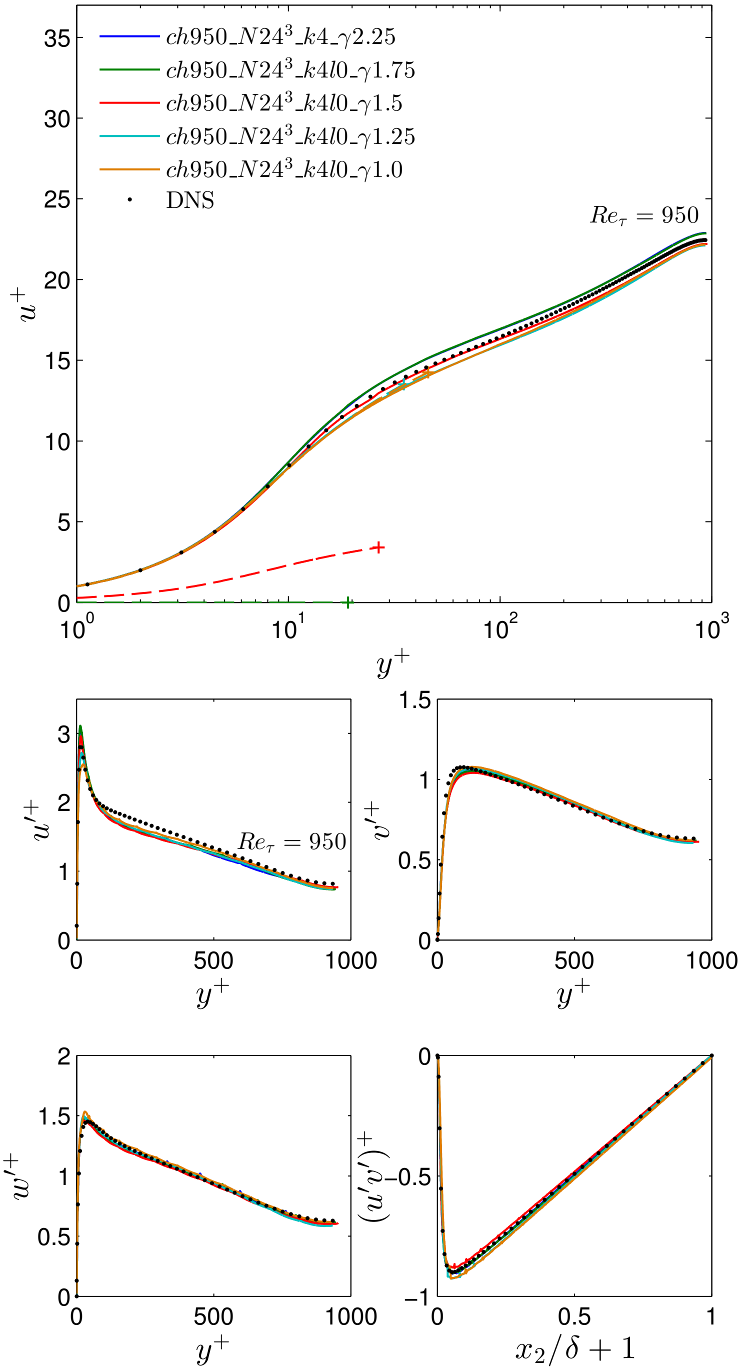


Figure 10.7.: MS-WMLES: Transition from wall-resolved to WMLES: Mean velocity (top) and RMS-velocities as well as RSS (bottom). All quantities are normalized according to $u^+ = \langle u_1\rangle/u_\tau$, $u'^+ = \sqrt{\langle u_1'^2\rangle}/u_\tau$, $v'^+ = \sqrt{\langle u_2^2\rangle}/u_\tau$, $w'^+ = \sqrt{\langle u_3^2\rangle}/u_\tau$, and $(u'v')^+ = \langle u_1u_2\rangle/u_\tau^2$. The full solution is displayed as solid line and the enrichment component as dashed line.

Transition WMLES–WRLES The transition from wall-modeled to wall-resolved LES is investigated in Figure 10.7. At least 24^3 cells are required for sufficiently resolving the energetic scales in the case without wall model, based on resolution guidelines by Chapman [42] and our experience with the present scheme (see Chapter 4, with $k+1$ grid points per cell, we have $\Delta y_1^+ \approx \Delta y_{1e}^+/(k+1) \approx 2$, $\Delta z_e^+ \approx \Delta z^+/(k+1) \approx 20$). In Figure 10.7, the grid stretching of the wall resolved LES is successively reduced until the wall model is fully active. The wall-resolved simulation slightly overpredicts the mean velocity and the turbulent fluctuations agree well with the reference. Regarding the case $ch950\_N24^3\_k4l0\_\gamma 1.75$, the wall model is taken into account in the algorithm, but the model is switched off during the entire simulation due to the condition described in Section 8.1. The results in Figure 10.7 are equivalent to the wall-resolved case. For the case $ch950\_N24^3\_k4l0\_\gamma 1.5$ the wall model is activated dynamically in a temporally varying fraction of the cells and the enrichment shape functions constitute a small part of the mean velocity. The mean velocity profile agrees well with the DNS, even better than the wall-resolved case. For smaller grid stretching factors, the wall model is fully switched on after the initial laminar-turbulent transition and the enrichment solution is equivalent to the mean velocity profile. The mean velocity in these simulation cases is slightly underpredicted compared to the reference data and the turbulent fluctuations agree well with the reference data. From this numerical test it may be concluded that the wall model is well capable of handling the full range from WRLES to WMLES.

Comparison $l = 0$ with $l = 1$. Until this point, all simulation cases have used a constant weighting of the enrichment function ($l = 0$) and we put forward reasons for this choice. Figure 10.8 compares two simulation cases, one with $l = 0$ and one with $l = 1$, where the latter represents a linear weighting of van Driest's law. The curves lie on top of each other, including the enrichment solution and the fluctuations. This suggests that, for attached boundary layers, there is no need for a weighting of the enrichment with more than a constant factor. Furthermore, the enrichment using linear functions requires additional DOFs (24 instead of 3 per cell) and, most importantly, results in approximately three times the number of linear iterations in the GMRES solver of the Helmholtz equation due to worse condition numbers. Regarding nonequilibrium boundary layers, a linear weighting may yield slightly better results due to the additional flexibility within the RANS solution, but the higher computational cost is not justified.

Performance Evaluation. We conclude the present section of the turbulent channel flow with a performance comparison of WRLES and WMLES. On the one hand, the wall model allows much coarser meshes near the wall, which also comes along with larger time steps due to the explicit time integration, and the lower grid anisotropy yields fewer iterations in the expensive pressure Poisson solver. On the other hand, the wall model requires additional computational effort due to the larger number of quadrature points in the enriched cells and supplementary algorithmic steps. Figure 10.9 compares the results of a wall-modeled case with the above coarse wall-resolved simulation as well as an additional finer wall-resolved case. The wall-modeled simulation exhibits better agreement with the DNS than the coarse wall-resolved simulation, but worse than the fine wall-resolved case. The computational cost is measured in number of processor cores times wall clock time of the entire simulation and the result is normalized by the wall-modeled simulation cost. The wall-modeled case ($ch950\_N12{\times}8{\times}12\_k4l0\_\gamma 1.6$) gives a cost of 1, the coarse LES simulation ($ch950\_N24^3\_k4\_\gamma 2.25$) yields a factor of 77, and the fine wall-resolved calculation ($ch950\_N32^3\_k4\_\gamma 2.0$) a cost of 158. Comparing the coarse wall-

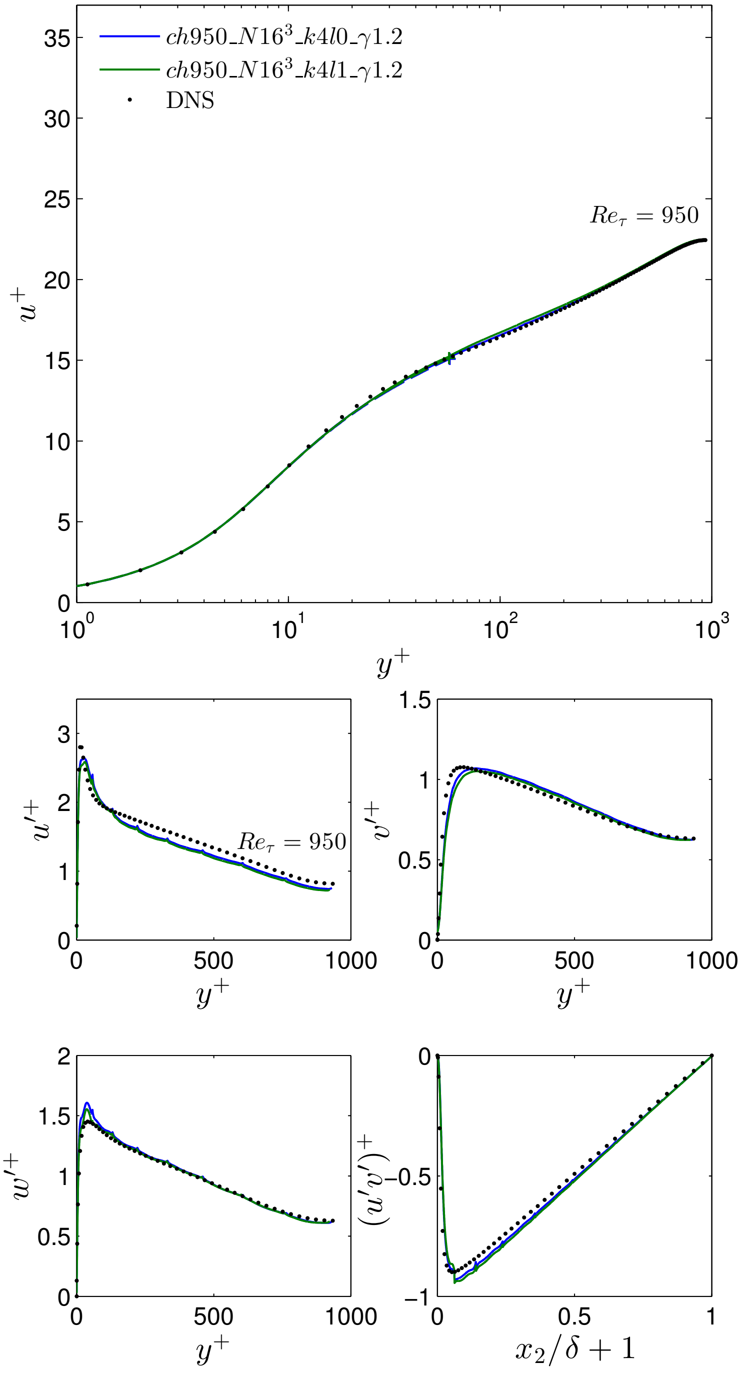


Figure 10.8.: MS-WMLES: Comparison of constant and linear shape functions for the weighting of the enrichment function: Mean velocity (top) and RMS-velocities as well as RSS (bottom). All quantities are normalized according to $u^+ = \langle u_1 \rangle / u_\tau$, $u'^+ = \sqrt{\langle u_1'^2 \rangle}/u_\tau$, $v'^+ = \sqrt{\langle u_2^2 \rangle}/u_\tau$, $w'^+ = \sqrt{\langle u_3^2 \rangle}/u_\tau$, and $(u'v')^+ = \langle u_1 u_2 \rangle / u_\tau^2$. The full solution is displayed as solid line and the enrichment component as dashed line.

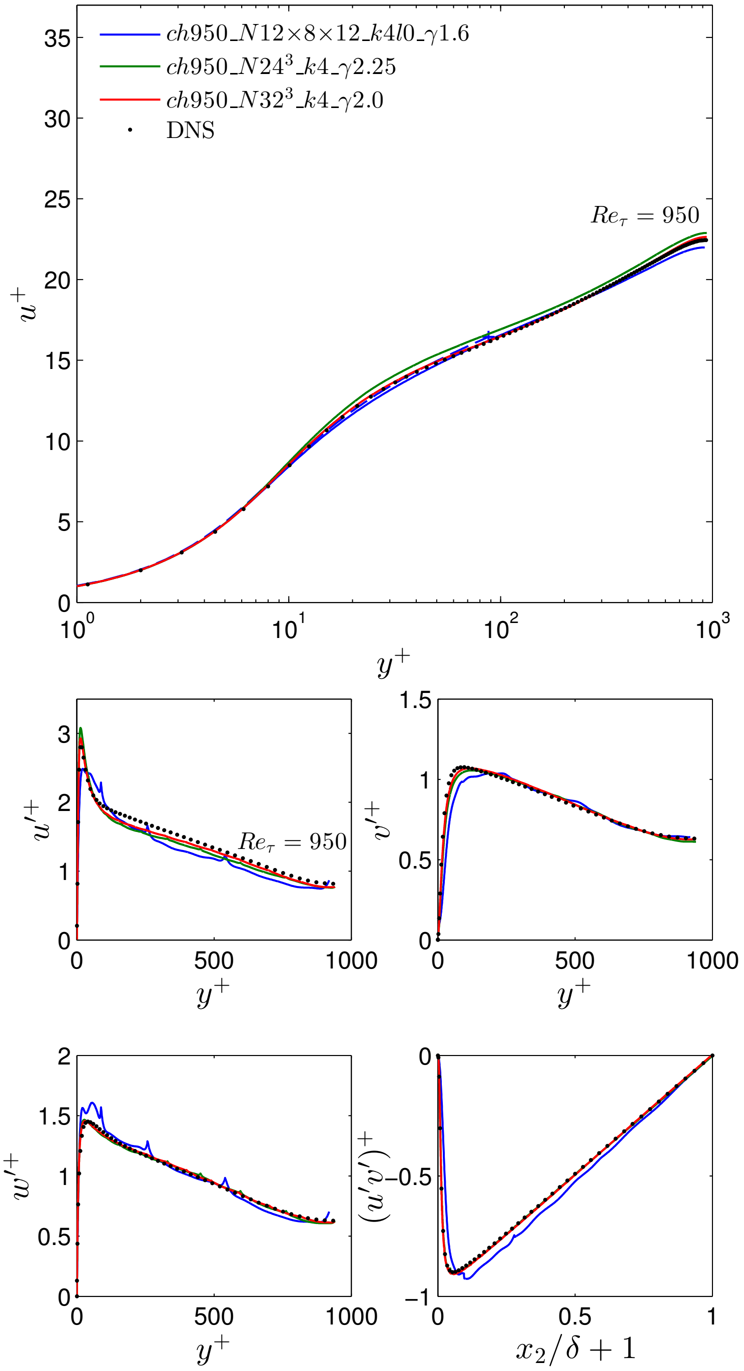


Figure 10.9.: MS-WMLES: Performance comparison of wall-resolved and wall-modeled cases: Mean velocity (top) and RMS-velocities as well as RSS (bottom). All quantities are normalized according to $u^+ = \langle u_1 \rangle / u_\tau$, $u'^+ = \sqrt{\langle u_1'^2 \rangle} / u_\tau$, $v'^+ = \sqrt{\langle u_2^2 \rangle} / u_\tau$, $w'^+ = \sqrt{\langle u_3^2 \rangle} / u_\tau$, and $(u'v')^+ = \langle u_1 u_2 \rangle / u_\tau^2$. The full solution is displayed as solid line and the enrichment component as dashed line. The wall-modeled simulation reduces the computational cost by a factor of 77 compared to the coarser and by a factor of 158 compared to the finer wall-resolved case.

resolved calculation and the wall-modeled calculation in more detail, the use of the wall model reduces the number of time steps by a factor of 5.7 (for the same simulation time) and reduces the number of cells by a factor of 12. In addition, the wall-resolved simulation requires approximately 22 Poisson solver iterations instead of 9 in the wall-modeled case to yield the same relative accuracy in the iterative solver due to the higher mesh stretching, which overcompensates the extra cost of the wall model. These results allow the conclusion that the wall model promises a speed-up compared to wall-resolved LES by a factor of approximately two orders of magnitude and we expect an even larger benefit for higher Reynolds numbers.

The multiscale wall model has been thoroughly investigated regarding its application limits, mesh dependence and performance. The major results are that the model can compute the whole range from wall-resolved to WMLES and is extremely robust with respect to different meshes as well as aspect ratios. However, the wall model should not be used beyond $y^+ = 120$ to guarantee accurate results due to a stability condition. Given this limitation, the wall model has accelerated the simulations of turbulent channel flow by a factor of approximately two orders of magnitude.

10.5.2. Flow over Periodic Hills

As a second benchmark example, we consider flow over periodic hills at the Reynolds numbers $\mathrm{Re}_H = 10{,}595$, $\mathrm{Re}_H = 19{,}000$, and $\mathrm{Re}_H = 37{,}000$ with the setup described in detail in Section 4.3. For most of the simulation cases, a mesh consisting of $32\times16\times16$ cells of degree $k = 4$ is employed, resulting in $160\times80\times80$ nodes, and the enrichment with $l = 0$ is included in the first off-wall element layer, as it is illustrated in Figure 10.10. At the highest Reynolds number, an additional simulation case is presented, in which the number of cells is doubled in each spatial direction. The grid is graded towards the walls and we investigate three grid stretching factors in order to show the influence of the width of the wall layer. According to Figure 10.11, the wall layer and thus the first off-wall cells span a range in y^+-units up to 86 for the lowest, 138 for the medium, and 156 for the highest Reynolds number, with peaks near the hill top and minima near the separation and reattachment points. The enrichment DOFs constitute equal or less than 0.1% of the overall number of DOFs. Further, the enrichment and thus the multiscale wall model are switched off dynamically if the boundary layer is sufficiently resolved. The wall model is therefore effectively not taken into account behind the hill crest and in the region of the flow reattachment. More cells are switched off in the lower Reynolds number cases; the active cells in one snapshot are depicted in Figure 10.10. The velocity field of the fully turbulent flow and turbulent vortex structures are visualized in Figure 10.12. All discretization cases are summarized in Table 10.2, including the labels employed in the subsequent figures. The investigations of the wall model include a comparison to one simulation for each Reynolds number, which uses the exact same mesh but without the wall model and is labeled with the addition NWM (no wall model) in Table 10.2.

We begin the discussion of the results with the friction and pressure coefficients for the lowest Reynolds number. They are defined in Equations (4.3) and (4.4) and the location of the reference pressure p_{ref} is chosen as $x_1 = 0$ at the upper wall. The curves are compared to the DNS reference data in Figure 10.13. Overall, good agreement of all curves with the DNS is observed and the largest error occurs near the hill top. Even the case without wall modeling is in acceptable agreement with the reference data, although the peak in the skin friction is predicted less distinct in that

Table 10.2.: MS-WMLES: Simulation cases and resolutions of the periodic hill flow. The cases at $\mathrm{Re}_H = 10{,}595$ and 19,000 use $32\times16\times16$ cells with varying grid stretching in vertical direction and, at $\mathrm{Re}_H = 37{,}000$, a finer mesh with $64\times32\times32$ grid cells is additionally considered. The polynomial degrees are $k = 4$ and $l = 0$ for all simulation cases, and the number of grid points per direction is $k + 1$ in each cell. The addition NWM stands for 'no wall model'. The separation and reattachment lengths $x_{1,\mathrm{sep}}$ and $x_{1,\mathrm{reatt}}$ correspond to the zero-crossings of the skin friction.

case	$N_{e1}\times N_{e2}\times N_{e3}$	$N_1\times N_2\times N_3$	Re_H	$\max(\Delta y_{1e}^+)$	$\frac{x_{1,\mathrm{sep}}}{H}$	$\frac{x_{1,\mathrm{reatt}}}{H}$
ph10595_stretch1	32×16×16	160×80×80	10,595	86	0.32	4.23
ph10595_stretch2	32×16×16	160×80×80	10,595	73	0.23	4.24
ph10595_stretch3	32×16×16	160×80×80	10,595	60	0.19	3.98
ph10595_stretch2_NWM	32×16×16	160×80×80	10,595	-	0.29	3.99
DNS 10595	-	896×448×448	10,595	-	0.19	4.51
ph19000_stretch1	32×16×16	160×80×80	19,000	138	0.42	2.85
ph19000_stretch2	32×16×16	160×80×80	19,000	113	0.35	3.60
ph19000_stretch3	32×16×16	160×80×80	19,000	95	0.25	3.64
ph19000_stretch2_NWM	32×16×16	160×80×80	19,000	-	0.32	1.99
RM_EXP [211]	-	-	19,000	-	-	3.94
ph37000_stretch3	32×16×16	160×80×80	37,000	156	0.37	2.78
ph37000_stretch3_fine	64×32×32	320×160×160	37,000	78	0.28	3.38
ph37000_stretch3_NWM	32×16×16	160×80×80	37,000	-	-	-
RM_EXP [211]	-	-	37,000	-	-	3.76
CM_WMLES [265]	-	128×64×64	37,000	-	-	2.3
CM_WMLES_fine [265]	-	256×128×128	37,000	-	-	2.8

case. Also, the separation and reattachment lengths in Table 10.2, given as the zero-crossings of the skin friction, are in good agreement with the reference. The reattachment lengths of the cases ph10595_stretch1 and ph10595_stretch2 at $\sim 4.25H$ are slightly underpredicted in comparison to the DNS reference of $4.51H$. Regarding the case ph10595_stretch3, the reattachment length is predicted even shorter as $3.98H$. With respect to the latter, a possible source of error could be the coarser mesh in the shear layer within the LES region, resulting in a more dissipative behavior of the numerical scheme and thus affecting the length of the recirulation zone.

The time-averaged streamwise velocity, vertical velocity, and RSS are depicted in Figure 10.14 at ten locations. The curves essentially lie on top of each other, including the wall-modeled simulations, the underresolved case, and the DNS. From this fact we conclude that the wall model shows an outstanding performance for nonequilibrium boundary layers, which confirms the results obtained for wall modeling via function enrichment presented in the previous chapters as well as the robustness of the method with respect to mesh aspect ratios observed in the previous section.

The fact that even the underresolved case without the wall model yields a good agreement with the reference data motivates an application of the same meshes to a higher Reynolds number. The mean streamwise velocity, vertical velocity, and RSS for $\mathrm{Re}_H = 19{,}000$ are shown in Figure 10.15. The wall model exhibits results of similar quality as for the lower Reynolds

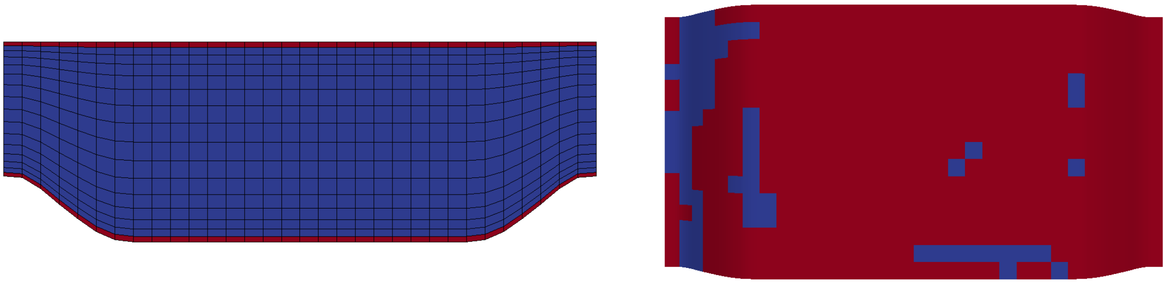


Figure 10.10.: MS-WMLES: Grid of the case ph10595_stretch2 (left). The wall-model cells are used in one layer near the wall. However, the enrichment is in these cells switched off dynamically if the polynomial is sufficient to resolve the turbulence. Therefore, the enrichment is in some cells at the lower wall inactive in the instantaneous flow field (right, computational domain shown from below). Enriched cells are depicted red and standard nonenriched cells blue.

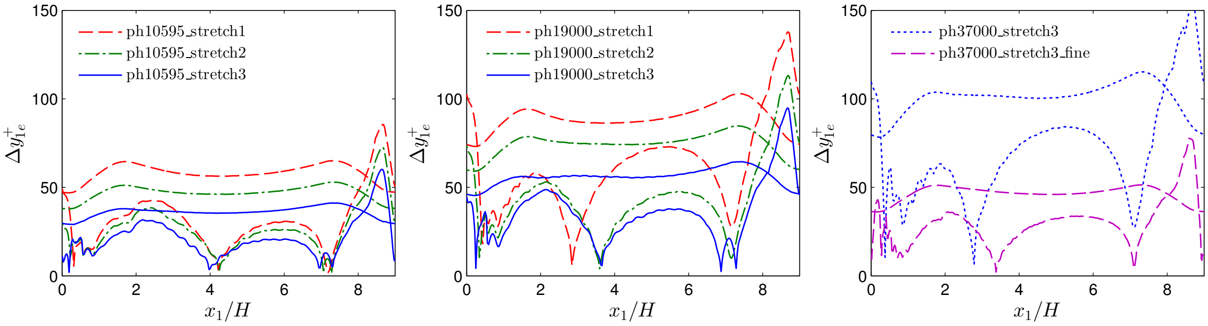


Figure 10.11.: MS-WMLES: Width of wall-layer (width of first off-wall cell) for $\mathrm{Re}_H = 10{,}595$, $\mathrm{Re}_H = 19{,}000$ (top right), and $\mathrm{Re}_H = 37{,}000$ (from left to right). The shallower curves correspond to the upper wall.

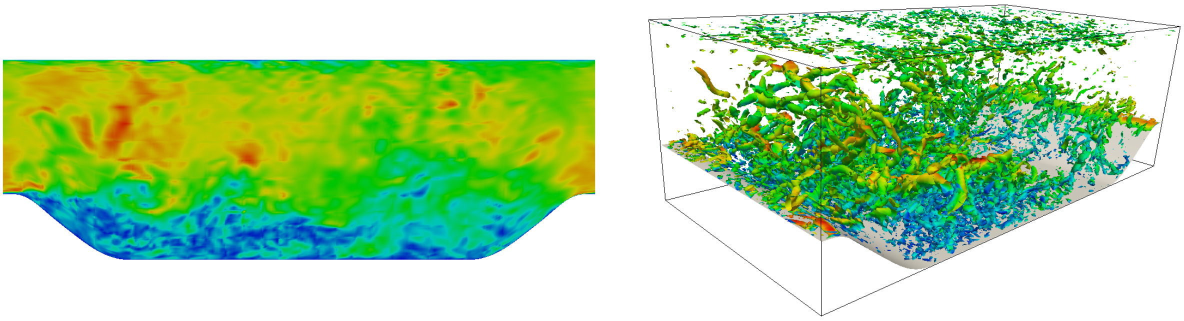


Figure 10.12.: MS-WMLES: Velocity magnitude (left) and visualization of turbulent eddies via the Q-criterion colored by velocity magnitude (right) of the case ph10595_stretch2. Red indicates high and blue low values.

number and the sensitivity regarding the grid stretching is slightly higher. However, the results for the case without wall model do not agree with the reference data at all. In particular, the reattachment length with $1.99H$ instead of $3.94H$ is substantially underpredicted.

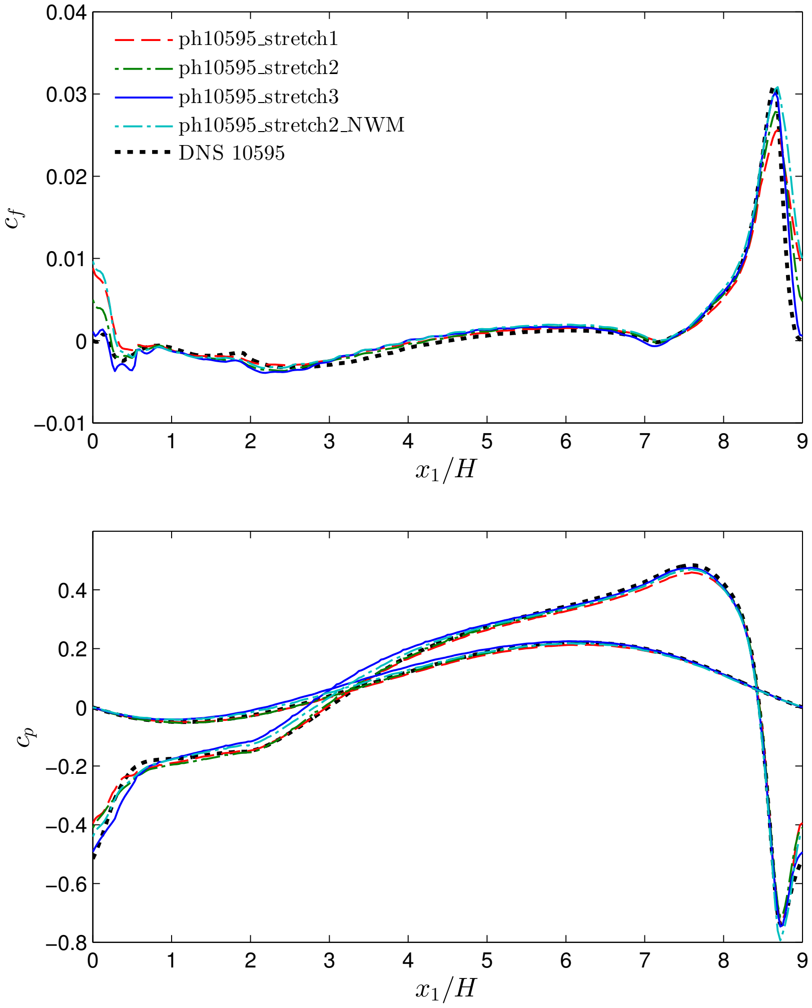


Figure 10.13.: MS-WMLES: Skin friction coefficient at the lower wall (top) and pressure coefficient at the lower and upper boundary (bottom). The shallower pressure coefficient curves correspond to the upper wall.

Finally, we assess whether the wall model gives any added value in separated flow conditions in comparison to the simpler wall-stress model by Carton de Wiart and Murman within a high-order DG method [265]. Two meshes are employed, the grid with the highest mesh stretching considered previously and a refined variant with twice as many cells in each spatial direction. These meshes are quite similar to the baseline and fine grid used by Wiart and Murman [265], labeled as CM_WMLES and CM_WMLES_fine, respectively. Slightly more points are used herein, the approach in the reference [265] exhibits an order of accuracy of 8 in comparison to the 4th order accuracy of the present method, however. A full data set including reference data is only available for the mean streamwise velocity, which is presented in Figure 10.16. The coarser case using wall modeling via function enrichment shows slightly better solutions than the coarse case with the equilibrium wall model, in particular with respect to the reattachment length. A possible explanation for this agreement may be that some of the relevant scales in the separation region

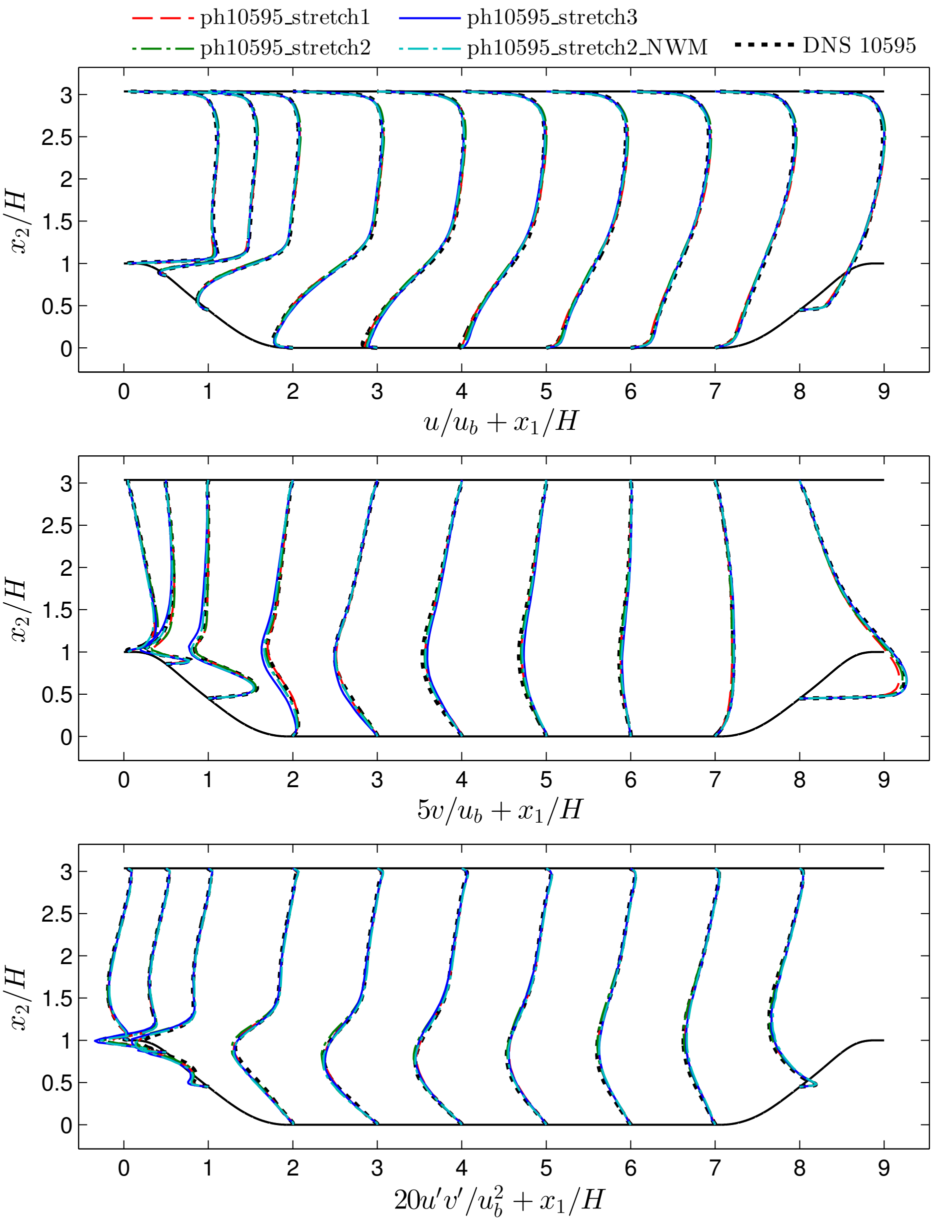


Figure 10.14.: MS-WMLES: Streamwise $u = \langle u_1 \rangle$ and vertical $v = \langle u_2 \rangle$ mean velocity, as well as RSS $u'v' = \langle u_1 u_2 \rangle - \langle u_1 \rangle \langle u_2 \rangle$ of the periodic hill flow at $\mathrm{Re}_H = 10{,}595$.

may not be sufficiently resolved, which would also explain the delayed separation length of the case ph37000_stretch3 of $0.37H$. Comparing the refined simulation cases, the equilibrium model does not show a substantial improvement in comparison to the coarse cases while the results of the present wall model are almost converged to the experimental reference data. An investigation of the reattachment lengths confirms these observations, as the simulation ph37000_stretch3_fine shows the best agreement considering this quantity. This superiority of the function-enrichment-based wall model in separated flow conditions is due to the full consistency of the method and

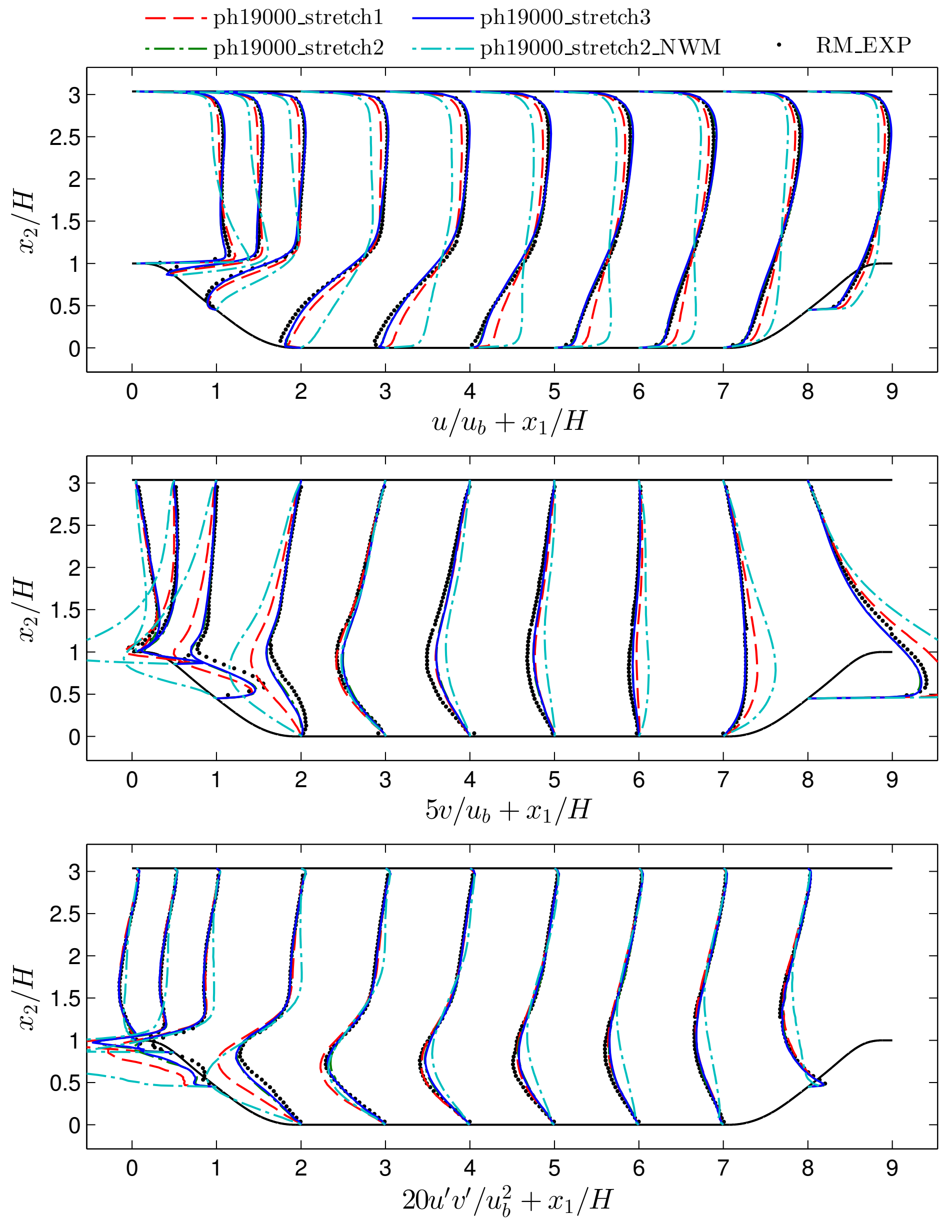


Figure 10.15.: MS-WMLES: Streamwise $u = \langle u_1 \rangle$ and vertical $v = \langle u_2 \rangle$ mean velocity, as well as RSS $u'v' = \langle u_1 u_2 \rangle - \langle u_1 \rangle \langle u_2 \rangle$ of the periodic hill flow at $\mathrm{Re}_H = 19{,}000$.

the consideration of all terms of the Navier–Stokes equations. The case without wall modeling does not show a separation region at all as the near-wall region is significantly underresolved.

In summary, the present multiscale wall model enables an accurate computation of separated boundary layers, is robust with regard to mesh sensitivities and exhibits considerably more accurate results compared to an equilibrium wall-stress model.

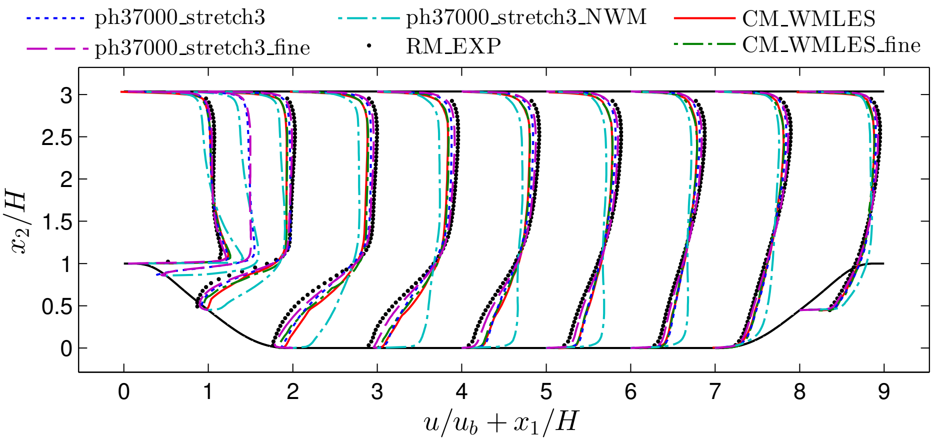


Figure 10.16.: MS-WMLES: Streamwise $u = \langle u_1 \rangle$ mean velocity of the periodic hill flow at $\mathrm{Re}_H = 37{,}000$.

10.6. Summary

In this chapter, we have developed a new approach to turbulence modeling within wall modeling via function enrichment. Based on a rigorous derivation of the modeling terms using Germano's framework of additive filtering, a RANS model is applied in a thin layer near the wall. Unlike existing wall modeling approaches, the RANS and an underresolved LES solution overlap inside the near-wall layer. This composition allows the LES solution to develop within the wall layer such that the typical issue of the RANS–LES transition is avoided. As the method uses wall modeling via function enrichment for the spatial discretization, the model is also capable of resolving the velocity gradient and thus the wall shear stress in the viscous sublayer with coarse meshes, where the first off-wall cell extends up to 120 wall units. The wall model has exhibited outstanding characteristics in attached and separated boundary layers and does not show a log-layer mismatch in contrast to the DDES model presented in the preceding chapter. The model has reduced the cost of a benchmark simulation by approximately two orders of magnitude in comparison to wall-resolved LES. This turbulence modeling approach is also considered as more consistent than the one used in Chapter 7 for WMLES.

11

Summary and Outlook

Summary. The research presented in this thesis makes major contributions to the next generation of CFD tools by increasing the ratio of accuracy and computational cost of the numerical simulations. The work has been presented in two parts. The first part further develops the state of the art in high-order DG methods for incompressible flow problems. The second part introduces a novel approach to wall modeling for Galerkin methods. The achievements may be summarized as follows:

In Part I of this thesis, a semi-explicit high-order DG solver for incompressible flows based on the dual-splitting scheme has been presented. The formulation of the spatial discretization has been extended in order to obtain a stable method in the limit of spatial underresolution and small time steps. This flow solver has been applied to DNS and ILES of two widely used benchmark examples, turbulent channel flow and flow over periodic hills. The numerical method is particularly suited for DNS due to its spectral capabilities, which yield low numerical dissipation properties and a high resolution power at high polynomial degrees, in addition to its excellent parallel scaling characteristics. Therefore, the code has been used to compute new reference data via DNS for the example of flow over periodic hills at two Reynolds numbers, which has been made available to the scientific community on a public repository [148]. The numerical method also shows very good results in ILES, where the numerical dissipation of the computational method accurately models the turbulent subgrid motions. The solver has been extended to the RANS equations based on the Spalart–Allmaras model and this algorithm has been applied to DDES of the previously mentioned examples.

Part II extends these numerical methods by a novel approach to computing turbulent boundary layer flows. The challenge of such flows is twofold. On the one hand, the mean velocity gradient becomes very high in the viscous sublayer at high Reynolds numbers. On the other hand, the flow is governed by a multitude of spatial and temporal scales. The accurate consideration of both of these problems in the numerical method is essential. The new wall modeling approach developed in this work primarily addresses the issue of the high gradient, but the approach can be used to formulate an elegant model for the unresolved turbulence as well. The wall model is based on the fundamental idea of Galerkin methods that the user chooses the shape functions, and the numerical method automatically computes the optimal solution with these given shape functions. By including a wall function in the function space of the near-wall cells, the method is *a priory* capable of resolving the sharp velocity gradient in the viscous sublayer. With such a function space available, the full incompressible Navier–Stokes equations are solved in the whole boundary layer such that all terms are satisfied discretely, including the convective term and the contribution of the pressure gradient. Since the polynomial shape functions are retained, the method can represent more general boundary layer profiles as well, such as in separated flow conditions. The wall model has been implemented in a high-order DG framework and a standard FEM solver.

The unresolved turbulence may be taken into account in the wall model in several ways. We have shown that the classical RANS methodology may be used, so the wall model can be included in a RANS solver in order to reduce the resolution requirements in the near-wall region. The presented numerical examples show that the first off-wall grid point may be placed at up to approximately $y_1^+ = 1{,}000$ wall units. This RANS solver may also be extended to DES in a straight-forward manner, where the Spalart–Allmaras model accurately models the unresolved turbulent scales in the near-wall region, while the model is used as an LES subgrid closure further away from the wall, where the energetic turbulent eddies are explicitly computed. In addition, a third approach has been developed, which interprets the function space as two components of the solution: the high-order polynomial component computes turbulent eddies via LES and the additional enrichment shape functions represent the unresolved part of the flow in an averaged sense. This particular interpretation and the variational multiscale method allow for the application of tailored turbulence models for each of these scales, a RANS model for the RANS solution component and an LES model for the LES solution component, and both solutions overlap inside the near-wall layer. Through this construction, the hybrid RANS/LES approach does not exhibit a log-layer mismatch by definition. The method has yielded a speed-up in comparison to WRLES by approximately two orders of magnitude. All applications of the wall model with different turbulence closures showed a high accuracy in attached boundary layers and a high level of flexibility in separated flows such as the periodic hill flow. The wall modeled simulations could reproduce the results of wall-resolved simulations with much coarser meshes.

Outlook. The work presented in this thesis is intended to lay the basis for further research in the fields of high-order DG methods and wall modeling. Several improvements of the high-order DG solver are currently under development or may be considered in future work:

- For the velocity and pressure variables, equal-order function spaces are used throughout this thesis, which is possible since the dual-splitting scheme enables stable computations with such a setup. A recent analysis of the topic [72] concludes that it is still beneficial to employ mixed-order elements, where the pressure shape functions are of one degree lower than the velocity shape functions. Additional enhancements of the numerical scheme are also discussed in that publication.

- The high-order DG scheme presented in this work has been stabilized by a divergence-penalty term, which reduces the divergence error inside the elements. Alternative measures should be considered that reduce the divergence error and they should be compared to the present method regarding their efficiency and accuracy (see, e.g., [171]).

- In Chapter 4, the subgrid turbulence was accounted for by the numerical dissipation of the scheme. While this approach yielded good results in these simulations, alternative explicit subgrid models may be investigated in order to allow for a critical comparison.

- DNS reference data for the periodic hill flow has been be computed up to $Re_H = 10{,}595$ in this work. A DNS of the next higher Reynolds number of $Re_H = 19{,}000$ is feasible on the generation of supercomputers, which is currently installed, with a similar simulation setup as used for the case $Re_H = 10{,}595$. A mesh of approximately $256\times128\times128$ cells with a polynomial degree of $k = 5$ in each cell and 900 million grid points, as well as 13 million time steps would be necessary. This reference data would be highly valuable for the assessment of wall models.

- The RANS and DES solver based on a semi-explicit approach requires many time steps in relation to the temporal scales of unsteady RANS simulations. A fully implicit formulation of the solver may be considered in future work. As a further development of the RANS solver presented, the SA equation may be implemented in an implicit step as part of the viscous solver of the dual-splitting scheme.

The wall modeling approach presented in Part II of this thesis should be seen as the starting point for a further development towards an application of the method in industrial simulation programs. The most important open topics are the following:

- Since RANS is still the state of the art in the industry, wall modeling via function enrichment should be developed further for this application. In particular, the wall model should be included in a steady-state solver and a fully implicit solver in order to avoid a dependency of the time step size of the diffusion number and to allow larger time step sizes of the order of the time scales of unsteady RANS. It may also be of interest to consider DES in the context of a fully implicit solver instead of the present semi-explicit approach.

- A fully implicit solver necessitates the development of efficient preconditioning strategies, such as multigrid preconditioners. To this end, one possibility of treating the enrichment would be the use of a p-multigrid step, in which the enrichment is included in the polynomial part. Subsequently, standard h-multigrid methods could be used.

- The topic of the numerical quadrature of the weak forms including the nonpolynomial enrichment shape functions is one of the aspects with the largest potential for the further development of the solvers presented in this work. The most promising approach seems to be an adaptive method that takes into account the locality of the high-gradient region near the viscous sublayer and uses less quadrature points further away from the wall.

- Wall modeling via function enrichment has been introduced in this work based on the XFEM framework. There exist several alternative numerical methods that enable the construction of function spaces based on enrichment functions (see Section 6.3.1). It would be interesting to extend the wall model to other enrichment methodologies in order to enable a critical comparison of several approaches with respect to their efficiency.

- Furthermore, wall modeling via function enrichment should be extended to the compressible Navier–Stokes equations.

- Wall modeling via function enrichment may also be extended to other RANS turbulence models in a straightforward manner, if the variables considered in additional transport equations vary smoothly inside the inner boundary layer, such that the standard polynomial component is sufficient to discretize these variables.

- The WMLES method presented in Chapter 10 would benefit from further development by mathematicians in order to remove the limit of application discussed in that chapter.

- The numerical examples presented in this work are canonical flows that allow a quantitative assessment of the model with relatively low computational effort. The wall model should be validated with additional industry-oriented examples with complex geometries that show the applicability of the wall model to real-size industrial problems.
- Finally, the wall model may be considered in multiphysics scenarios. One of the motivations for developing the present wall models for eddy-resolving CFD is an application of the WMLES solution to aeroacoustics and fluid-structure interaction, which are both relevant for example in the automotive industry.

A

Evaluation of Enrichment Functions and Spatial Derivatives

The implementation of wall modeling via function enrichment uses the following algorithms and equations. They describe the numerical evaluation of two wall functions and the computation of their spatial derivatives with respect to the Cartesian coordinates, which are necessary for the evaluation of weak forms.

A.1. Numerical Evaluation of Spalding's Law

There are many ways to evaluate Spalding's implicitly given wall function (Equation (6.7)). In this work, Newton's method is used, where the residual is defined as

$$\mathcal{F}_j = -y_h^+ + \frac{\psi_j}{\kappa} + e^{-\kappa B}\left(e^{\psi_j} - 1 - \psi_j - \frac{\psi_j^2}{2!} - \frac{\psi_j^3}{3!} - \frac{\psi_j^4}{4!}\right) \tag{A.1}$$

with j indicating the iteration number. As initial condition, we use the explicit expressions for the viscous sublayer and the logarithmic region:

$$\psi_0 = \begin{cases} y_h^+ \kappa, & \text{if } y_h^+ < 11, \\ \ln(y_h^+) + \kappa B, & \text{if } y_h^+ \geq 11. \end{cases} \tag{A.2}$$

We iterate until either the residual $|\mathcal{F}_j|$ or the increment $|\Delta\psi_{j+1}|$ with $\Delta\psi_{j+1} = -\frac{\mathcal{F}_j}{\mathrm{d}\mathcal{F}_j/\mathrm{d}\psi_j}$ reach an accuracy of 10^{-14}, which takes approximately three to eight iterations depending on y_h^+. Step size limiting, line search and other modifications have not been considered.

A.2. Numerical Evaluation of van Driest's Law

The enrichment function ψ given as van Driest's wall-law in Equation (6.8) has to be computed on each quadrature point during evaluation of the weak forms. The wall function necessitates the evaluation of an integral in the interval $[0, y^+]$. In order to avoid computing the integral over the entire interval each time the wall function is evaluated, we tabulate the wall function ψ_i at several discrete values of the wall coordinate y_i^+. When the integral is evaluated at run time, solely a small slice of the interval has to be computed, namely $[y_i^+, y^+]$ with i such that $y_i^+ \leq y^+ < y_{i+1}^+$:

$$\psi = \psi_i + \int_{y_i^+}^{y^+} \frac{2\,\mathrm{d}y^+}{1 + \sqrt{1 + \left(2\kappa y^+ (1 - e^{(-y^+/A^+)})\right)^2}} \tag{A.3}$$

Table A.1.: Location of support points y_i^+ and corresponding values ψ_i for the efficient evaluation of the enrichment function.

i	y_i^+	ψ_i
0	0.0	0.0
1	5.0	4.88298776233176
2	11.0	8.91824406645381
3	24.0	12.3978516813118
4	59.0	15.1875389926298
5	144.0	17.4177125619900
6	361.0	19.6484300823042
7	946.0	21.9930107788854
8	2517.0	24.3778307011372

This remaining integral is evaluated employing a nine-point Gaussian quadrature rule (with a polynomial order of accuracy of 17) irrespective of the width of the interval in our code. The fixed selection allows for efficient SIMD instructions that include quadrature points in several elements, each with different parameters. The tabulated quantities have been computed using 32 digit precision in Matlab and are listed in Table A.1 for $\kappa = 0.41$ and $A^+ = 26$ and yield a guaranteed relative accuracy for ψ of 10^{-12} up to $y^+ = 5{,}000$. The rapid change of the y_i^+ values shows that the near-wall area necessitates a denser distribution of the support points, while the integrand varies more slowly beyond $y^+ > 1{,}000$, such that only a few quadrature nodes are sufficient. As the wall function is only computed once per time step on each quadrature point and stored, the evaluation cost is negligible. Aiming at fine-tuning the approach, one may consider to use more support points y_i^+ and a Gaussian rule with fewer support points.

Besides the enrichment function itself, the spatial derivative $\mathrm{d}\psi/\mathrm{d}y^+$ is also required for evaluating the weak forms. This derivative is given by the integrand in Equation (A.3).

It is noted that a scaling of the enrichment function with a factor of 0.1 is considered in the applications of the DG code, which yields better conditioned systems by scaling the enrichment shape functions to a similar value range as the standard polynomial shape functions.

A.3. Derivatives of the Enrichment in Cartesian Coordinates

The first and second derivatives of the enrichment with respect to Cartesian coordinates, which are required for the evaluation of Galerkin formulations with enriched function spaces, are obtained by applying the chain rule iteratively starting from Equation (6.5) for the DG case and Equation (6.6) for the standard FEM case. Since the standard FEM enrichment is more complex it is detailed here. Regarding DG, the ramp function may simply be set to $r_h = 1$ such that all derivatives are canceled. Likewise, the nodal values of the enrichment function considered below, $\psi(\boldsymbol{x}_B, t)$, are canceled. The equations are split into three groups: (i.) expressions in Cartesian coordinates, (ii.) transformation to the wall coordinate y^+ and (iii.) derivative with respect to y^+. Indices define space dimensions $i, j \in \{1, 2, 3\}$.

i. The first derivative is

$$\begin{aligned}
&\frac{\partial}{\partial x_i}\widetilde{\boldsymbol{u}}_h(\boldsymbol{x},t) = \\
&\sum_{B\in N^l}\Big(\frac{\partial N_B^l(\boldsymbol{x})}{\partial x_i}(\psi(\boldsymbol{x},t)-\psi(\boldsymbol{x}_B,t))r_h(\boldsymbol{x}) + N_B^l(\boldsymbol{x})\frac{\partial \psi(\boldsymbol{x},t)}{\partial x_i}r_h(\boldsymbol{x}) \\
&+ N_B^l(\boldsymbol{x})(\psi(\boldsymbol{x},t)-\psi(\boldsymbol{x}_B,t))\frac{\partial r_h(\boldsymbol{x})}{\partial x_i}\Big)\widetilde{\boldsymbol{u}}_B(t)
\end{aligned} \tag{A.4}$$

and the second derivative gives

$$\begin{aligned}
&\frac{\partial^2}{\partial x_i x_j}\widetilde{\boldsymbol{u}}_h(\boldsymbol{x},t) = \\
&\sum_{B\in N^l}\Big(\frac{\partial^2 N_B^l(\boldsymbol{x})}{\partial x_i \partial x_j}(\psi(\boldsymbol{x},t)-\psi(\boldsymbol{x}_B,t))r_h(\boldsymbol{x}) + \frac{\partial N_B^l(\boldsymbol{x})}{\partial x_i}\frac{\partial \psi(\boldsymbol{x},t)}{\partial x_j}r_h(\boldsymbol{x}) \\
&+ \frac{\partial N_B^l(\boldsymbol{x})}{\partial x_i}(\psi(\boldsymbol{x},t)-\psi(\boldsymbol{x}_B,t))\frac{\partial r_h(\boldsymbol{x})}{\partial x_j} + \frac{\partial N_B^l(\boldsymbol{x})}{\partial x_j}\frac{\partial \psi(\boldsymbol{x},t)}{\partial x_i}r_h(\boldsymbol{x}) \\
&+ N_B^l(\boldsymbol{x})\frac{\partial^2 \psi(\boldsymbol{x},t)}{\partial x_i \partial x_j}r_h(\boldsymbol{x}) + N_B^l(\boldsymbol{x})\frac{\partial \psi(\boldsymbol{x},t)}{\partial x_i}\frac{\partial r_h(\boldsymbol{x})}{\partial x_j} \\
&+ \frac{\partial N_B^l(\boldsymbol{x})}{\partial x_j}(\psi(\boldsymbol{x},t)-\psi(\boldsymbol{x}_B,t))\frac{\partial r_h(\boldsymbol{x})}{\partial x_i} + N_B^l(\boldsymbol{x})\frac{\psi(\boldsymbol{x},t)}{\partial x_j}\frac{\partial r_h(\boldsymbol{x})}{\partial x_i} \\
&+ N_B^l(\boldsymbol{x})(\psi(\boldsymbol{x},t)-\psi(\boldsymbol{x}_B,t))\frac{\partial^2 r_h(\boldsymbol{x})}{\partial x_i \partial x_j}\Big)\widetilde{\boldsymbol{u}}_B(t).
\end{aligned} \tag{A.5}$$

The ramp function is defined node-wise and interpolated with the standard linear FE expansion, allowing for the straight-forward computation of its derivatives.

ii. As the enrichment function is defined via the wall coordinate, its derivatives are transformed to y^+ yielding

$$\frac{\partial \psi(\boldsymbol{x},t)}{\partial x_i} = \frac{\mathrm{d}\psi}{\mathrm{d}y^+}\frac{\partial y^+}{\partial x_i} \tag{A.6}$$

with

$$\frac{\partial y^+}{\partial x_i} = \frac{\frac{\partial y_h}{\partial x_i}}{\nu}\sqrt{\frac{\tau_{w,h}}{\rho}} + \frac{y_h\frac{\partial \tau_{w,h}}{\partial x_i}}{2\nu\sqrt{\tau_{w,h}\rho}} \tag{A.7}$$

where $\frac{\partial y_h}{\partial x_i}$ and $\frac{\partial \tau_{w,h}}{\partial x_i}$ are obtained in a straight-forward manner via the standard linear FE expansion in Equation (6.13) and (6.12). Applying the chain rule successively, the second derivative becomes

$$\frac{\partial^2 \psi(\boldsymbol{x},t)}{\partial x_i \partial x_j} = \frac{\mathrm{d}^2\psi}{\mathrm{d}y^{+2}}\frac{\partial y^+}{\partial x_j}\frac{\partial y^+}{\partial x_i} + \frac{\mathrm{d}\psi}{\mathrm{d}y^+}\frac{\partial^2 y^+}{\partial x_i \partial x_j} \tag{A.8}$$

with

$$\frac{\partial^2 y^+}{\partial x_i \partial x_j} = \\ \frac{\frac{\partial^2 y_h}{\partial x_i \partial x_j}}{\nu}\sqrt{\frac{\tau_{w,h}}{\rho}} + \frac{\frac{\partial y_h}{\partial x_i}\frac{\partial \tau_{w,h}}{\partial x_j}}{2\nu\sqrt{\tau_{w,h}\rho}} + \frac{\frac{\partial y_h}{\partial x_j}\frac{\partial \tau_{w,h}}{\partial x_i}}{2\nu\sqrt{\tau_{w,h}\rho}} - \frac{y_h\frac{\partial \tau_{w,h}}{\partial x_i}\frac{\partial \tau_{w,h}}{\partial x_j}}{4\nu\sqrt{\rho}(\tau_{w,h})^{3/2}} + \frac{y_h\frac{\partial^2 \tau_{w,h}}{\partial x_i \partial x_j}}{2\nu\sqrt{\tau_{w,h}\rho}}. \tag{A.9}$$

iii. The derivatives of $\psi(\boldsymbol{x}, t)$ based on Spalding's law with respect to y^+ may be obtained explicitly with given ψ as

$$\frac{\mathrm{d}\psi}{\mathrm{d}y^+} = \frac{1}{\frac{1}{\kappa} + e^{-\kappa B}(e^{\psi} - 1 - \psi - \frac{\psi^2}{2!} - \frac{\psi^3}{3!})} \tag{A.10}$$

and

$$\frac{\mathrm{d}^2\psi}{\mathrm{d}y^{+2}} = -e^{-\kappa B}\left(e^{\psi} - 1 - \psi - \frac{\psi^2}{2!}\right)\left(\frac{\mathrm{d}\psi}{\mathrm{d}y^+}\right)^3. \tag{A.11}$$

The first spatial derivative of van Driest's law with respect to y^+ is given through the integrand of the definition of the wall function,

$$\frac{\mathrm{d}\psi}{\mathrm{d}y^+} = \frac{2}{1 + \sqrt{1 + (2\kappa y^+(1 - e^{(-y^+/A^+)}))^2}}. \tag{A.12}$$

The second derivative may be obtained by differentiation and application of the chain rule, but it is not required for the weak forms presented in this thesis.

Bibliography

[1] S. Abbas, A. Alizada, T.P. Fries, The XFEM for high-gradient solutions in convection-dominated problems, *International Journal for Numerical Methods in Engineering* **82**, 1044–1072, 2010.

[2] R. Abgrall, A. Krust, An adaptive enrichment algorithm for advection-dominated problems, *International Journal for Numerical Methods in Fluids* **72**, 359–374, 2013.

[3] M. Adams, M. Brezina, J. Hu, R. Tuminaro, Parallel multigrid smoothing: Polynomial versus Gauss–Seidel, *Journal of Computational Physics* **188**, 593–610, 2003.

[4] N. Ahmed, T. Chacón Rebollo, V. John, S. Rubino, A review of variational multiscale methods for the simulation of turbulent incompressible flows, *Archives of Computational Methods in Engineering* **24**, 115–164, 2017.

[5] M. Akbas, A. Linke, L.G. Rebholz, P.W. Schroeder, The analogue of grad-div stabilization in DG methods for incompressible flows: Limiting behavior and extension to tensor-product meshes, *Computer Methods in Applied Mechanics and Engineering* **341**, 917–938, 2018.

[6] J.C. Del Álamo, J. Jiménez, Spectra of the very large anisotropic scales in turbulent channels, *Physics of Fluids* **15**, L41–L44, 2003.

[7] S.R. Allmaras, F.T. Johnson, P.R. Spalart, Modifications and clarifications for the implementation of the Spalart–Allmaras turbulence model, in: *Seventh International Conference on Computational Fluid Dynamics (ICCFD7)*, Big Island, Hawaii, 2012.

[8] C. Altmann, A. Beck, A. Birkefeld, F. Hindenlang, M. Staudenmaier, G. Gassner, C.D. Munz, Discontinuous Galerkin for high performance computational fluid dynamics (hpcdg), in: W.E. Nagel, D.B. Kröner, M.M. Resch (eds.), *High Performance Computing in Science and Engineering '11*, Springer, 2012.

[9] D. Arndt, W. Bangerth, D. Davydov, T. Heister, L. Heltai, M. Kronbichler, M. Maier, J.P. Pelteret, B. Turcksin, D. Wells, The `deal.II` library, version 8.5, *Journal of Numerical Mathematics* **25**, 137–145, 2017.

[10] D.N. Arnold, An interior penalty finite element method with discontinuous elements, *SIAM Journal on Numerical Analysis* **19**, 742–760, 1982.

[11] J.S. Baggett, On the feasibility of merging LES with RANS for the near-wall region of attached turbulent flows, in: *Center for Turbulence Research, Annual Research Briefs*, 267–277, 1998.

[12] J.S. Baggett, J. Jiménez, A.G. Kravchenko, Resolution requirements in large-eddy simulations of shear flows, in: *Center for Turbulence Research, Annual Research Briefs*, 51–66, 1997.

[13] P. BALAKUMAR, G.I. PARK, B. PIERCE, DNS, LES, and wall-modeled LES of separating flow over periodic hills, in: *Center for Turbulence Research, Proceedings of the Summer Program*, 2014.

[14] E. BALARAS, C. BENOCCI, U. PIOMELLI, Two-layer approximate boundary conditions for large-eddy simulations, *AIAA Journal* **34**, 1111–1119, 1996.

[15] F. BASSI, L. BOTTI, A. COLOMBO, A. CRIVELLINI, A. GHIDONI, F. MASSA, On the development of an implicit high-order discontinuous Galerkin method for DNS and implicit LES of turbulent flows, *European Journal of Mechanics - B/Fluids* **55, Part 2**, 367–379, 2016.

[16] F. BASSI, A. CRIVELLINI, D.A. DI PIETRO, S. REBAY, An artificial compressibility flux for the discontinuous Galerkin solution of the incompressible Navier–Stokes equations, *Journal of Computational Physics* **218**, 794–815, 2006.

[17] F. BASSI, A. GHIDONI, A. PERBELLINI, S. REBAY, A. CRIVELLINI, N. FRANCHINA, M. SAVINI, A high-order discontinuous Galerkin solver for the incompressible RANS and $k-\omega$ turbulence model equations, *Computers & Fluids* **98**, 54–68, 2014.

[18] Y. BAZILEVS, C. MICHLER, V.M. CALO, T.J.R. HUGHES, Weak Dirichlet boundary conditions for wall-bounded turbulent flows, *Computer Methods in Applied Mechanics and Engineering* **196**, 4853–4862, 2007.

[19] A.D. BECK, T. BOLEMANN, D. FLAD, H. FRANK, G.J. GASSNER, F. HINDENLANG, C.D. MUNZ, High-order discontinuous Galerkin spectral element methods for transitional and turbulent flow simulations, *International Journal for Numerical Methods in Fluids* **76**, 522–548, 2014.

[20] A.D. BECK, G.J. GASSNER, T. BOLEMANN, H. FRANK, F. HINDENLANG, C.D. MUNZ, Underresolved turbulence simulations with stabilized high order discontinuous Galerkin methods, in: J. FRÖHLICH, H. KUERTEN, B.J. GEURTS, V. ARMENIO (eds.), *Direct and Large-Eddy Simulation IX*, Springer, 2015.

[21] T. BELYTSCHKO, T. BLACK, Elastic crack growth in finite elements with minimal remeshing, *International Journal for Numerical Methods in Engineering* **45**, 601–620, 1999.

[22] T. BELYTSCHKO, R. GRACIE, G. VENTURA, A review of extended/generalized finite element methods for material modeling, *Modelling and Simulation in Materials Science and Engineering* **17**, 043001, 2009.

[23] C. BENOCCI, A. PINELLI, The role of the forcing term in the large eddy simulation of equilibrium channel flow, *Engineering Turbulence Modeling and Experiments* **1**, 287–296, 1990.

[24] T. BLACHA, M.M. GREGERSEN, M. ISLAM, H. BENSLER, Application of the adjoint method for vehicle aerodynamic optimization, in: *SAE Technical Paper*, SAE International, 2016.

[25] P. BOCHEV, R.B. LEHOUCQ, On the finite element solution of the pure Neumann problem, *SIAM Review* **47**, 50–66, 2005.

[26] J. BODART, J. LARSSON, Wall-modeled large eddy simulation in complex geometries with application to high-lift devices, in: *Center for Turbulence Research, Annual Research Briefs*, 37–48, 2011.

[27] J. BODART, J. LARSSON, Sensor-based computation of transitional flows using wall-modeled large eddy simulation, in: *Center for Turbulence Research, Annual Research Briefs*, 229–240, 2012.

[28] T. BOLEMANN, A. BECK, D. FLAD, H. FRANK, V. MAYER, C.D. MUNZ, High-order discontinuous Galerkin schemes for large-eddy simulations of moderate Reynolds number flows, in: *IDIHOM: Industrialization of High-Order Methods – A Top-Down Approach*, Springer, 2015.

[29] R. BORKER, C. FARHAT, R. TEZAUR, A discontinuous Galerkin method with Lagrange multipliers for spatially-dependent advection-diffusion problems, *Computer Methods in Applied Mechanics and Engineering* **327**, 93–117, 2017.

[30] L. BOTTI, D.A. DI PIETRO, A pressure-correction scheme for convection-dominated incompressible flows with discontinuous velocity and continuous pressure, *Journal of Computational Physics* **230**, 572–585, 2011.

[31] M. BREUER, B. JAFFRÉZIC, K. ARORA, Hybrid LES–RANS technique based on a one-equation near-wall model, *Theoretical and Computational Fluid Dynamics* **22**, 157–187, 2008.

[32] M. BREUER, N. PELLER, CH. RAPP, M. MANHART, Flow over periodic hills – numerical and experimental study in a wide range of Reynolds numbers, *Computers & Fluids* **38**, 433–457, 2009.

[33] S. BROGNIEZ, C. FARHAT, E. HACHEM, A high-order discontinuous Galerkin method with Lagrange multipliers for advection-diffusion problems, *Computer Methods in Applied Mechanics and Engineering* **264**, 49–66, 2013.

[34] A.N. BROOKS, T.J.R. HUGHES, 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, 1982.

[35] N.K. BURGESS, D.J. MAVRIPLIS, Robust computation of turbulent flows using a discontinuous Galerkin method, *AIAA paper*, no. 2012-457, 2012.

[36] E. BURMAN, P. ZUNINO, Numerical approximation of large contrast problems with the unfitted Nitsche method, in: J. BLOWEY, M. JENSEN (eds.), *Frontiers in Numerical Analysis – Durham 2010. Lecture Notes in Computational Science and Engineering* **85**, pp. 227–282, Springer, 2011.

[37] C. BURSTEDDE, L.C. WILCOX, O. GHATTAS, `p4est`: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees, *SIAM Journal on Scientific Computing* **33**, 1103–1133, 2011.

[38] G.C. BURTON, W.J.A. DAHM, Multifractal subgrid-scale modeling for large-eddy simulation. I. Model development and a priori testing, *Physics of Fluids* **17**, 075111, 2005.

[39] G.C. BURTON, W.J.A. DAHM, Multifractal subgrid-scale modeling for large-eddy simulation. II. Backscatter limiting and a posteriori evaluation, *Physics of Fluids* **17**, 075112, 2005.

[40] K.M. CASE, F.J. DYSON, E.A. FRIEMAN, C.E. GROSCH, F.W. PERKINS, Numerical simulation of turbulence, Technical report AD-774 161, Stanford Research Institute, 1973.

[41] T. CHACÓN REBOLLO, M. GÓMEZ MÁRMOL, S. RUBINO, Numerical analysis of a finite element projection-based VMS turbulence model with wall laws, *Computer Methods in Applied Mechanics and Engineering* **285**, 379–405, 2015.

[42] D.R. CHAPMAN, Computational aerodynamics development and outlook, *AIAA Journal* **17**, 1293–1313, 1979.

[43] L. CHEN, W.N. EDELING, S.J. HULSHOFF, POD enriched boundary models and their optimal stabilisation, *International Journal for Numerical Methods in Fluids* **77**, 92–107, 2014.

[44] Z.L. CHEN, S. HICKEL, A. DEVESA, J. BERLAND, N.A. ADAMS, Wall modeling for implicit large-eddy simulation and immersed-interface methods, *Theoretical and Computational Fluid Dynamics* **28**, 1–21, 2014.

[45] H. CHOI, P. MOIN, Grid-point requirements for large eddy simulation: Chapmans estimates revisited, *Physics of Fluids* **24**, 011702, 2012.

[46] D. CHUNG, D.I. PULLIN, Large-eddy simulation and wall modelling of turbulent channel flow, *Journal of Fluid Mechanics* **631**, 281–309, 2009.

[47] F.H. CLAUSER, The turbulent boundary layer, *Advances in Applied Mechanics* **4**, 1–51, 1956.

[48] B. COCKBURN, G. KANSCHAT, D. SCHÖTZAU, A locally conservative LDG method for the incompressible Navier–Stokes equations, *Mathematics of Computation* **74**, 1067–1095, 2005.

[49] B. COCKBURN, G. KANSCHAT, D. SCHÖTZAU, An equal-order DG method for the incompressible Navier–Stokes equations, *Journal of Scientific Computing* **40**, 188–210, 2009.

[50] R. CODINA, Stabilized finite element approximation of transient incompressible flows using orthogonal subscales, *Computer Methods in Applied Mechanics and Engineering* **191**, 4295–4321, 2002.

[51] D. COLES, The law of the wake in the turbulent boundary layer, *Journal of Fluid Mechanics* **1**, 191–226, 1956.

[52] S.S. COLLIS, *Discontinuous Galerkin methods for turbulence simulation*, in: *Center for Turbulence Research, Proceedings of the Summer Program*, 2002.

[53] A. CRIVELLINI, V. D'ALESSANDRO, F. BASSI, A Spalart-Allmaras turbulence model implementation in a discontinuous Galerkin solver for incompressible flows, *Journal of Computational Physics* **241**, 388–415, 2013.

[54] A. CRIVELLINI, V. D'ALESSANDRO, F. BASSI, Assessment of a high-order discontinuous Galerkin method for incompressible three-dimensional Navier–Stokes equations: Benchmark results for the flow past a sphere up to Re = 500, *Computers & Fluids* **86**, 442–458, 2013.

[55] D.L. DARMOFAL, S.R. ALLMARAS, M. YANO, J. KUDO, An adaptive, higher-order discontinuous Galerkin finite element method for aerodynamics, *AIAA paper*, no. 2013-2871, 2013.

[56] P. DAVIDSON, *Turbulence: An introduction for scientists and engineers*, Oxford University Press, USA, 2015.

[57] D. DAVYDOV, T. GERASIMOV, J.P. PELTERET, P. STEINMANN, On the h-adaptive PUM and the hp -adaptive FEM approaches applied to PDEs in quantum mechanics, 2016, arxiv preprint arxiv:1612.02305.

[58] R.B. DEAN, A single formula for the complete velocity profile in a turbulent boundary layer, *Journal of Fluids Engineering* **98**, 723–726, 1976.

[59] R.B. DEAN, Reynolds number dependence of skin friction and other bulk flow variables in two-dimensional rectangular duct flow, *Journal of Fluids Engineering* **100**, 215–223, 1978.

[60] J.W. DEARDORFF, A numerical study of three-dimensional turbulent channel flow at large Reynolds numbers, *Journal of Fluid Mechanics* **41**, 453–480, 1970.

[61] M.O. DEVILLE, P.F. FISCHER, E.H. MUND, *High-order methods for incompressible fluid flow*, vol. 9, Cambridge University Press, 2002.

[62] L.T. DIOSADY, S.M. MURMAN, DNS of flows over periodic hills using a discontinuous Galerkin spectral-element method, *AIAA paper*, no. 2014-2784, 2014.

[63] E.R. VAN DRIEST, On turbulent flow near a wall, *Journal of the Aeronautical Sciences* **23**, 1007–1011, 1956.

[64] P.A. DURBIN, Limiters and wall treatments in applied turbulence modeling, *Fluid Dynamics Research* **41**, 012203, 2009.

[65] A. Ehrl, A. Popp, V. Gravemeier, W.A. Wall, A dual mortar approach for mesh tying within a variational multiscale method for incompressible flow, *International Journal for Numerical Methods in Fluids* **76**, 1–27, 2014.

[66] N. Emamy, *Numerical simulation of deformation of a droplet in a stationary electric field using DG*, Dissertation, Technische Universität Darmstadt, 2014.

[67] N. Emamy, F. Kummer, M. Mrosek, M. Karcher, M. Oberlack, Implicit-explicit and explicit projection schemes for the unsteady incompressible Navier–Stokes equations using a high-order DG method, *Computers & Fluids* **154**, 285–295, 2017.

[68] J.A. Escobar-Vargas, P.J. Diamessis, T. Sakai, A spectral quadrilateral multidomain penalty method model for high Reynolds number incompressible stratified flows, *International Journal for Numerical Methods in Fluids* **75**, 403–425, 2014.

[69] C. Farhat, I. Harari, L.P. Franca, The discontinuous enrichment method, *Computer Methods in Applied Mechanics and Engineering* **190**, 6455–6479, 2001.

[70] C. Farhat, I. Kalashnikova, R. Tezaur, A higher-order discontinuous enrichment method for the solution of high Péclet advection-diffusion problems on unstructured meshes, *International Journal for Numerical Methods in Engineering* **81**, 604–636, 2010.

[71] N. Fehn, *A discontinuous Galerkin approach for the unsteady incompressible Navier–Stokes equations*, Master's thesis, Technical University of Munich, 2015.

[72] N. Fehn, W.A. Wall, M. Kronbichler, On the stability of projection methods for the incompressible Navier–Stokes equations based on high-order discontinuous Galerkin discretizations, *Journal of Computational Physics* **351**, 392–421, 2017.

[73] E. Ferrer, *A high order discontinuous Galerkin–Fourier incompressible 3D Navier–Stokes solver with rotating sliding meshes for simulating cross-flow turbines*, Dissertation, University of Oxford, 2012.

[74] E. Ferrer, An interior penalty stabilised incompressible discontinuous Galerkin–Fourier solver for implicit large eddy simulations, *Journal of Computational Physics* **348**, 754–775, 2017.

[75] E. Ferrer, D. Moxey, R.H.J. Willden, S.J. Sherwin, Stability of projection methods for incompressible flows using high order pressure-velocity pairs of same degree: Continuous and discontinuous Galerkin formulations, *Communications in Computational Physics* **16**, 817–840, 2014.

[76] E. Ferrer, R.H.J. Willden, A high order discontinuous Galerkin finite element solver for the incompressible Navier–Stokes equations, *Computers & Fluids* **46**, 224–230, 2011.

[77] E. Ferrer, R.H.J. Willden, A high order discontinuous Galerkin–Fourier incompressible 3D Navier-Stokes solver with rotating sliding meshes, *Journal of Computational Physics* **231**, 7037–7056, 2012.

[78] P.F. FISCHER, Analysis and application of a parallel spectral element method for the solution of the Navier–Stokes equations, *Computer Methods in Applied Mechanics and Engineering* **80**, 483–491, 1990.

[79] P.F. FISCHER, H. LEE-WING, G.E. KARNIADAKIS, E.M. RØNOUIST, A.T. PATERA, Recent advances in parallel spectral element simulation of unsteady incompressible flows, *Computers & Structures* **30**, 217–231, 1988.

[80] P.F. FISCHER, A.T. PATERA, Parallel spectral element solution of the Stokes problem, *Journal of Computational Physics* **92**, 380–421, 1991.

[81] P.F. FISCHER, E.M. RØNQUIST, D. DEWEY, A.T. PATERA, Spectral element methods: Algorithms and architectures, in: *Proceedings of the First International Conference on Domain Decomposition Methods for Partial Differential Equations*, SIAM, Philadelphia, 1988.

[82] M. FORTIN, F. BREZZI, *Mixed and hybrid finite element methods*, Springer, 1991.

[83] L.C. FOUCARD, F.J. VERNEREY, An X-FEM-based numerical-asymptotic expansion for simulating a Stokes flow near a sharp corner, *International Journal for Numerical Methods in Engineering* **102**, 79–98, 2015.

[84] L.P. FRANCA, S.L. FREY, Stabilized finite element methods: II. The incompressible Navier–Stokes equations, *Computer Methods in Applied Mechanics and Engineering* **99**, 209–233, 1992.

[85] L.P. FRANCA, A.L. MADUREIRA, Element diameter free stability parameters for stabilized methods applied to fluids, *Computer Methods in Applied Mechanics and Engineering* **105**, 395–403, 1993.

[86] M. FRANCIOLINI, A. CRIVELLINI, A. NIGRO, On the efficiency of a matrix-free linearly implicit time integration strategy for high-order discontinuous Galerkin solutions of incompressible turbulent flows, *Computers & Fluids* **159**, 276–294, 2017.

[87] H.M. FRANK, C.D. MUNZ, Direct aeroacoustic simulation of acoustic feedback phenomena on a side-view mirror, *Journal of Sound and Vibration* **371**, 132–149, 2016.

[88] A. FRÈRE, C. CARTON DE WIART, K. HILLEWAERT, P. CHATELAIN, G. WINCKELMANS, Application of wall-models to discontinuous Galerkin LES, *Physics of Fluids* **29**, 085111, 2017.

[89] A. FRÈRE, K. HILLEWAERT, P. CHATELAIN, G. WINCKELMANS, High Reynolds number airfoil: From wall-resolved to wall-modeled LES, *Flow, Turbulence and Combustion* **101**, 457–476, 2018.

[90] A. FRÈRE, N.N. SØRENSEN, K. HILLEWAERT, G. WINCKELMANS, Discontinuous Galerkin methodology for large-eddy simulations of wind turbine airfoils, *Journal of Physics: Conference Series* **753**, 022037, 2016.

[91] T.P. FRIES, A corrected XFEM approximation without problems in blending elements, *International Journal for Numerical Methods in Engineering* **75**, 503–532, 2008.

[92] T.P. FRIES, T. BELYTSCHKO, The extended/generalized finite element method: An overview of the method and its applications, *International Journal for Numerical Methods in Engineering* **84**, 253–304, 2010.

[93] J. FRÖHLICH, C.P. MELLEN, W. RODI, L. TEMMERMAN, M.A. LESCHZINER, Highly resolved large-eddy simulation of separated flow in a channel with streamwise periodic constrictions, *Journal of Fluid Mechanics* **526**, 19–66, 2005.

[94] J. FRÖHLICH, D. VON TERZI, Hybrid LES/RANS methods for the simulation of turbulent flows, *Progress in Aerospace Sciences* **44**, 349–377, 2008.

[95] P. GAMNITZER, V. GRAVEMEIER, W.A. WALL, A mixed/hybrid Dirichlet formulation for wall-bounded flow problems including turbulent flow, *Computer Methods in Applied Mechanics and Engineering* **245**, 22–35, 2012.

[96] G. GASSNER, D.A. KOPRIVA, A comparison of the dispersion and dissipation errors of Gauss and Gauss–Lobatto discontinuous Galerkin spectral element methods, *SIAM Journal on Scientific Computing* **33**, 2560–2579, 2011.

[97] M. GERMANO, Properties of the hybrid RANS/LES filter, *Theoretical and Computational Fluid Dynamics* **17**, 225–231, 2004.

[98] F.X. GIRALDO, J.S. HESTHAVEN, T. WARBURTON, Nodal high-order discontinuous Galerkin methods for the spherical shallow water equations, *Journal of Computational Physics* **181**, 499–525, 2002.

[99] F.X. GIRALDO, M. RESTELLI, A study of spectral element and discontinuous Galerkin methods for the Navier–Stokes equations in nonhydrostatic mesoscale atmospheric modeling: Equation sets and test cases, *Journal of Computational Physics* **227**, 3849–3877, 2008.

[100] V. GRAVEMEIER, The variational multiscale method for laminar and turbulent flow, *Archives of Computational Methods in Engineering* **13**, 249, 2006.

[101] V. GRAVEMEIER, M.W. GEE, M. KRONBICHLER, W.A. WALL, An algebraic variational multiscale-multigrid method for large eddy simulation of turbulent flow, *Computer Methods in Applied Mechanics and Engineering* **199**, 853–864, 2010.

[102] V. GRAVEMEIER, M.W. GEE, W.A. WALL, An algebraic variational multiscale-multigrid method based on plain aggregation for convection-diffusion problems, *Computer Methods in Applied Mechanics and Engineering* **198**, 3821–3835, 2009.

[103] V. GRAVEMEIER, M. KRONBICHLER, M.W. GEE, W.A. WALL, An algebraic variational multiscale-multigrid method for large-eddy simulation: generalized-α time integration, Fourier analysis and application to turbulent flow past a square-section cylinder, *Computational Mechanics* **47**, 217–233, 2011.

[104] V. Gravemeier, W.A. Wall, E. Ramm, A three-level finite element method for the instationary incompressible Navier–Stokes equations, *Computer Methods in Applied Mechanics and Engineering* **193**, 1323–1366, 2004.

[105] J.L. Guermond, P. Minev, J. Shen, An overview of projection methods for incompressible flows, *Computer Methods in Applied Mechanics and Engineering* **195**, 6011–6045, 2006.

[106] J.L. Guermond, J. Shen, Velocity-correction projection methods for incompressible flows, *SIAM Journal on Numerical Analysis* **41**, 112–134, 2003.

[107] E. Hanert, E. Deleersnijder, S. Blaise, J.F. Remacle, Capturing the bottom boundary layer in finite element ocean models, *Ocean Modelling* **17**, 153–162, 2007.

[108] I. Harari, T.J.R. Hughes, What are C and h?: Inequalities for the analysis and design of finite element methods, *Computer Methods in Applied Mechanics and Engineering* **97**, 157–192, 1992.

[109] R. Hartmann, Adjoint consistency analysis of discontinuous Galerkin discretizations, *SIAM Journal on Numerical Analysis* **45**, 2671–2696, 2007.

[110] R. Hartmann, H. McMorris, T. Leicht, Curved grid generation and DG computation for the DLR-F11 high lift configuration, in: M. Papadrakakis, V. Papadopoulos, G. Stefanou, V. Plevris (eds.), *Proceedings of the ECCOMAS Congress 2016*, Crete Island, Greece, 5–10 June, 2016.

[111] G. Hermange, *Extension of an enrichment-based wall model to complex geometries*, Semesterarbeit, Technical University of Munich, 2016.

[112] J.S. Hesthaven, T. Warburton, *Nodal discontinuous Galerkin methods: Algorithms, analysis, and applications*, Springer, 2008.

[113] K. Hillewaert, *Development of the discontinuous Galerkin method for high-resolution, large scale CFD and acoustics in industrial geometries*, Dissertation, Université Catholique de Louvain, 2013.

[114] F. Hindenlang, T. Bolemann, C.D. Munz, Mesh curving techniques for high order discontinuous Galerkin simulations, in: N. Kroll, C. Hirsch, F. Bassi, C. Johnston, K. Hillewaert (eds.), *IDIHOM: Industrialization of High-Order Methods - A Top-Down Approach: Results of a Collaborative Research Project Funded by the European Union, 2010 - 2014*, Springer, 2015.

[115] F. Hindenlang, G.J. Gassner, C. Altmann, A. Beck, M. Staudenmaier, C.D. Munz, Explicit discontinuous Galerkin methods for unsteady problems, *Computers & Fluids* **61**, 86–93, 2012.

[116] S. Hoyas, J. Jiménez, Scaling of the velocity fluctuations in turbulent channels up to Re_τ=2003, *Physics of Fluids* **18**, 011702, 2006.

[117] T.J.R. Hughes, 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**, 387–401, 1995.

[118] T.J.R. Hughes, G.R. Feijóo, L. Mazzei, J.B. Quincy, The variational multiscale method – a paradigm for computational mechanics, *Computer Methods in Applied Mechanics and Engineering* **166**, 3–24, 1998.

[119] T.J.R. Hughes, L.P. Franca, M. Balestra, A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška–Brezzi condition: A stable Petrov–Galerkin formulation of the Stokes problem accommodating equal-order interpolations, *Computer Methods in Applied Mechanics and Engineering* **59**, 85–99, 1986.

[120] T.J.R. Hughes, L. Mazzei, K.E. Jansen, Large eddy simulation and the variational multiscale method, *Computing and Visualization in Science* **3**, 47–59, 2000.

[121] J.C.R. Hunt, A.A. Wray, P. Moin, Eddies, streams, and convergence zones in turbulent flows, in: *Center for Turbulence Research, Proceedings of the Summer Program*, 1988.

[122] S. Jäger, *Investigation of a wall model for large-eddy simulation of the periodic hill*, Semesterarbeit, Technical University of Munich, 2015.

[123] S. Jakirlić, Periodic flow over a 2-D hill cross-plots $Re_H = 10600$, Slides of presentation held at Chalmers University in Gothenburg, Sweden, in April 2010, last accessed 24 November 2017,
http://cfd.mace.manchester.ac.uk/twiki/bin/view/ATAAC/TestCase001PeriodicHill.

[124] S. Jakirlić, R. Maduta, On steady RANS modeling for improved prediction of wall-bounded separation, *AIAA paper*, no. 2014-0586, 2014.

[125] S. Jakirlić, R. Maduta, Extending the bounds of steady RANS closures: Toward an instability-sensitive Reynolds stress model, *International Journal of Heat and Fluid Flow* **51**, 175–194, 2015.

[126] K.E. Jansen, C.H. Whiting, G.M. Hulbert, A generalized-α method for integrating the filtered Navier-Stokes equations with a stabilized finite element method, *Computer Methods in Applied Mechanics and Engineering* **190**, 305–319, 2000.

[127] J. Jiménez, Cascades in wall-bounded turbulence, *Annual Review of Fluid Mechanics* **44**, 27–45, 2012.

[128] V. John, Reference values for drag and lift of a two-dimensional time-dependent flow around a cylinder, *International Journal for Numerical Methods in Fluids* **44**, 777–788, 2004.

[129] S.M. Joshi, P.J. Diamessis, D.T. Steinmoeller, M. Stastna, G.N. Thomsen, A post-processing technique for stabilizing the discontinuous pressure projection operator in marginally-resolved incompressible inviscid flow, *Computers & Fluids* **139**, 120–129, 2016.

[130] S. Jovic, D.M. Driver, Backward-facing step measurements at low Reynolds number, $Re_h = 5000$, NASA TM 108807, 1994.

[131] C.J. Kähler, S. Scharnowski, C. Cierpka, Highly resolved experimental results of the separated flow in a channel with streamwise periodic constrictions, *Journal of Fluid Mechanics* **796**, 257–284, 2016.

[132] I. Kalashnikova, C. Farhat, R. Tezaur, A discontinuous enrichment method for the finite element solution of high Péclet advection-diffusion problems, *Finite Elements in Analysis and Design* **45**, 238–250, 2009.

[133] I. Kalashnikova, R. Tezaur, C. Farhat, A discontinuous enrichment method for variable-coefficient advection-diffusion at high Péclet number, *International Journal for Numerical Methods in Engineering* **87**, 309–335, 2011.

[134] G. Kalitzin, G. Medic, G. Iaccarino, P. Durbin, Near-wall behavior of RANS turbulence models and implications for wall functions, *Journal of Computational Physics* **204**, 265–291, 2005.

[135] G.E. Karniadakis, M. Israeli, S.A. Orszag, High-order splitting methods for the incompressible Navier–Stokes equations, *Journal of Computational Physics* **97**, 414–443, 1991.

[136] G.E. Karniadakis, S.J. Sherwin, *Spectral/hp element methods for computational fluid dynamics*, Oxford University Press, 2013.

[137] S. Kawai, J. Larsson, Wall-modeling in large eddy simulation: Length scales, grid resolution, and accuracy, *Physics of Fluids* **24**, 015105, 2012.

[138] S. Kawai, J. Larsson, Dynamic non-equilibrium wall-modeling for large eddy simulation at high Reynolds numbers, *Physics of Fluids* **25**, 015105, 2013.

[139] A. Keating, U. Piomelli, A dynamic stochastic forcing method as a wall-layer model for large-eddy simulation, *Journal of Turbulence* **7**, N12, 2006.

[140] R.M. Kirby, G.E. Karniadakis, De-aliasing on non-uniform grids: Algorithms and applications, *Journal of Computational Physics* **191**, 249–264, 2003.

[141] B. Klein, F. Kummer, M. Keil, M. Oberlack, An extension of the SIMPLE based discontinuous Galerkin solver to unsteady incompressible flows, *International Journal for Numerical Methods in Fluids* **77**, 571–589, 2015.

[142] B. Klein, F. Kummer, M. Oberlack, A SIMPLE based discontinuous Galerkin solver for steady incompressible flows, *Journal of Computational Physics* **237**, 235–250, 2013.

[143] T. Knopp, T. Alrutz, D. Schwamborn, A grid and flow adaptive wall-function method for RANS turbulence modelling, *Journal of Computational Physics* **220**, 19–40, 2006.

[144] A.N. KOLMOGOROV, The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers, in: *Dokl. Akad. Nauk SSSR*, vol. 30, 1941.

[145] M. KOMPENHANS, E. FERRER, G. RUBIO, E. VALERO, Comparisons of compressible and incompressible solvers: Flat plate boundary layer and NACA airfoils, in: *2nd International Workshop on High Order Methods*, Cologne, 2013.

[146] D.A. KOPRIVA, *Implementing spectral methods for partial differential equations: Algorithms for scientists and engineers*, Springer, 2009.

[147] B. KRANK, N. FEHN, W.A. WALL, M. KRONBICHLER, A high-order semi-explicit discontinuous Galerkin solver for 3D incompressible flow with application to DNS and LES of turbulent channel flow, *Journal of Computational Physics* **348**, 634–659, 2017.

[148] B. KRANK, M. KRONBICHLER, W.A. WALL, DNS reference data of flow over periodic hills at Re=5600 and 10595, *mediaTUM Public Repository*, Technical University of Munich, 2018, https://doi.org/10.14459/2018mp1415670.

[149] B. KRANK, M. KRONBICHLER, W.A. WALL, Direct numerical simulation of flow over periodic hills up to $Re_H = 10{,}595$, *Flow, Turbulence and Combustion* **101**, 521–551, 2018.

[150] B. KRANK, M. KRONBICHLER, W.A. WALL, A multiscale approach to hybrid RANS/LES wall modeling within a high-order discontinuous Galerkin scheme using function enrichment, *International Journal for Numerical Methods in Fluids*, 1–33, 2019, https://doi.org/10.1002/fld.4712.

[151] B. KRANK, M. KRONBICHLER, W.A. WALL, Wall modeling via function enrichment: Extension to detached-eddy simulation, *Computers & Fluids*, 2018, https://doi.org/10.1016/j.compfluid.2018.07.019.

[152] B. KRANK, M. KRONBICHLER, W.A. WALL, Wall modeling via function enrichment within a high-order DG method for RANS simulations of incompressible flow, *International Journal for Numerical Methods in Fluids* **86**, 107–129, 2018.

[153] B. KRANK, W.A. WALL, A new approach to wall modeling in LES of incompressible flow via function enrichment, *Journal of Computational Physics* **316**, 94–116, 2016.

[154] B. KRANK, W.A. WALL, A novel approach for wall modeling in LES of wall-bounded high-Reynolds-number flow via function enrichment, in: D.G. GRIGORIADIS, B.J. GEURTS, H. KUERTEN, J. FRÖHLICH, V. ARMENIO (eds.), *Direct and Large-Eddy Simulation X*, Springer, 2018.

[155] G. KREISS, B. KRANK, G. EFRAIMSSON, Analysis of stretched grids as buffer zones in simulations of wave propagation, *Applied Numerical Mathematics* **107**, 1–17, 2016.

[156] M. KRONBICHLER, K. KORMANN, A generic interface for parallel cell-based finite element operator application, *Computers & Fluids* **63**, 135–147, 2012.

[157] M. Kronbichler, K. Kormann, Fast matrix-free evaluation of discontinuous Galerkin finite element operators, 2017, arXiv preprint arXiv:1711.03590.

[158] M. Kronbichler, K. Kormann, I. Pasichnyk, M. Allalen, Fast matrix-free discontinuous Galerkin kernels on modern computer architectures, in: J.M. Kunkel, R. Yokota, P. Balaji, D. Keyes (eds.), *High Performance Computing: 32nd International Conference, ISC High Performance 2017*, Frankfurt, Germany, June 18–22, 2017, Springer, 2017.

[159] M. Kronbichler, B. Krank, N. Fehn, S. Legat, W.A. Wall, A new high-order discontinuous Galerkin solver for DNS and LES of turbulent incompressible flow, in: A. Dillmann, G. Heller, E. Krämer, C. Wagner, S. Bansmer, R. Radespiel, R. Semaan (eds.), *New Results in Numerical and Experimental Fluid Mechanics XI. Notes on Numerical Fluid Mechanics and Multidisciplinary Design* **136**, Springer, 2018.

[160] M. Kronbichler, B. Krank, W.A. Wall, Fast matrix-free evaluation of weak forms including function enrichments, in preparation.

[161] M. Kronbichler, S. Schoeder, C. Müller, W.A. Wall, Comparison of implicit and explicit hybridizable discontinuous Galerkin methods for the acoustic wave equation, *International Journal for Numerical Methods in Engineering* **106**, 712–739, 2016.

[162] M. Kronbichler, W.A. Wall, A performance comparison of continuous and discontinuous Galerkin methods with fast multigrid solvers, *SIAM Journal on Scientific Computing* **40**, A3423–A3448, 2018.

[163] M. Kufner, *Untersuchung einer Quadraturmethode für Anreicherungsansätze mit hohen Gradienten*, Semesterarbeit, Technical University of Munich, 2015.

[164] E. Labourasse, P. Sagaut, Reconstruction of turbulent fluctuations using a hybrid RANS/LES approach, *Journal of Computational Physics* **182**, 301–336, 2002.

[165] E. Labourasse, P. Sagaut, Advance in RANS–LES coupling, a review and an insight on the NLDE approach, *Archives of Computational Methods in Engineering* **11**, 199–256, 2004.

[166] B. Landmann, *A parallel discontinuous Galerkin code for the Navier–Stokes and Reynolds-averaged Navier–Stokes equations*, Dissertation, University of Stuttgart, 2007.

[167] B. Landmann, M. Kessler, S. Wagner, E. Krämer, A parallel, high-order discontinuous Galerkin code for laminar and turbulent flows, *Computers & Fluids* **37**, 427–438, 2008.

[168] J. Larsson, S. Kawai, J. Bodart, I. Bermejo-Moreno, Large eddy simulation with modeled wall-stress: Recent progress and future directions, *Mechanical Engineering Reviews* **3**, 15–00418–15–00418, 2016.

[169] B.E. Launder, D.B. Spalding, The numerical computation of turbulent flows, *Computer Methods in Applied Mechanics and Engineering* **3**, 269–289, 1974.

[170] H. LE, P. MOIN, J. KIM, Direct numerical simulation of turbulent flow over a backward-facing step, *Journal of Fluid Mechanics* **330**, 349–374, 1997.

[171] P.L. LEDERER, C. LEHRENFELD, J. SCHÖBERL, Hybrid discontinuous Galerkin methods with relaxed H(div)-conformity for incompressible flows. Part I, *SIAM Journal on Numerical Analysis* **56**, 2070–2094, 2018.

[172] M. LEE, R.D. MOSER, Direct numerical simulation of turbulent channel flow up to $Re_\tau \approx 5200$, *Journal of Fluid Mechanics* **774**, 395–415, 2015.

[173] S. LEGAT, *Advancement of a semi-explicit discontinuous Galerkin approach for simulation of turbulent incompressible flow*, Master's thesis, Technical University of Munich, 2016.

[174] C. LEHRENFELD, J. SCHÖBERL, High order exactly divergence-free hybrid discontinuous Galerkin methods for unsteady incompressible flows, *Computer Methods in Applied Mechanics and Engineering* **307**, 339–361, 2016.

[175] A. LEONARD, Energy cascade in large-eddy simulations of turbulent fluid flows, in: F. FRENKIEL, R. MUNN (eds.), *Turbulent Diffusion in Environmental Pollution. Advances in Geophysics* **18**, Elsevier, 1975.

[176] E. LERICHE, G. LABROSSE, High-order direct Stokes solvers with or without temporal splitting: Numerical investigations of their comparative properties, *SIAM Journal on Scientific Computing* **22**, 1386–1410, 2000.

[177] E. LERICHE, E. PERCHAT, G. LABROSSE, M.O. DEVILLE, Numerical evaluation of the accuracy and stability properties of high-order direct Stokes solvers with or without temporal splitting, *Journal of Scientific Computing* **26**, 25–43, 2006.

[178] M. DE LA LLAVE PLATA, V. COUAILLIER, M.C. LE PAPE, On the use of a high-order discontinuous Galerkin method for DNS and LES of wall-bounded turbulence, *Computers & Fluids* **176**, 320-337, 2018.

[179] C. LÜBON, *Turbulenzmodellierung und Detached-Eddy-Simulationen mit einem Discontinuous-Galerkin-Verfahren hoher Ordnung*, Dissertation, University of Stuttgart, 2011.

[180] Y. MADAY, A.T. PATERA, Spectral element methods for the incompressible Navier–Stokes equations, in: *State-of-the-Art Surveys on Computational Mechanics* (A90-47176 21-64). New York, ASME, 1989.

[181] M. MAREK, A. TYLISZCZAK, A. BOGUSŁAWSKI, Large eddy simulation of incompressible free round jet with discontinuous Galerkin method, *International Journal for Numerical Methods in Fluids* **79**, 164–182, 2015.

[182] G. MEDIC, G. DAENINCK, J.A. TEMPLETON, G. KALITZIN, A framework for near-wall RANS/LES coupling, in: *Center for Turbulence Research, Annual Research Briefs*, 169–182, 2005.

[183] J.M. MELENK, I. BABUŠKA, The partition of unity finite element method: Basic theory and applications, *Computer Methods in Applied Mechanics and Engineering* **139**, 289–314, 1996.

[184] P. MOIN, K. MAHESH, Direct numerical simulation: A tool in turbulence research, *Annual Review of Fluid Mechanics* **30**, 539–578, 1998.

[185] D. MORO, N.C. NGUYEN, J. PERAIRE, Navier–Stokes solution using hybridizable discontinuous Galerkin methods, *AIAA paper*, no. 2011-3407, 2011.

[186] P.J. MORRIS, L.N. LONG, A. BANGALORE, Q. WANG, A parallel three-dimensional computational aeroacoustics method using nonlinear disturbance equations, *Journal of Computational Physics* **133**, 56–74, 1997.

[187] R.D. MOSER, J. KIM, N.N. MANSOUR, Direct numerical simulation of turbulent channel flow up to Re_τ=590, *Physics of Fluids* **11**, 943–945, 1999.

[188] R.C. MOURA, G. MENGALDO, J. PEIRÓ, S.J. SHERWIN, On the eddy-resolving capability of high-order discontinuous Galerkin approaches to implicit LES / under-resolved DNS of Euler turbulence, *Journal of Computational Physics* **330**, 615–623, 2017.

[189] R.C. MOURA, S.J. SHERWIN, J. PEIRÓ, Linear dispersion–diffusion analysis and its application to under-resolved turbulence simulations using discontinuous Galerkin spectral/hp methods, *Journal of Computational Physics* **298**, 695–710, 2015.

[190] S.E. MOUSAVI, J.E. PASK, N. SUKUMAR, Efficient adaptive integration of functions with sharp gradients and cusps in n-dimensional parallelepipeds, *International Journal for Numerical Methods in Engineering* **91**, 343–357, 2012.

[191] S.E. MOUSAVI, H. XIAO, N. SUKUMAR, Generalized Gaussian quadrature rules on arbitrary polygons, *International Journal for Numerical Methods in Engineering* **82**, 99–113, 2010.

[192] J.A. MULLIN, W.J.A. DAHM, Dual-plane stereo particle image velocimetry measurements of velocity gradient tensor fields in turbulent shear flow. II. Experimental results, *Physics of Fluids* **18**, 035102, 2006.

[193] N.C. NGUYEN, J. PERAIRE, B. COCKBURN, An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier–Stokes equations, *Journal of Computational Physics* **230**, 1147–1170, 2011.

[194] N.C. NGUYEN, P.O. PERSSON, J. PERAIRE, RANS solutions using high order discontinuous Galerkin methods, *AIAA paper*, no. 2007-914, 2007.

[195] N.V. NIKITIN, F. NICOUD, B. WASISTHO, K.D. SQUIRES, P.R. SPALART, An approach to wall modeling in large-eddy simulations, *Physics of Fluids* **12**, 1629–1632, 2000.

[196] G. NOVENTA, F. MASSA, F. BASSI, A. COLOMBO, N. FRANCHINA, A. GHIDONI, A high-order discontinuous Galerkin solver for unsteady incompressible turbulent flows, *Computers & Fluids* **139**, 248–260, 2016.

[197] M. Olshanskii, G. Lube, T. Heister, J. Löwe, Grad–div stabilization and subgrid pressure models for the incompressible Navier–Stokes equations, *Computer Methods in Applied Mechanics and Engineering* **198**, 3975–3988, 2009.

[198] S.A. Orszag, Spectral methods for problems in complex geometries, *Journal of Computational Physics* **37**, 70–92, 1980.

[199] S.A. Orszag, M. Israeli, M.O. Deville, Boundary conditions for incompressible flows, *Journal of Scientific Computing* **1**, 75–111, 1986.

[200] G.I. Park, P. Moin, An improved dynamic non-equilibrium wall-model for large eddy simulation, *Physics of Fluids* **26**, 015108, 2014.

[201] A.T. Patera, A spectral element method for fluid dynamics: Laminar flow in a channel expansion, *Journal of Computational Physics* **54**, 468–488, 1984.

[202] M. Piatkowski, S. Müthing, P. Bastian, A stable and high-order accurate discontinuous Galerkin based splitting method for the incompressible Navier–Stokes equations, *Journal of Computational Physics* **356**, 220–239, 2018.

[203] U. Piomelli, Wall-layer models for large-eddy simulations, *Progress in Aerospace Sciences* **44**, 437–446, 2008.

[204] U. Piomelli, E. Balaras, Wall-layer models for large-eddy simulations, *Annual Review of Fluid Mechanics* **34**, 349–374, 2002.

[205] U. Piomelli, S. Radhakrishnan, G. De Prisco, Turbulent eddies in the RANS/LES transition region, in: S.H. Peng, W. Haase (eds.), *Advances in Hybrid RANS–LES Modelling: Papers contributed to the 2007 Symposium of Hybrid RANS-LES Methods*, Corfu, Greece, 17-18 June 2007, Springer, 2008.

[206] S.B. Pope, *Turbulent Flows*, Cambridge University Press, 2000.

[207] M. Popovac, K. Hanjalic, Compound wall treatment for RANS computation of complex turbulent flows and heat transfer, *Flow, Turbulence and Combustion* **78**, 177–202, 2007.

[208] B. Rajamani, J. Kim, A hybrid-filter approach to turbulence simulation, *Flow, Turbulence and Combustion* **85**, 421–441, 2010.

[209] Ch. Rapp, *Experimentelle Studie der turbulenten Strömung über periodische Hügel*, Dissertation, Technical University of Munich, 2009.

[210] Ch. Rapp, M. Breuer, M. Manhart, N. Peller, 2D Periodic hill flow, 2013, http://qnet-ercoftac.cfms.org.uk/, last accessed 24 November 2017.

[211] Ch. Rapp, M. Manhart, Flow over periodic hills: An experimental study, *Experiments in Fluids* **51**, 247–269, 2011.

[212] U. RASTHOFER, *Computational multiscale methods for turbulent single and two-phase flows*, Dissertation, Technical University of Munich, 2015.

[213] U. RASTHOFER, G.C. BURTON, W.A. WALL, V. GRAVEMEIER, An algebraic variational multiscale-multigrid-multifractal method (AVM4) for large-eddy simulation of turbulent variable-density flow at low Mach number, *International Journal for Numerical Methods in Fluids* **76**, 416–449, 2014.

[214] U. RASTHOFER, G.C. BURTON, W.A. WALL, V. GRAVEMEIER, Multifractal subgrid-scale modeling within a variational multiscale method for large-eddy simulation of passive-scalar mixing in turbulent flow at low and high Schmidt numbers, *Physics of Fluids* **26**, 055108, 2014.

[215] U. RASTHOFER, V. GRAVEMEIER, Multifractal subgrid-scale modeling within a variational multiscale method for large-eddy simulation of turbulent flow, *Journal of Computational Physics* **234**, 79–107, 2013.

[216] U. RASTHOFER, V. GRAVEMEIER, Recent developments in variational multiscale methods for large-eddy simulation of turbulent flow, *Archives of Computational Methods in Engineering* **25**, 647–690, 2018.

[217] H. REICHARDT, Vollständige Darstellung der turbulenten Geschwindigkeitsverteilung in glatten Leitungen, *Journal of Applied Mathematics and Mechanics* **31**, 208–219, 1951.

[218] S. RHEBERGEN, B. COCKBURN, J.J.W. VAN DER VEGT, A space–time discontinuous Galerkin method for the incompressible Navier–Stokes equations, *Journal of Computational Physics* **233**, 339–358, 2013.

[219] H. RIOU, P. LADEVÈZE, A new numerical strategy for the resolution of high-Péclet advection-diffusion problems, *Computer Methods in Applied Mechanics and Engineering* **241**, 302–310, 2012.

[220] B. RIVIÈRE, M.F. WHEELER, V. GIRAULT, Improved energy estimates for interior penalty, constrained and discontinuous Galerkin methods for elliptic problems. Part I, *Computational Geosciences* **3**, 337–360, 1999.

[221] S.K. ROBINSON, Coherent motions in the turbulent boundary layer, *Annual Review of Fluid Mechanics* **23**, 601–639, 1991.

[222] P. SAGAUT, *Large eddy simulation for incompressible flows: An introduction*, Springer, 2006.

[223] P. SAGAUT, S. DECK, M. TERRACOL, *Multiscale and multiresolution approaches in turbulence: LES, DES and hybrid RANS/LES methods: applications and guidelines*, World Scientific, 2013.

[224] S. ŠARIĆ, S. JAKIRLIĆ, M. BREUER, B. JAFFRÉZIC, G. DENG, O. CHIKHAOUI, J. FRÖHLICH, D. VON TERZI, M. MANHART, N. PELLER, Evaluation of detached eddy simulations for predicting the flow over periodic hills, in: *ESAIM: Proceedings* **16**, 2007.

[225] M. SCHÄFER, S. TUREK, F. DURST, E. KRAUSE, R. RANNACHER, Benchmark computations of laminar flow around a cylinder, Springer, 1996.

[226] B. SCHOTT, *Stabilized cut finite element methods for complex interface coupled flow problems*, Dissertation, Technical University of Munich, 2017.

[227] B. SCHOTT, S. SHAHMIRI, R. KRUSE, W.A. WALL, A stabilized Nitsche-type extended embedding mesh approach for 3D low- and high-Reynolds-number flows, *International Journal for Numerical Methods in Fluids* **82**, 289–315, 2016.

[228] B. SCHOTT, W.A. WALL, A new face-oriented stabilized XFEM approach for 2D and 3D incompressible Navier–Stokes equations, *Computer Methods in Applied Mechanics and Engineering* **276**, 233–265, 2014.

[229] D. SCHÖTZAU, C. SCHWAB, A. TOSELLI, Mixed hp-DGFEM for incompressible flows, *SIAM Journal on Numerical Analysis* **40**, 2171–2194, 2002.

[230] P.W. SCHROEDER, G. LUBE, Stabilised DG-FEM for incompressible natural convection flows with boundary and moving interior layers on non-adapted meshes, *Journal of Computational Physics* **335**, 760–779, 2017.

[231] U. SCHUMANN, Subgrid scale model for finite difference simulations of turbulent flows in plane channels and annuli, *Journal of Computational Physics* **18**, 376–404, 1975.

[232] D. SCHWAMBORN, M. STRELETS, ATAAC – an EU-project dedicated to hybrid RANS/LES methods, in: S. FU, W. HAASE, S.H. PENG, D. SCHWAMBORN (eds.), *Progress in Hybrid RANS–LES Modelling: Papers Contributed to the 4th Symposium on Hybrid RANS–LES Methods*, Beijing, China, September 2011, Springer, 2012.

[233] K. SHAHBAZI, An explicit expression for the penalty parameter of the interior penalty method, *Journal of Computational Physics* **205**, 401–407, 2005.

[234] K. SHAHBAZI, P.F. FISCHER, C. ROSS ETHIER, A high-order discontinuous Galerkin method for the unsteady incompressible Navier–Stokes equations, *Journal of Computational Physics* **222**, 391–407, 2007.

[235] T.H. SHIH, L.A. POVINELLI, N.S. LIU, M.G. POTAPCZUK, J.L. LUMLEY, A generalized wall function, NASA TM-1999-209398, 1999.

[236] M. SHUR, P.R. SPALART, M. STRELETS, A. TRAVIN, Detached-eddy simulation of an airfoil at high angle of attack, *Engineering Turbulence Modelling and Experiments* **4**, 669–678, 1999.

[237] M.L. SHUR, P.R. SPALART, M.KH. STRELETS, A.K. TRAVIN, A hybrid RANS–LES approach with delayed-DES and wall-modelled LES capabilities, *International Journal of Heat and Fluid Flow* **29**, 1638–1649, 2008.

[238] R.L. SIMPSON, A model for the backflow mean velocity profile, *AIAA Journal* **21**, 142–143, 1983.

[239] P.R. SPALART, Direct simulation of a turbulent boundary layer up to $\mathrm{Re}_\theta = 1410$, *Journal of Fluid Mechanics* **187**, 61–98, 1988.

[240] P.R. SPALART, Strategies for turbulence modelling and simulations, *International Journal of Heat and Fluid Flow* **21**, 252–263, 2000.

[241] P.R. SPALART, Detached-eddy simulation, *Annual Review of Fluid Mechanics* **41**, 181–202, 2009.

[242] P.R. SPALART, S.R. ALLMARAS, A one-equation turbulence model for aerodynamic flows, *La Recherche Aérospatiale* **1**, 5–21, 1994.

[243] P.R. SPALART, S. DECK, M.L. SHUR, K.D. SQUIRES, M.KH. STRELETS, A. TRAVIN, A new version of detached-eddy simulation, resistant to ambiguous grid densities, *Theoretical and Computational Fluid Dynamics* **20**, 181–195, 2006.

[244] P.R. SPALART, W.H. JOU, M. STRELETS, S.R. ALLMARAS, Comments on the feasibility of LES for wings, and on a hybrid RANS/LES approach, in: C. LIU, Z. LIU (eds.), *Advances in DNS/LES*, Greyden Press Columbus, OH, 1997.

[245] D.B. SPALDING, A single formula for the "law of the wall", *Journal of Applied Mechanics* **28**, 455–458, 1961.

[246] D.T. STEINMOELLER, M. STASTNA, K.G. LAMB, A short note on the discontinuous Galerkin discretization of the pressure projection operator in incompressible flow, *Journal of Computational Physics* **251**, 480–486, 2013.

[247] Y. SUDHAKAR, W.A. WALL, Quadrature schemes for arbitrary convex/concave volumes and integration of weak form in enriched partition of unity methods, *Computer Methods in Applied Mechanics and Engineering* **258**, 39–54, 2013.

[248] M. TAVELLI, M. DUMBSER, A staggered space–time discontinuous Galerkin method for the three-dimensional incompressible Navier–Stokes equations on unstructured tetrahedral meshes, *Journal of Computational Physics* **319**, 294–323, 2016.

[249] L. TEMMERMAN, M.A. LESCHZINER, C.P. MELLEN, J. FRÖHLICH, Investigation of wall-function approximations and subgrid-scale models in large eddy simulation of separated flow in a channel with streamwise periodic constrictions, *International Journal of Heat and Fluid Flow* **24**, 157–180, 2003.

[250] T.E. TEZDUYAR, S. MITTAL, S.E. RAY, R. SHIH, Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity-pressure elements, *Computer Methods in Applied Mechanics and Engineering* **95**, 221–242, 1992.

[251] H.M. TUFO, P.F. FISCHER, Terascale spectral element algorithms and implementations, in: *Proceedings of the 1999 ACM/IEEE conference on Supercomputing*, ACM, 1999.

[252] D. TURNER, D. NOBLE, Analytic enrichment of finite element formulations for capturing boundary layer behavior, *AIAA paper*, no. 2010-4996, 2010.

[253] D.Z. TURNER, K.B. NAKSHATRALA, K.D. HJELMSTAD, A stabilized formulation for the advection-diffusion equation using the generalized finite element method, *International Journal for Numerical Methods in Fluids* **66**, 64–81, 2011.

[254] R.S. VARGA, *Matrix iterative analysis*, Springer, 2009.

[255] W. VIESER, T. ESCH, F. MENTER, Heat transfer predictions using advanced two-equation turbulence models, *CFX Validation Report*, Report No. CFX-VAL 10, 0602, 2002.

[256] A.W. VREMAN, J.G.M. KUERTEN, Comparison of direct numerical simulation databases of turbulent channel flow at $Re_\tau = 180$, *Physics of Fluids* **26**, 015102, 2014.

[257] D. WANG, S.J. RUUTH, Variable step-size implicit-explicit linear multistep methods for time-dependent partial differential equations, *Journal of Computational Mathematics* **26**, 838–855, 2008.

[258] W.A. WALL, C. AGER, M. GRILL, M. KRONBICHLER, A. POPP, A. SEITZ, BACI: A multiphysics simulation environment. Technical report, Institute for Computational Mechanics, Technical University of Munich, 2017.

[259] L. WANG, W. KYLE ANDERSON, T. ERWIN, S. KAPADIA, High-order discontinuous Galerkin method for computation of turbulent flows, *AIAA Journal* **53**, 1159–1171, 2015.

[260] M. WANG, P. MOIN, Dynamic wall modeling for large-eddy simulation of complex turbulent flows, *Physics of Fluids* **14**, 2043–2051, 2002.

[261] H. WERNER, H. WENGLE, Large-eddy simulation of turbulent flow over and around a cube in a plate channel, in: F. DURST, R. FRIEDRICH, B.E. LAUNDER, F.W. SCHMIDT, U. SCHUMANN, J.H. WHITELAW (eds.), *Turbulent Shear Flows 8: Selected Papers from the Eighth International Symposium on Turbulent Shear Flows*, Munich, Germany, September 9 – 11, 1991, Springer, 1993.

[262] C. CARTON DE WIART, K. HILLEWAERT, L. BRICTEUX, G. WINCKELMANS, Implicit LES of free and wall-bounded turbulent flows based on the discontinuous Galerkin/symmetric interior penalty method, *International Journal for Numerical Methods in Fluids* **78**, 335–354, 2015.

[263] C. CARTON DE WIART, K. HILLEWAERT, M. DUPONCHEEL, G. WINCKELMANS, Assessment of a discontinuous Galerkin method for the simulation of vortical flows at high Reynolds number, *International Journal for Numerical Methods in Fluids* **74**, 469–493, 2014.

[264] C. CARTON DE WIART, K. HILLEWAERT, E. LORRIAUX, G. VERHEYLEWEGEN, Development of a discontinuous Galerkin solver for high quality wall-resolved/modelled DNS and LES of practical turbomachinery flows on fully unstructured meshes, ASME GT2015-43428, Montreal, Canada, 2015.

[265] C. CARTON DE WIART, S.M. MURMAN, Assessment of wall-modeled LES strategies within a discontinuous-Galerkin spectral-element framework, *AIAA paper*, no. 2017-1223, 2017.

[266] D.C. WILCOX, *Turbulence modeling for CFD*, vol. 2, DCW industries La Canada, CA, 1998.

[267] M. WURST, M. KESSLER, E. KRÄMER, Aerodynamic and acoustic analysis of an extruded airfoil with a trailing edge device using detached eddy simulation with a discontinuous Galerkin method, *AIAA paper*, no. 2013-2429, 2013.

[268] X.I.A. YANG, G.I. PARK, P. MOIN, Log-layer mismatch and modeling of the fluctuating wall stress in wall-modeled large-eddy simulations, *Physical Review Fluids* **2**, 104601, 2017.

[269] X.I.A. YANG, J. SADIQUE, R. MITTAL, C. MENEVEAU, Integral wall model for large eddy simulations of wall-bounded turbulent flows, *Physics of Fluids* **27**, 025112, 2015.

[270] J. ZHENHUA, Y. CHAO, Y. JIAN, Q. FENG, Y. WU, A Spalart–Allmaras turbulence model implementation for high-order discontinuous Galerkin solution of the Reynolds-averaged Navier–Stokes equations, *Flow, Turbulence and Combustion* **96**, 623–638, 2016.

Verzeichnis der betreuten Studienarbeiten

Im Rahmen dieser Dissertation entstanden am Lehrstuhl für Numerische Mechanik (LNM) in den Jahren von 2013 bis 2017 unter wesentlicher wissenschaftlicher, fachlicher und inhaltlicher Anleitung des Autors die im Folgenden aufgeführten studentischen Arbeiten. Der Autor dankt allen Sudierenden für ihr Engagement bei der Unterstützung dieser wissenschaftlichen Arbeit.

Studierende(r)	**Studienarbeit**
Sebastian Büchner	*Investigation of subgrid scale models for LES of a generic blood flow*, Semesterarbeit, 2014
Stephan Jäger	*Investigation of a wall model for large-eddy simulation of the periodic hill*, Semesterarbeit, 2015, eingeflossen in Abschnitt 7.3.2
Markus Koschi	*Better consistency of residual-based stabilization for linear finite elements applied to the incompressible Navier–Stokes equations*, Semesterarbeit, 2015, Betreuung in Zusammenarbeit mit Georg Hammerl
Matthias Kufner	*Untersuchung einer Quadraturmethode für Anreicherungsansätze mit hohen Gradienten*, Semesterarbeit, 2015, eingeflossen in Abschnitt 6.3.3.1
Deniz Bezgin	*Investigation of explicit scale-separation approaches for HDG*, Bachelorarbeit, 2015
Charles Giachetti	*Investigation of an enrichment-based wall model applied to a cylinder in turbulent axial flow*, Semesterarbeit, 2016
Gurvan Hermange	*Extension of an enrichment-based wall model to complex geometries*, Semesterarbeit, 2016, eingeflossen in Abschnitt 7.1
Niklas Fehn	*A discontinuous Galerkin approach for the unsteady incompressible Navier–Stokes equations*, Masterarbeit, 2015, eingeflossen in Abschnitt 3.2, 3.3, 3.5 und 3.6, Betreuung in Zusammenarbeit mit Dr. Martin Kronbichler
Stefan Legat	*Advancement of a semi-explicit discontinuous Galerkin approach for simulation of turbulent incompressible flow*, Masterarbeit, 2016, eingeflossen in Abschnitt 3.3.3 und 4.3