Inaugural - Dissertation

zur
Erlangung der Doktorwürde
der
Naturwissenschaftlich-Mathematischen Gesamtfakultät
der
Ruprecht-Karls-Universität
Heidelberg

vorgelegt von

Dipl.-Math. B.Sc. Physik

Stephan-Marian Piatkowski

aus Heidelberg

Tag der mündlichen Prüfung: .....

A Spectral Discontinuous Galerkin method for incompressible flow

with Applications to turbulence

Betreuer: Prof. Dr. Peter Bastian

Prof. Dr. Kurt Roth

Abstract

In this thesis we develop a numerical solution method for the instationary incompressible Navier-Stokes equations. The approach is based on projection methods for discretization in time and a higher order discontinuous Galerkin discretization in space. We propose an upwind scheme for the convective term that chooses the direction of flux across cell interfaces by the mean value of the velocity and has favorable properties in the context of DG. We present new variants of solenoidal projection operators in the Helmholtz decomposition which are indeed discrete projection operators. The discretization is accomplished on quadrilateral or hexahedral meshes where sum-factorization in tensor product finite elements can be exploited. Sum-factorization significantly reduces algorithmic complexity during assembling. In this thesis we thereby build efficient scalable matrix-free solvers and preconditioners to tackle the arising subproblems in the discretization. Conservation properties of the numerical method are demonstrated for both problems with exact solution and turbulent flows. Finally, the presented DG solver enables long time stable direct numerical simulations of the Navier-Stokes equations. As an application we perform computations on a model of the atmospheric boundary layer and demonstrate the existence of surface renewal.

Zusammenfassung

In der vorliegenden Arbeit wird eine numerische Methode zur Lösung der instationären inkompressiblen Navier-Stokes Gleichungen entwickelt. Die Methode basiert in der Zeitdiskretisierung auf Projektionsverfahren und verwendet in der Ortsdiskretisierung ein DG-Verfahren höherer Ordnung. Ein Upwinding des Konvektionsterms wird vorgeschlagen, welches die Richtung des Flusses über die Gitterelemente durch den Mittelwert der Geschwindigkeit wählt und im Zusammenhang mit DG-Verfahren vorteilhaft ist. Neue Klassen von divergenzfreien Projektionsoperatoren in der Helmholtz-Zerlegung werden vorgestellt, welche tatsächlich diskrete Projektionsoperatoren sind. Die Diskretisierung geschieht auf Viereckgittern oder Hexaedergittern, auf denen die Tensorproduktstruktur in Finiten Elementen für Summenfaktorisierung ausgenutzt werden kann. Summenfaktorisierung reduziert die algorithmische Komplexität im Assemblierungsprozess wesentlich. Damit werden im Rahmen dieser Arbeit effiziente, skalierbare matrixfreie Löser und Vorkonditionierer entwickelt, welche die Teilprobleme innerhalb der Diskretisierung lösen. Erhaltungseigenschaften des numerischen Verfahrens werden sowohl für Probleme mit exakter Lösung als auch für turbulente Strömungen demonstriert. Schließlich ermöglicht der vorgestellte DG-basierte Löser langzeitstabile direkte numerische Simulationen der Navier-Stokes Gleichungen. Eine Anwendung ist die Berechnung eines Modells für die Grenzschicht von Boden und Atmosphäre, welche das Vorhandensein der Oberflächenerneuerungstheorie zeigt.

Acknowledgments

Hereby I would like to express my deep gratitude to both my supervisors Peter Bastian and Kurt Roth for their constant encouragement during my doctoral studies, countless valuable remarks and motivating critique at every point of my research.

In addition I would like to emphasize my gratitude to my family and colleagues. Thank you for your support and forbearance.

My PhD studies at the Faculty of Mathematics and Computer Science of Heidelberg University and so this dissertation were supported by a stipend of the Heidelberg Graduate School of Mathematical and Computational Methods for the Sciences (GSC 220).

Contents

List of Acronyms

ABL	Atmospheric Boundary Layer
AMG	algebraic multigrid
CDF	cumulative distribution function
CDN	classical do-nothing condition
CFD	Computational Fluid Dynamics
CSS	Consistent Splitting Scheme
DDN	directional do-nothing condition
DG	discontinuous Galerkin
DNS	Direct Numerical Simulation
DOF	degree of freedom
DUNE	Distributed and Unified Numerics Environment
EXA-DUNE	exascale fork of DUNE
fBm	fractional Brownian motion
GLL	Gauss-Lobatto-Lagrange polynomials
GS	Gauss-Seidel
HiOCFD	High-Order CFD Methods
HIT	Homogeneous Isotropic Turbulence
HPC	High-Performance Computing
IIPG	Incomplete Interior Penalty Galerkin
IPCS	Incremental Pressure Correction Scheme
JFNK	Jacobian-Free Newton-Krylov
K41	Kolmogorov's 1941 theory of turbulence
LIC	Line Integral Convolution
MF	matrix-free
MOL	method of lines
MX	matrix-based
NIPG	Non-symmetric Interior Penalty Galerkin
PDF	probability density function

PMF	partially matrix-free
RIPCS	Rotational Incremental Pressure Correction Scheme
rms	root-mean-square
SDGM	Spectral Discontinuous Galerkin Method
SGS	symmetric Gauss-Seidel
SIPG	Symmetric Interior Penalty Galerkin
SR	surface renewal, in German: Oberflächenerneuerungstheorie
SSS	Standard Splitting Scheme

0 Introduction

0.1 Motivation

The simulation of atmosphere dynamics occupies a central and active area of research including the coupling to the subsurface. Observations reveal that in the Atmospheric Boundary Layer (ABL) coherent turbulent structures develop which move towards the surface where they are reflected, an effect called surface renewal (SR). In consequence, water vapor and trace gases come into contact with the surface and reside on this interface. During these quiescent periods concentration builds up and parcels are suddenly released by such renewal events. These repeated patterns of build-up and release observed in concentration time series are referred to as the ramp effect. An illustration of this process is depicted in figure 1.6.

In this thesis we want to demonstrate the existence of surface renewal and ramp effects in silico by DNS. We will solely concentrate on the ABL and its sublayers. Therefore, efficient numerical methods are required that can run and scale on high performance computers.

The mathematical model consists of the Navier-Stokes equations with variable density considered by the Boussinesq approximation, and the transport of scalars. If buoyancy and tracer transport are further neglected the system is referred to as incompressible Navier-Stokes. Flow in the ABL for instance is fully turbulent. Numerical simulation of the processes is a very challenging task due to the broad range of spatial and temporal scales involved, dominating convective nonlinearity and unpredictable behavior in space and time. The foremost goal is to develop an efficient and accurate numerical method for the Navier-Stokes Boussinesq system where the incompressible Navier-Stokes equations constitute numerically the most demanding fraction either as part of the system or standalone. The method shall be capable of solving three-dimensional problems in a reasonable amount of time, as well as be able to run on High-Performance Computing (HPC) platforms.

0.2 Numerical approach and research results

The numerical approach considers interior penalty discontinuous Galerkin (DG) methods for the incompressible Navier-Stokes equations in the framework of projection methods. The application of DG methods is popular due to their potentially high order of convergence, the inf-sup stability and local mass conservation. In addition to the 2×2 block structure arising from the saddle point system discontinuous Galerkin methods offer a further block structure when the unknowns associated per one cell of the mesh

are grouped together. This data structure is essential for high-performance implementations of DG methods as it avoids costly memory gather and scatter operations when compared to conforming finite element methods.

Operator splitting methods for solving the instationary Navier-Stokes equations have been subject to detailed investigations for the recent decades. One possibility in the splitting methods is to split the convective term leaving a saddle point system of the Stokes type. Another possibility is to split between incompressibility and dynamics which has been independently developed by Chorin [Cho68] and Témam [Té69] and is referred to as Chorin's projection method. The latter splitting schemes have the appealing feature that at each time step - instead of solving a saddle point system - only a Helmholtz equation for the velocity (in the Stokes case) and a Poisson equation for the pressure have to be solved. The choice of artificial boundary conditions on the pressure Poisson equation is a delicate issue in the projection methods of this class [Ran92, EL95, EL96]. In this thesis we concentrate on several higher-order extensions of Chorin's first order method that have been suggested in the literature [TMVDV96, GS03, GS04, KS05, GMS06].

The use of a DG spatial discretization within splitting schemes is a current subject of active research. Major contributions which are described in this thesis are: (I) We propose a modified upwind scheme based on the Vijayasundaram numerical flux that has favourable properties in the context of DG. The numerical flux is adapted from DG methods for inviscid compressible flows and takes into account that the velocity field need not be in $H(\mathrm{div})$. (II) We present novel postprocessing techniques in the Helmholtz projection step based on $H(\mathrm{div})$ reconstruction. A naive computation of the divergence-free velocity by subtraction of the (rotation-free) Helmholtz correction is reported to be unstable when the spatial mesh is coarse and the time step is small, see [SSL13, JDS$^+$16, KFWK17]. In these references several local postprocessing techniques are discussed to overcome this difficulty. In this thesis we propose new discrete types of solenoidal postprocessing: The first one is based on $H(\mathrm{div})$ reconstruction of the discrete pressure gradient, the second one is based on $H(\mathrm{div})$ reconstruction of the full Helmholtz flux. Both variants define a discrete projection operator, provide a velocity that satisfies the discrete continuity equation and is in consequence locally mass conservative. For the former variant also lower order $H(\mathrm{div})$ approximations can be used. The latter variant returns a velocity field that is moreover pointwise divergence-free. These properties are not satisfied by the postprocessing schemes available in the literature.

The transport equations for the tracers are also discretized by the DG method using upwinding. Numerical time integration for the model equations is fully implicit. Parallelization on the highest level (in terms of domain decomposition) is done by MPI where scalable communication infrastructure exists in the Distributed and Unified Numerics Environment (DUNE). Fast solution for in-time advancement requires optimal algorithms with respect to complexity. Efficient element-wise assembling of high-order DG methods can be accomplished on general quadrilateral/hexahedral grids by exploiting tensor product structure and using sum-factorization. Sum-factorization yields a reduction in the computational complexity, and in addition parallelization on a lower level is realized by SIMD vector instructions. This complexity reduction is particularly attrac-

tive when matrix-free solvers are used. The traditional approach however is to assemble the operator and perform dense matrix solves in each grid cell. This becomes first of all expensive because the cell-local matrices size as $\mathcal{O}((p+1)^d)$ where p denotes the polynomial degree and d the space dimension, and counteracts any improvements achieved in matrix-free methods. To overcome this issue, we present a new matrix-free implementation of stationary iterative methods where the block diagonal entries are inverted iteratively on-the-fly. It is well known that stationary iterative methods (and therefore the matrix-free versions) can be used as a preconditioner within Krylov subspace methods or as a smoother in a DG multigrid algorithm. The latter is used here for solving the pressure Poisson equation with an algebraic multigrid in the low order subspace where only low order components are explicitly assembled. This hybrid approach has shown effectiveness in the solution of a set of representative partial differential equations, [BMMP18].

We summarize the major contributions developed in this dissertation where the first two items have already been mentioned above.

- A modified upwind scheme for the inviscid flux which is based on the Vijayasundaram numerical flux and has favorable properties in the context of DG. The numerical flux is adapted from DG methods for compressible flows and takes into account that the velocity field need not be in $H(\mathrm{div})$.

- Postprocessing techniques in the Helmholtz projection step based on $H(\mathrm{div})$ reconstruction. Two variants are derived, the first one is based on Raviart-Thomas reconstruction of the discrete pressure gradient, the second one is based on Raviart-Thomas reconstruction of the Helmholtz flux. Both variants define a discrete projection operator as the returned velocity field satisfies the discrete continuity equation and is in consequence locally mass conservative. The velocity field in the latter variant is moreover pointwise divergence-free. These properties are not satisfied by the postprocessing schemes available in the literature.

- High-Performance implementation of the Spectral Discontinuous Galerkin Method (SDGM) and sum-factorization kernels in the exascale fork of DUNE (EXA-DUNE), as well as matrix-free nonlinear and linear solvers. This has been a multiple-person effort where the author has targeted in particular the application to incompressible Navier-Stokes and extension of interfaces.

- Development of a scalable parallel implementation which shows effectiveness up to $\sim$ 6100 cores and is able to achieve a significant fraction of node-level peak performance as well.

- Investigation and demonstration of the surface renewal effect in the Atmospheric Boundary Layer by direct numerical simulation with up to $\sim$ 19000 time steps taken.

0.3 Outline

The thesis is organized as follows: In Chapter 1 the model equations are presented which consist of the instationary Navier-Stokes equations in the Boussinesq limit. We give a brief overview of turbulence in fluids, where the aforementioned coherent structures arise, and in addition of turbulence theories. Then we summarize the surface renewal effect and give a description how such repeated patterns can be detected in a numerical simulation. This analysis will be utilized in a subsequent chapter to recorded DNS data.

In Chapter 2 we present the numerical discretization of the incompressible Navier-Stokes equations. Interior penalty DG methods are combined with the framework of projection methods. The above mentioned upwind scheme based on the Vijayasundaram numerical flux is derived. Moreover, the discrete Helmholtz projection operators are introduced and favorable properties like local mass conservation are shown. Numerical results demonstrate the properties of the discretization for different polynomial degrees applied to 2D and 3D problems with known analytical solution.

In Chapter 3 the solver for the Navier-Stokes Boussinesq system is discussed. The SDGM and sum-factorization technique are recapitulated at first, and then applied to the discretization from the previous chapter. We therefore present algorithms for linearized operator application that repeatedly arise in the nonlinear equations. The aforementioned matrix-free preconditioners are constructed from the Jacobians. Node-level performance as well as weak and strong scaling of the code is measured towards the end of this chapter.

In Chapter 4 we present simulation results of incompressible fluid flow. Problems range over two-dimensional and three-dimensional turbulence and we investigate the properties of our numerical discretization in these settings. We propose a model for numerical study of the surface renewal effect in the ABL. The model is then analyzed with the tools introduced in the first chapter. At the end we conclude in Chapter 5.

0.4 List of Notations

Navier-Stokes

∇f	Jacobian matrix of scalar-/vector-valued f, see Appendix A.1
$\mathbb{\nabla} f$	transposed Jacobian of scalar-valued f (equals column vector)
$\nabla^m f$	tensor containing m-th order derivatives, especially $\nabla^2 f$ denotes the Hessian
e_s	canonical basis vector in s-th direction
x_i	i-th component of a vector x. Subscripts $1, 2, \ldots$ usually refer to a component of a vector
d	physical space dimension
v	velocity
p	pressure
ρ	density

μ	dynamic viscosity
ν	kinematic viscosity, $\nu = \mu/\rho$
f	external volume force
$F(v)$	convective flux $F(v) = v \otimes v$
g	standard acceleration of gravity, unambiguously also Dirichlet boundary function g
θ	active scalar in the Navier-Stokes Boussinesq system, temperature (deviation)
c	passive scalar in the Navier-Stokes Boussinesq system
β	thermal expansion coefficient, unambiguously also switch for outflow condition type β
κ	thermal diffusivity, unambiguously also von Kármán constant κ
D	diffusivity of passive scalar
l	characteristic length scale
u	characteristic velocity
u_b	bulk velocity
κ	von Kármán constant
Ω	open domain in $\mathbb{R}^d$
n	unit outer normal vector
$\mathrm{d}x$	volume measure
$\mathrm{d}s$	surface measure
Re	Reynolds number
Cr	Courant number
Pe	Peclet number
Ra	Rayleigh number
Pr	Prandtl number
Ro	Rossby number
Sc	Schmidt number
$\mathbf{Re}_\tau$	friction Reynolds number
$\mathbf{Re}_b$	bulk Reynolds number
S	strain tensor $S = \frac{1}{2}(\nabla v + \nabla v^T)$
ζ	vorticity $\zeta = \nabla \times v$
Z	vorticity tensor $\mathrm{Z} = \nabla v - S = \frac{1}{2}(\nabla v - \nabla v^T)$
Q	second invariant of ∇v, $Q = \frac{1}{2}(\mathrm{Z} : \mathrm{Z} - S : S)$
λ_2	second largest eigenvalue of $S^2 + \mathrm{Z}^2$

$E(t)$	total energy of Navier-Stokes Boussinesq system, kinetic energy in incompressible Navier-Stokes
$\epsilon(t)$	dissipation rate, $\epsilon(t) = -\frac{\mathrm{d}E(t)}{\mathrm{d}t}$
$\mathcal{E}(t)$	enstrophy
$H(t)$	helicity
$\bar{v}$	background velocity
$\tilde{v}$	variation of background velocity
$\bar{p}$	background pressure
$\tilde{p}$	variation of background pressure
$\langle\cdot\rangle$	averaging procedures, e.g. ensemble averages, temporal averages
$\mathcal{F}v$	Fourier transform of v
k	wave vector magnitude, wavenumber
$E(k,t)$	(kinetic) energy spectrum

Finite element discretization

$\hat{E}$	volume reference element $(0,1)^d$
E	cell of mesh $\mathcal{E}_h$
μ_E	transformation $\hat{E} \to E$
$\hat{e}$	face reference element $(0,1)^{d-1}$
e	facet
μ_e	transformation $\hat{e} \to e$
n_e	unit outer normal vector w.r.t. orientation on face e
$\hat{x}$	coordinates in $\hat{E}$
x	coordinates in E, Ω
$\hat{s}$	coordinates in $\hat{e}$
s	coordinates in e
E^{int}	interior element w.r.t. orientation on face e
E^{ext}	exterior element w.r.t. orientation on face e
h	maximum diameter of mesh $\mathcal{E}_h$
$\mathbf{Re}_h$	grid Reynolds number
$\mathbf{Pe}_h$	grid Peclet number
p	polynomial degree, pressure p then becomes p_h
$\mathbb{Q}_{p,d}$	set of polynomials of maximum degree p in d variables
Q_h^p	scalar valued DG space of polynomial degree p

X_h^p	DG space for the velocity
M_h^{p-1}	DG space for the pressure
g	velocity Dirichlet boundary function
$\hat{F}_e(v, n_e)$	numerical flux of $F(v)$ on face e
β	switch between CDN and DDN
$a(\cdot,\cdot)$	DG variational form, $a = d + J_0$, for viscous term in momentum equation
$d(\cdot,\cdot)$	volume and consistency term
$J_0(\cdot,\cdot)$	symmetrization and penalty term
$b(\cdot,\cdot)$	left-hand side of continuity equation in DG
$c(\cdot,\cdot)$	variational form c for the convective part, concentration c then becomes c_h
$l(\cdot;t)$	right-hand side linear form in momentum equation
$r(\cdot;t)$	right-hand side linear form in continuity equation
$s_o(\cdot,\cdot)$	variational form for the DDN contribution
h_e	mesh dependence on face e in penalty factor
ϵ	symmetrization, anti-symmetrization or none, $\epsilon \in \{-1, 1, 0\}$ (SIPG, NIPG, IIPG), unambiguously dissipation rate $\epsilon(t)$
$\alpha(\cdot,\cdot)$	DG variational form, $\alpha = b + j_0$, for pressure-Poisson equation
$j_0(\cdot,\cdot)$	symmetrization and penalty term for pressure-Poisson equation
ϕ, ϕ_h	test functions $\in Q_h^p$
φ, φ_h	test functions $\in X_h^p$
q, q_h	test functions $\in M_h^{p-1}$
$\hat{v}_h, \hat{p}_h, \ldots$	hat denotes functions on the reference element
$\mathcal{P}_h$	discrete Helmholtz projection operator
ψ, ψ_h	correction in Helmholtz decomposition
δv_h	update/correction (from e.g. Newton's method) to linearization point v_h
$z_h, \delta z_h$	coefficient vectors belonging to v_h, δv_h
t^k	time index k
v_h^k, p_h^k	approximate velocity, pressure in corresponding DG spaces at time t^k
M_h	velocity mass matrix
$\mathcal{L}_h(\cdot, t)$	spatial nonlinear residual vector in discretized momentum equation
$\mathrm{\alpha}_{ij}, \mathrm{\beta}_{ij}, \mathrm{\delta}_j$	coefficients of spatial and temporal Butcher subtableau (upright Greek letters)
$A_h(z_h)$	Jacobian matrix of $\mathcal{L}_h$ evaluated at z_h: $A_h(z_h) = \nabla_z \mathcal{L}_h(z,t)\big\vert_{z=z_h}$
$A_h(z_h)\delta z_h$	linearized operator application (matrix-vector product)

$a'(v_h, \varphi)$	Fréchet derivative of $a(v_h, \varphi)$ with respect to v_h
$a'(v_h, \varphi)\delta v_h$	linearized operator application (in finite element space X_h^p here)

1 Turbulence, coherent structures and surface renewal

This chapter collects the physical processes that are revisited in Chapter 4 and used for discussion of numerical simulation results. Its outline is summarized as follows: Section 1.1 presents the governing equations that are the instationary incompressible Navier-Stokes equations under the Boussinesq approximation. In Section 1.2 we give a brief overview of turbulence in fluids, the transition to turbulence and the mathematical tools required for description. Section 1.3 gives a description of the surface renewal effect - repeated patterns caused by turbulent coherent structures - and its analysis. In Section 1.4 we review several vortex identification techniques that serve as visualization of turbulent flows in numerical simulations.

1.1 Navier-Stokes Boussinesq system

The Navier-Stokes equations follow from Newton's equations of motion and the conservation of mass. They form a fundamental model of non-relativistic fluid dynamics. In this thesis we only consider a simplified form of the Navier-Stokes equations that are the incompressible Navier-Stokes equations where density variations are included by the Boussinesq approximation, [Les08, ESW14]. Therein, density is linearized around the value in the incompressible limit and variations only come into play by temperature. The dependence is written as a source term that only acts in the direction of gravity.

The Navier-Stokes Boussinesq system in the space-time cylinder $\Omega \times (0, t_f]$ ($\Omega \subset \mathbb{R}^d$ open, $d = 2, 3$) for the velocity v, pressure p, temperature T and tracer concentration c, with given density, viscosity ρ, μ and external forcing f reads

$$\rho \partial_t v - \mu \Delta v + \rho (v \cdot \nabla) v + \nabla p = f + \rho \beta \theta g e_d - \rho \omega_{Cor} e_d \times v \tag{1.1}$$

$$\nabla \cdot v = 0 \tag{1.2}$$

$$\partial_t \theta - \kappa \Delta \theta + \nabla \cdot (\theta v) = 0 \tag{1.3}$$

$$\partial_t c - D \Delta c + \nabla \cdot (c v) = 0 \, . \tag{1.4}$$

As described above the density variations are written as a forcing term, $\rho(1 - \beta(T - T_{ref}))ge_d$, where β is called the thermal expansion coefficient. Note that the potential energy $\rho g x_d$ is already included in the pressure. The Coriolis force $e_d \times v$ is a term that only appears in three dimensions. $\theta = T - T_{ref}$ is the temperature deviation from a reference temperature. In the equation for θ, κ is the thermal diffusivity. Since θ feeds back the velocity through the buoyancy forcing term, it is called an *active scalar*. One

can also consider the evolution of a *passive scalar* c. It satisfies the same type of equation as θ, with possibly different diffusivity D, but has no influence on the evolution of v. This set of equations is supplemented with initial conditions, and boundary conditions if Ω is bounded, for the velocity, active and passive scalars.

The total energy of the velocity field (with no external forcing, $f = 0$) and the squared L^2-norm of a scalar (here active) are

$$E(t) = \int_\Omega \frac{1}{2} v(x,t) \cdot v(x,t) - g\beta\theta(x,t)x_d \mathrm{d}x \ , \quad E_\theta(t) = \frac{1}{2}\int_\Omega \theta(x,t)^2 \mathrm{d}x. \tag{1.5}$$

The dissipation rate of the velocity field or an advected scalar is the negative temporal derivative of the energies defined above. With the help of Reynolds transport theorem one can derive evolution equations for $\epsilon(t)$ and $\epsilon_\theta(t)$, respectively:

$$\epsilon(t) = -\frac{\mathrm{d}E(t)}{\mathrm{d}t} = \int_\Omega \nu \nabla v(x,t) : \nabla v(x,t) + g\beta x_d \kappa \Delta\theta(x,t) \mathrm{d}x \tag{1.6}$$

$$\epsilon_\theta(t) = -\frac{\mathrm{d}E_\theta(t)}{\mathrm{d}t} = \int_\Omega \kappa \nabla\theta(x,t) \cdot \nabla\theta(x,t) \mathrm{d}x. \tag{1.7}$$

These integral quantities of the flow can only decrease in time, except from the source term $g\beta x_d \kappa \Delta\theta$, and they measure the loss of energy through internal friction which will become important later in the mathematical description of turbulence 1.2. In order to describe the behavior of this system one associates a characteristic velocity denoted by u and a characteristic length scale denoted by l. Together with the material properties μ, ρ, κ one can identify dimensionless quantities that measure the relative size of the various terms in the Navier-Stokes Boussinesq system. This also leads to the well-known procedure of non-dimensionalization of the Navier-Stokes Boussinesq system. We list and explain here the dimensionless quantities that are going to be used in this thesis.

Reynolds number The Reynolds number Re is defined as

$$\mathrm{Re} = \frac{\rho u l}{\mu} = \frac{u l}{\nu} \tag{1.8}$$

where u and l are the characteristic velocity and length scale of the flow. Here μ is called the *dynamic viscosity* and $\nu = \mu/\rho$ the *kinematic viscosity*. It measures the ratio between inertia and friction. In turbulent flows ($\mathrm{Re} \gg 1$) the frictional term is small, at least at length scale l. The dominance of the convective term leads to nonlinear dynamics, creation of complex flow structures and a chaotic evolution. Paradoxically, however, the dissipation term (1.6) controls the energies of the system. Thus, there must be a profound difference in solutions between the asymptotic tendency as $\mathrm{Re} \to \infty$, and the Euler limit $\mathrm{Re} = \infty$ or $\mu = 0$. The difference is that as long as $\mathrm{Re} < \infty$, there are small scales at which friction becomes important and Re is small.

Peclet number The Peclet number Pe is defined as

$$\mathrm{Pe} = \frac{ul}{D} \tag{1.9}$$

and the direct analogy of Re for an active or passive scalar with diffusivity D, or κ as well. It measures the relative importance between advection and diffusion. At large Pe, the evolution is dominated by advection. Once more, the limit $\mathrm{Pe} \to \infty$ is very different from $\mathrm{Pe} = \infty$, because dissipation, no matter how small, is eventually responsible for removing structure from the scalar field.

Prandtl number The Prandtl number Pr is defined as

$$\mathrm{Pr} = \frac{\nu}{\kappa} \tag{1.10}$$

and measures the ratio of friction and thermal diffusivity. It is a property of the fluid, not of the particular flow. The Peclet number can be obtained together with the Reynolds number, $\mathrm{Pe} = \mathrm{Re}\,\mathrm{Pr}$.

Schmidt number The Schmidt number Sc is defined as

$$\mathrm{Sc} = \frac{\nu}{D} \tag{1.11}$$

and specifies the ratio of friction and diffusion D. It is a property of a passive scalar that is advected in the velocity field. The corresponding Peclet number can be obtained together with the Reynolds number, $\mathrm{Pe} = \mathrm{Re}\,\mathrm{Sc}$.

Rayleigh number The Rayleigh number Ra is defined as

$$\mathrm{Ra} = \frac{g\beta\ \delta T\ l^3}{\nu\kappa} \tag{1.12}$$

and measures the ratio of buoyancy and diffusion. In the formula for the Rayleigh number δT denotes the temperature difference (e.g. between a hot and cold plate that enclose the fluid) and shall not be confused with the temperature deviation θ. In buoyancy driven flows the characteristic velocity is given by $u = \sqrt{g\beta\ \delta T\ l}$ and thus, together with the Prandtl number, Reynolds and Peclet number can be retrieved:

$$\mathrm{Re} = \sqrt{\mathrm{Ra}/\mathrm{Pr}}\,, \quad \mathrm{Pe} = \sqrt{\mathrm{Ra}\ \mathrm{Pr}}\,. \tag{1.13}$$

Rossby number The Rossby number is defined as

$$\mathrm{Ro} = \frac{u}{\omega_{Cor} l} \tag{1.14}$$

and measures the relative importance of inertia and the Coriolis force. Ro describes the effects of a rotating reference system: $\mathrm{Ro} \gg 1$ characterizes essentially non-rotating turbulence, while $\mathrm{Ro} \leq 1$ flows are strongly affected by rotation.

1.1.1 Non-dimensional system

With the introduction of the non-dimensional quantities above, it is convenient to normalize the Navier-Stokes Boussinesq system with help of the characteristic velocity and length scale. This calculation is commonly known as *non-dimensionalization*, [ESW14, Ran17], and shall not be repeated here in detail. We only present now the final form of the normalized system and for ease of writing, the subscript indicating non-dimensional quantities will be omitted from this point on:

$$\partial_t v - \frac{1}{\mathrm{Re}}\Delta v + (v\cdot\nabla)v + \nabla p = f + \theta e_d - \frac{1}{\mathrm{Ro}} e_d \times v \tag{1.15}$$

$$\nabla \cdot v = 0 \tag{1.16}$$

$$\partial_t \theta - \frac{1}{\mathrm{Re\ Pr}}\Delta\theta + \nabla\cdot(\theta v) = 0 \tag{1.17}$$

$$\partial_t c - \frac{1}{\mathrm{Re\ Sc}}\Delta c + \nabla\cdot(cv) = 0\ . \tag{1.18}$$

Again this set of equations needs to be supplemented with initial and boundary conditions for the velocity, active and passive scalars. Unless stated explicitly otherwise we are only going to consider flows in non-rotating frames of reference, i.e. $\omega_{Cor} = 0$ or $\mathrm{Ro} = \infty$. Buoyancy driven flows are already characterized by specifying the Rayleigh and Prandtl number because of (1.13). In the non-Boussinesq limit the Reynolds number is calculated from (1.8).

1.1.2 Properties of turbulent flows

As pointed out in Uriel Frisch's book on turbulence [Fri95], the Navier-Stokes equations probably contain all of turbulence. In the upcoming section we want to list as a start properties of turbulent flows before giving a mathematical description in Section 1.2. It is physically expected that for any positive finite viscosity the solution of the Navier-Stokes equations is sufficiently smooth. This justifies the existence of higher order derivatives of the velocity field and pressure, integration by parts and the permutation of mixed partial derivatives - therefore most of the preparatory steps in Section 1.1.3.

Broadband spectrum in space and time

Turbulent flows are characterized by structures on a broad range of spatial and temporal scales, even given smooth or periodic initial condition and forcing. In other words, turbulent flows have broadband spectrum both in time and space. If l is again the characteristic length scale (that describes the largest scale motions) and if l_d is the length scale of the smallest motions in a flow, then a large range of spatial scales implies $l \gg l_d$. The scale l_d is typically the scale at which dissipation becomes important and removes energy from the flow. It will play an important role in theories of turbulence to be described later. The scale l, instead, is set by the forcing mechanisms from the large-scale flow. We are going to see later that the ratio l/l_d can be related to the Reynolds number.

Dominated by convective nonlinearity

A field of non-interacting linear internal waves with many different frequencies and wave-numbers can also have a large range of length scales, but it is not turbulent. In a turbulent flow the different scales do interact through the nonlinear term in the equation of motion. It is the convection term that redistributes energy among the various scales of motion without affecting the global energy budget. Thus the broadband spectrum appears as a result of the internal dynamics. In a field of linear internal waves, instead, the broadband spectrum is generated by external controls like forcing, initial or boundary conditions and the different modes have no coupling between each other.

Unpredictable in space and time

Turbulent flows are predictable for only short times and small distances. Even though we know the equations that govern the evolution of the fluid, we cannot make predictions about the details of the flow due to its sensitive dependence on initial and boundary conditions. This sensitive dependence is once more a result of the strong nonlinearity of the model. Predictability, however, can be recovered in a statistical sense, as will be pointed out in Section 1.2.2. Sensitivity on initial and boundary conditions is a fundamental property of *chaotic* systems. But turbulence and chaos are not synonymous, turbulent motions are indeed chaotic, but chaotic motions need not be turbulent. Chaos may involve only a small number of degrees of freedom, i.e. it can be narrow band in space and/or time. There are numerous examples of chaotic systems characterized by temporal complexity, but spatial simplicity, like the Lorenz's system. Turbulence is different, because it is always complex both in space and time.

Time reversible and irreversible

Flow at low Reynolds numbers exhibits time reversibility. Turbulent motions, as time goes on however, tend to forget their initial conditions and reach some equilibrated state. This fact is neither incompatible with time reversibility nor time irreversibility. For example, the Euler equations are time reversible and still turbulence mixes stuff up, it does not unmix it. The Navier-Stokes equations, unlike the Euler equations, are time irreversible due to the presence of internal friction and no matter how small viscosity is, it will still become important at small scales. This once again points out that flow at $\mathrm{Re} \gg 1$ is fundamentally different than flow at $\mathrm{Re} = \infty$.

1.1.3 Nomenclature and future preparations

We start this section by introducing quantities frequently used to describe the motion in fluid mechanics:

The strain tensor S is the symmetric part of the velocity Jacobian ∇v,

$$S = \frac{1}{2}(\nabla v + \nabla v^T) \; . \tag{1.19}$$

The vorticity tensor Z is the anti-symmetric part of the velocity jacobian,

$$\mathrm{Z} = \nabla v - S = \frac{1}{2}(\nabla v - \nabla v^T) \; . \tag{1.20}$$

The vorticity is denoted by $\zeta = \nabla \times v$ whose components are also in the off-diagonal of Z. Note that we choose the letter Z for the vorticity tensor to avoid a conflicting notation with the domain Ω and therefore, to be consistent, the letter ζ for the curl of the velocity field. Furthermore, note that at first the curl operator is defined in three dimensions, but it can be extended to two dimensions: The curl of 2D vector fields is interpreted as the 3D curl applied to such vector fields with vanishing third component. The result can thus be viewed as a scalar function. The curl of 2D scalar functions is interpreted as the 3D curl applied to e_3 times the scalar function. The result can thus be viewed as a two-dimensional vector.

Next, the enstrophy of a velocity field is defined as

$$\mathcal{E}(t) = \frac{1}{2}\int_\Omega (\nabla \times v(x,t)) \cdot (\nabla \times v(x,t)) \mathrm{d}x = \frac{1}{2}\int_\Omega \zeta(x,t) \cdot \zeta(x,t) \mathrm{d}x \tag{1.21}$$

and the helicity of a velocity field is defined as

$$H(t) = \frac{1}{2}\int_\Omega v(x,t) \cdot (\nabla \times v(x,t)) \mathrm{d}x = \frac{1}{2}\int_\Omega v(x,t) \cdot \zeta(x,t) \mathrm{d}x \; . \tag{1.22}$$

Based on the remark above, the helicity vanishes in two dimensions because the vorticity points to $\zeta \parallel e_3$ and is orthogonal to the velocity field. In Equation (1.5) the integrand without the potential energy term is called kinetic energy density, in Equations (1.21) and (1.22) the integrands are called enstrophy density and helicity density, respectively:

$$E(x,t) = \frac{1}{2}\left\|v(x,t)\right\|_2^2 \tag{1.23}$$

$$\mathcal{E}(x,t) = \frac{1}{2}\left\|\nabla \times v(x,t)\right\|_2^2 \tag{1.24}$$

$$H(x,t) = \frac{1}{2}H_d = \frac{1}{2}v(x,t) \cdot (\nabla \times v(x,t)) \; . \tag{1.25}$$

The nonlinear convection term has multiple representations, for instance

$$\begin{aligned} (v \cdot \nabla)v &= \nabla v \, v \\ &= \frac{1}{2}\nabla(v \cdot v) - v \times (\nabla \times v) \\ &= \nabla v^T v - v \times (\nabla \times v) \; . \end{aligned} \tag{1.26}$$

The representation involving the cross product depictly demonstrates the fundamentally different behavior between two-dimensional and three-dimensional turbulence. For the following preparations we list some helpful identities from vector calculus: For v, w vector fields, sufficiently smooth, it holds (also in 2D)

$$-\Delta v = \nabla \times (\nabla \times v) - \nabla(\nabla \cdot v) \tag{1.27}$$

$$\nabla \times (v \times w) = (\nabla \cdot w)v + (w \cdot \nabla)v - (\nabla \cdot v)w - (v \cdot \nabla)w \tag{1.28}$$
$$\nabla \cdot (w \times v) = v \cdot \nabla \times w - w \cdot \nabla \times v \; . \tag{1.29}$$

The first identity is known as the *rotational form of the Laplacian*. The second is the curl of a cross product which helps below to calculate the curl of the convection term. The last identity states how the divergence of a cross product can be expanded.

The enstrophy can also be brought in connection to the kinetic energy, dissipation rate. This is based on an observation regarding divergence and curl of vector fields - an orthogonal decomposition of the Jacobian that is also related to the Helmholtz decomposition which we review in detail in Section 2.2.1. Therefore let v be a (sufficiently smooth) vector field that vanishes on $\partial\Omega$ for Ω being bounded or that is periodic within a box Ω, then one can show with integration by parts and formulas above

$$\int_\Omega \nabla v : \nabla v \mathrm{d}x = \int_\Omega (\nabla \cdot v)^2 \mathrm{d}x + \int_\Omega (\nabla \times v) \cdot (\nabla \times v) \mathrm{d}x \; . \tag{1.30}$$

As a consequence, we have for incompressible flow fields

$$-\frac{\mathrm{d}E(t)}{\mathrm{d}t} = \epsilon(t) = 2\nu\mathcal{E}(t) \; . \tag{1.31}$$

Also for incompressible velocity fields the pressure in the Navier-Stokes equations can be recovered by a Poisson equation,

$$-\Delta p = \rho \nabla \cdot ((v \cdot \nabla)v) = \rho \nabla v^T : \nabla v \tag{1.32}$$

for divergence-free f. The momentum and mass conservation equation can be expressed in different variables than the current *primitive variables* (v, p). With equation (1.28) the curl of the convection term can be transformed to $\nabla \times (v \cdot \nabla)v = -\nabla \times (v \times \zeta) = -(\zeta \cdot \nabla)v + (v \cdot \nabla)\zeta$. When applying the curl to the whole momentum equation, we can readily derive

$$\begin{aligned} \rho\partial_t \zeta - \mu\Delta\zeta + \rho\nabla \times (\zeta \times v) &= 0 \\ -\Delta v &= \nabla \times \zeta \end{aligned} \tag{1.33}$$

for irrotational f. This system corresponds to the formulation of the incompressible Navier-Stokes equations in the so-called *velocity-vorticity variables*. The counterpart to the momentum equation is the vorticity transport equation.

The evolution of enstrophy density can be obtained by projecting (1.33) onto ζ:

$$\rho\partial_t \mathcal{E} + \nabla \cdot (\rho v\mathcal{E} - \mu\nabla\mathcal{E}) = \rho\zeta^T S\zeta - \mu\nabla\zeta : \nabla\zeta \; . \tag{1.34}$$

The first term on the right-hand side is called the enstrophy production term. It vanishes in 2D which only leaves $\nabla\zeta : \nabla\zeta$, called the *palinstrophy density*, on the right-hand side. This has important consequences for two-dimensional turbulence that we will touch on in Section 1.2.5. The remark stated can further be understood since in two dimensions

the *vortex-stretching term* $(\zeta \cdot \nabla)v$ vanishes. The vorticity transport equation (1.33) simplifies to

$$\rho\partial_t\zeta - \mu\Delta\zeta + \rho(v \cdot \nabla)\zeta = 0 \,. \tag{1.35}$$

As a consequence the vorticity of an inviscid fluid is conserved in two dimensions. In fact any differentiable function in ζ is a conserved quantity. For the second-order monomial we obtain that the enstrophy is time-independent in 2D. This can also be seen from the evolution of enstrophy density, again for $\nu \neq 0$ below - by either leaving out the production term $\zeta^T S\zeta$ in (1.34) or by multiplying (1.35 with ζ:

$$\partial_t\mathcal{E} + \nabla \cdot (v\mathcal{E} - \nu\nabla\mathcal{E}) = -\nu\nabla\zeta : \nabla\zeta \,. \tag{1.36}$$

1.2 Turbulence

1.2.1 Transition to turbulence

We have mentioned that a property of turbulent flows is the sensitivity on initial and boundary conditions. Small disturbances lead to a drastically different flow field as time goes on. Experiments observe that for example stable planar background flows such as Hagen-Poiseuille flow or Couette flow show transition to turbulence at a critical Reynolds number. In order to understand this transition to turbulence we recall here the linear stability theory. We follow the presentation of [CHQZ07, Les08]. Towards the end of this section we furthermore give a deterministic model for chaos, [Fri95].

Linear perturbation theory

Rayleigh (1880) initiated the development of incompressible, inviscid linear stability theory. He established this theory as an eigenvalue problem by a second-order ordinary differential equation for the amplitude of the disturbance, with the disturbance wavenumber and frequency as parameters. This theory was followed by the viscous linear stability analysis which became fairly successful because it validated the observations from experiments and it is able to predict the critical value of the Reynolds number at which instability commences.

We start the presentation of the linear stability theory in the more general setting of linear perturbation theory. Therein the solution (v, p) of the Navier-Stokes equations is written as $v = \bar{v} + \tilde{v},\ p = \bar{p} + \tilde{p}$. The background state $(\bar{v}, \bar{p})$ satisfies

$$\begin{aligned} \rho\partial_t\bar{v} - \mu\Delta\bar{v} + \rho(\bar{v} \cdot \nabla)\bar{v} + \nabla\bar{p} &= f \\ \nabla \cdot \bar{v} &= 0 \end{aligned}$$

and $(\tilde{v}, \tilde{p})$ is a perturbation to the background flow. The Navier-Stokes equations for (v, p) are linearized around $(\bar{v}, \bar{p})$ assuming that the perturbation is small compared to $\bar{v}$, i.e. $\|\tilde{v}\|_2 / \|\bar{v}\|_2 \ll 1$.

$$\rho\partial_t(\bar{v} + \tilde{v}) - \mu\Delta(\bar{v} + \tilde{v}) + \rho(\bar{v} + \tilde{v}) \cdot \nabla(\bar{v} + \tilde{v}) + \nabla(\bar{p} + \tilde{p}) = f$$

$$\nabla \cdot (\bar{v} + \tilde{v}) = 0 \, .$$

Since $\|\tilde{v}\|_2 / \|\bar{v}\|_2 \ll 1$ the quadratic term in $\tilde{v}$ can be dropped in the expansion. We then obtain the linearized Navier-Stokes equations for the perturbation $(\tilde{v}, \tilde{p})$:

$$\rho \partial_t \tilde{v} - \mu \Delta \tilde{v} + \rho(\bar{v} \cdot \nabla)\tilde{v} + \rho(\tilde{v} \cdot \nabla)\bar{v} + \nabla \tilde{p} = 0 \tag{1.37}$$

$$\nabla \cdot \tilde{v} = 0 \, . \tag{1.38}$$

Two-dimensional Orr-Sommerfeld equation

Now let us assume that the background velocity takes the form

$$\bar{v} = \bar{v}_1(x_2) e_1$$

(this is also the starting point for the linear stability theory in three dimensions). In two dimensions one can write the velocity perturbation in the stream function formulation, $\psi(x_1, x_2, t) = \phi(x_2) \exp(i(\alpha x_1 - \lambda t))$, such disturbances are called Tollmien-Schlichting waves. The components of the perturbation can be retrieved as

$$\tilde{v} = \left(\frac{\partial \psi}{\partial x_2}, -\frac{\partial \psi}{\partial x_1} \right)^T \text{ and therefore } \nabla \cdot \tilde{v} = 0 \, .$$

With this particular form of the background velocity, equation (1.37) simplifies to

$$\begin{aligned} \rho \partial_t \tilde{v}_1 + \rho \bar{v}_1 \partial_{x_1} \tilde{v}_1 + \rho \tilde{v}_2 \partial_{x_2} \bar{v}_1 &= -\partial_{x_1} \tilde{p} + \mu \Delta \tilde{v}_1 \\ \rho \partial_t \tilde{v}_2 + \rho \bar{v}_1 \partial_{x_1} \tilde{v}_2 &= -\partial_{x_2} \tilde{p} + \mu \Delta \tilde{v}_2 \, . \end{aligned}$$

Now, the perturbation is written in terms of the stream function which yields

$$\begin{aligned} -\rho e^{i(\alpha x_1 - \lambda t)} \left[-i\lambda \partial_{x_2} \phi + i\alpha \bar{v}_1 \partial_{x_2} \phi - i\alpha \phi \partial_{x_2} \bar{v}_1 - \nu \left(-\alpha^2 \partial_{x_2} \phi + \partial_{x_2}^3 \phi \right) \right] &= \partial_{x_1} \tilde{p} \\ -\rho e^{i(\alpha x_1 - \lambda t)} \left[-\alpha \lambda \phi + \bar{v}_1 \alpha^2 \phi - \nu \left(i\alpha^3 \phi - i\alpha \partial_{x_2}^2 \phi \right) \right] &= \partial_{x_2} \tilde{p} \, . \end{aligned}$$

Eliminate the pressure fluctuation by differentiating the x_1- and x_2-momentum equation by x_2 and x_1, respectively.[1] Combining those two equations gives

$$-i\lambda \partial_{x_2}^2 \phi + i\alpha \bar{v}_1 \partial_{x_2}^2 \phi - i\alpha \phi \partial_{x_2}^2 \bar{v}_1 + \nu \left(2\alpha^2 \partial_{x_2}^2 \phi - \partial_{x_2}^4 \phi - \alpha^4 \phi \right) + i\alpha^2 \lambda \phi - i\bar{v}_1 \alpha^3 \phi = 0 \, .$$

In the last step we rearrange the terms to arrive at the final form of the *Orr-Sommerfeld equation*:

$$\left(\frac{\mathrm{d}^2}{\mathrm{d}x_2^2} - \alpha^2 \right)^2 \phi - \frac{i\alpha}{\nu} \bar{v}_1 \left(\frac{\mathrm{d}^2}{\mathrm{d}x_2^2} - \alpha^2 \right) \phi + \frac{i\alpha}{\nu} \frac{\mathrm{d}^2 \bar{v}_1}{\mathrm{d}x_2^2} \phi = -\frac{i\lambda}{\nu} \left(\frac{\mathrm{d}^2}{\mathrm{d}x_2^2} - \alpha^2 \right) \phi \, . \tag{1.39}$$

It is a fourth-order equation for the amplitude of the disturbance and needs to be supplemented with the boundary conditions $\phi = \mathrm{d}\phi/\mathrm{d}x_2 = 0$ on the endpoints in x_2-direction.

[1] Note that the expression in the rectangular brackets of the x_2-momentum equation does not depend on x_1.

The Orr-Sommerfeld equation contains the information on linear stability for viscous, incompressible flows. If equation (1.39) is multiplied with ν and afterwards $\nu = 0$ is set, it simplifies to the *Rayleigh equation* for inviscid, incompressible flows. We briefly remark that with the same type of perturbation in 3D for the components $\tilde{v}_1, \tilde{v}_2, \tilde{v}_3$, the similar three-dimensional Orr-Sommerfeld equation for the amplitude can be derived.

Equation (1.39) describes a dispersion relation between α and λ with Re as a parameter. If λ is fixed the problem approach is called *spatial stability*. However if α is fixed this problem is one of *temporal stability* and the dispersion relation constitutes a generalized eigenvalue problem for λ. In this stability analysis one searches for complex eigenvalues λ with $\Im\lambda > 0$ to obtain a mode that grows in time and hence the background solution is unstable. In the generalized eigenvalue problem the continuous linear left-hand side operator

$$A = \left(\frac{\mathrm{d}^2}{\mathrm{d}x_2^2} - \alpha^2\right)^2 - \frac{i\alpha}{\nu}\bar{v}_1\left(\frac{\mathrm{d}^2}{\mathrm{d}x_2^2} - \alpha^2\right) + \frac{i\alpha}{\nu}\frac{\mathrm{d}^2\bar{v}_1}{\mathrm{d}x_2^2}$$

and right-hand side operator

$$B = -\frac{i}{\nu}\left(\frac{\mathrm{d}^2}{\mathrm{d}x_2^2} - \alpha^2\right)$$

must be discretized with high numerical accuracy.

Returning to the transition to turbulence, we now want to solve equation (1.39) for the Hagen-Poiseuille channel flow with $\bar{v}_1(x_2) = 1 - x_2^2,\ x_2 \in (-1, 1)$ for different Reynolds numbers $\mathrm{Re} = 1/\nu$. We utilize the Orr-Sommerfeld eigenvalue solver implemented in [LMW12, Tre06] that uses spectral collocation with Chebyshev polynomials for the representation of ϕ. Figure 1.1 shows the distribution of the eigenvalues for the Reynolds numbers 2000, 4000, 8000 with $\alpha = 1$ and 60 Chebyshev points. The eigenvalue with largest imaginary part is marked with the corresponding value in the plots. For Reynolds numbers 2000 and 4000 observe that all λ's have a negative imaginary part. At Reynolds number 8000 the transition of the background flow to turbulence happens and there is a single eigenvalue with positive imaginary part, $\lambda = 0.24708 + 0.00266i$. From the associated eigenfunction ϕ the velocity perturbation can be calculated as $(\partial_{x_2}\psi, -\partial_{x_1}\psi)$ and then by taking the real part. Figure 1.2 shows the velocity disturbance where the unstable mode ϕ was scaled to have a maximum of 10^{-4}. It can be seen that the peak is located at the walls of the channel and therefore this particular mode is called a wall mode, [CHQZ07].

A deterministic model for chaos

In this section let us study the following map

$$\begin{aligned} \dot{\nu} &= - A\nu^2 - \mathrm{Re}^{-1}k^2\nu \\ \partial_t v &= -((v\cdot\nabla)v + \nabla p) + \mathrm{Re}^{-1}\Delta v + f \end{aligned} \tag{1.40}$$

with the Navier-Stokes momentum equation written directly underneath. [Fri95] presents a similar map for ν which is here the special case of $\mathrm{Re} = \infty$ as we will see later. The

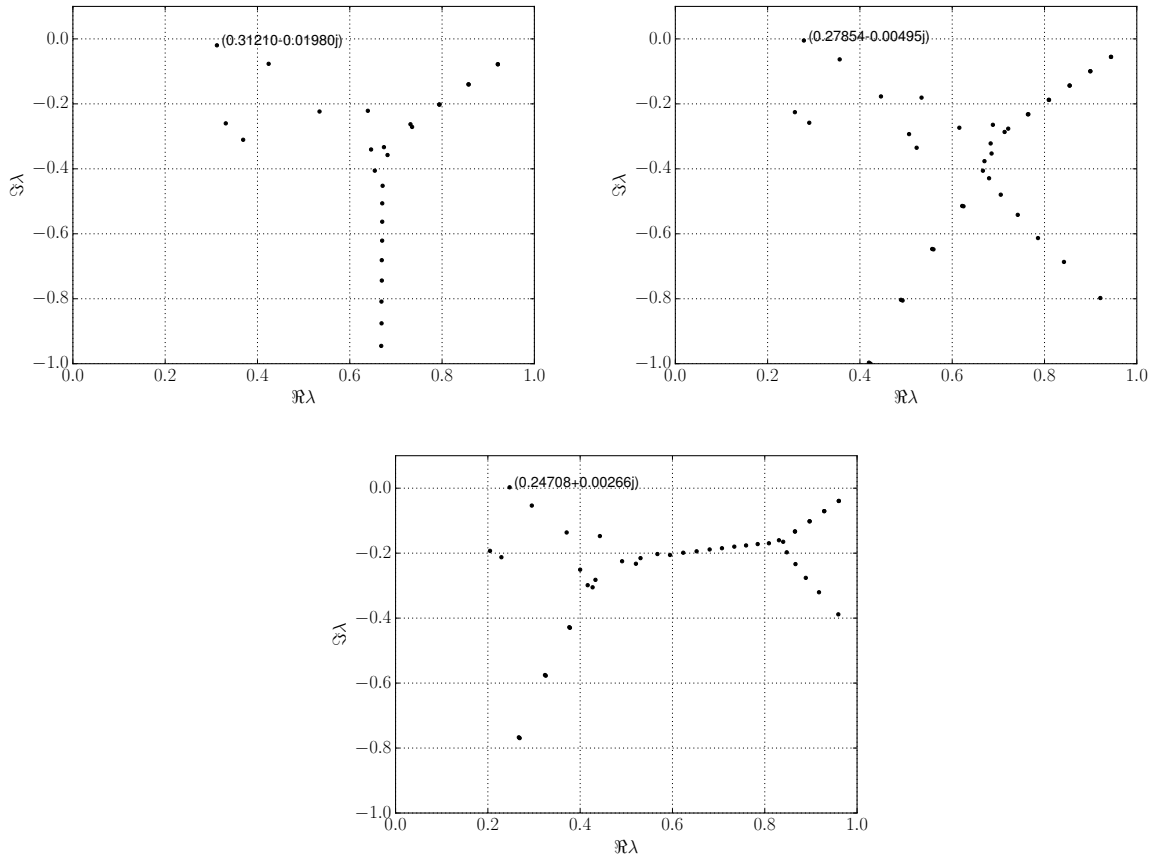


Figure 1.1: Distribution of the eigenvalues in the Orr-Sommerfeld problem for Reynolds numbers 2000 (top left), 4000 (top right) and 8000 (bottom).

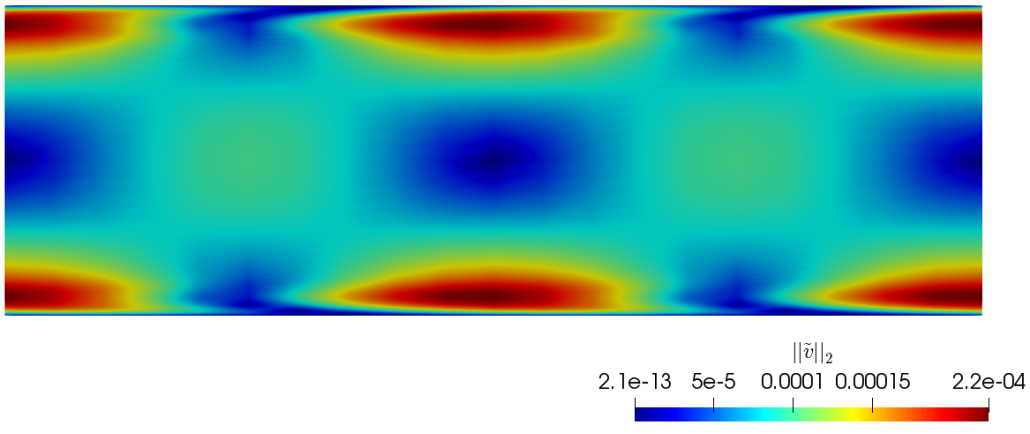


Figure 1.2: Magnitude of the velocity perturbation from the single unstable mode at Re = 8000. The mode was scaled to have a maximum of 10^{-4}.

evolution of v is called the *logistic map* or the *poor man's Navier-Stokes equation* that mimics some of the properties of the Navier-Stokes equations but has no spatial structure.

Straightforward the explicit Euler method is used to solve (1.40) numerically:

$$\frac{v_{n+1} - v_n}{\tau} = -Av_n^2 - \mathrm{Re}^{-1}k^2 v_n \ .$$

The equation can be solved for v_{n+1} and under appropriate choice of time step τ and A, the forward iteration can be written as

$$v_{n+1} = 4(1 - \mathrm{Re}^{-1}\tau k^2)v_n(1 - v_n) = rv_n(1 - v_n) \ . \tag{1.41}$$

r can take the values in [0,4] such that $v_n \in [0,1] \Rightarrow v_{n+1} \in [0,1]$ and the limit $r \to 4$ from below corresponds to $\mathrm{Re} \to \infty$. Except for the initial conditions $v_0 = 0,\ v_0 = 1$ the logistic map (1.41) has a single fixed point for all r in [0,4]. Though for different ranges of r the progress of iterates differ significantly. For the values $r \leq 3$ the fixed point is stable, almost all initial conditions converge to this point. These solutions correspond to the laminar flows at small Reynolds numbers. Nonlinearities are subdominant. Above the value 3 there is the first bifurcation. The solution converges to an orbit which alternately visits two values. This regime mimics a periodic wake behind a cylinder obstacle. At $r = 3.5$ the trajectory converges to a period four orbit. Further increasing values exhibit orbits that become very irregular and we can no longer see any oscillations. At $r = 4$ the map is fully chaotic. Slight variations in the initial condition yield different results. All of this is illustrated in figure 1.3. Observe that for $r = 4$ the iterations appear to be completely random yet there are time windows appearing - called islands of stability - where the trajectory is close to the fixed point 0.75 before it eventually moves away. Despite this semblance of randomness the histogram of the velocities taken by the orbit of an arbitrary initial condition are reproducible. Figure 1.4 shows the normalized histogram taken of a 10000 iterates sample. To give insight in the reason of this reproducibility, let us take the transform

$$v_n = \sin^2\left(\frac{\pi}{2}z_n\right) \ , \quad 0 \leq z_n \leq 1 \ . \tag{1.42}$$

The iterates z_n are described by the *tent map*,

$$z_{n+1} = \begin{cases} 2z_n & \text{for } 0 \leq z_n \leq \frac{1}{2} \\ 2 - 2z_n & \text{for } \frac{1}{2} \leq z_n \leq 1 \end{cases} \ , \tag{1.43}$$

such that with (1.43) $v_n = \sin^2\left(\frac{\pi}{2}z_n\right) \Rightarrow v_{n+1} = \sin^2\left(\frac{\pi}{2}z_{n+1}\right)$. Now the z_n are equally distributed on [0,1]. A calculation shows that the v_n are distributed along

$$\frac{\mathrm{d}P(v)}{\mathrm{d}v} = \frac{1}{\pi\sqrt{v(1-v)}} \tag{1.44}$$

such that

$$\int_0^1 \frac{\mathrm{d}P(v)}{\mathrm{d}v}\mathrm{d}v = \int_0^1 \mathrm{d}P = 1 \ .$$

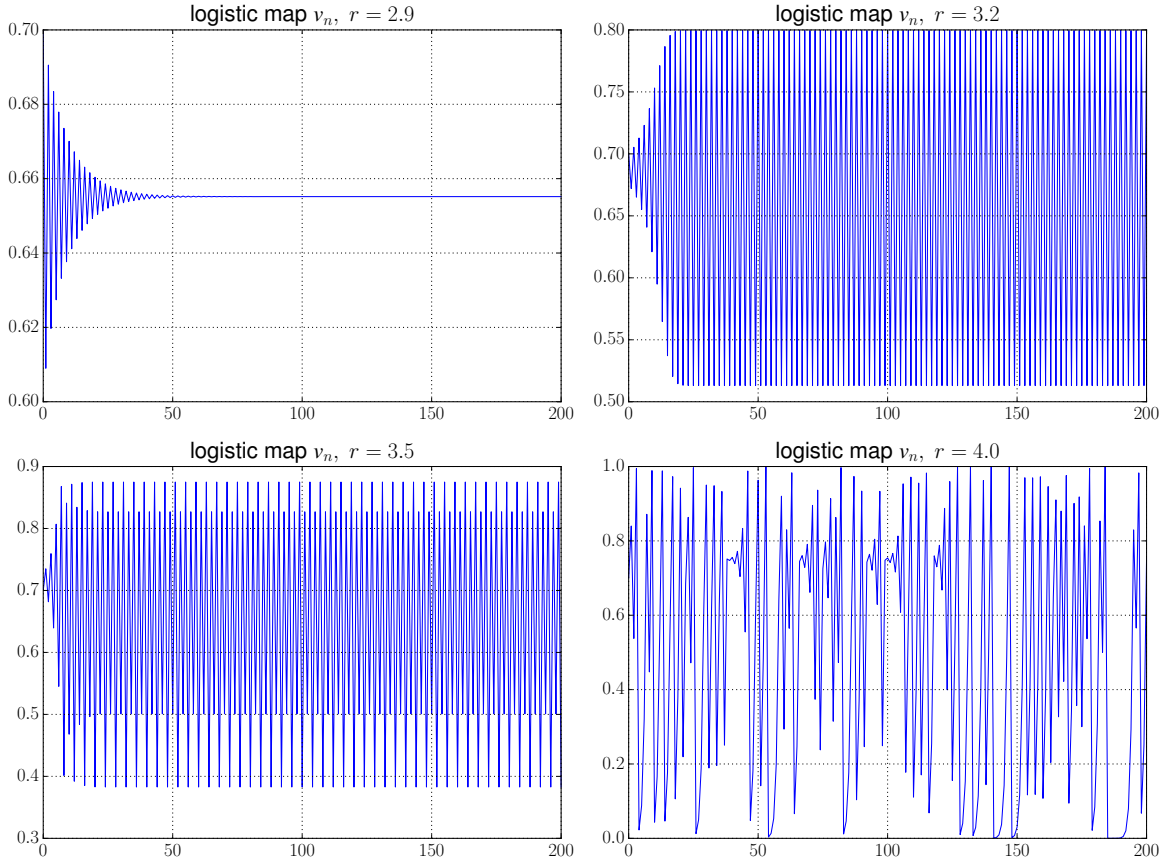


Figure 1.3: The first 200 iterates of the logistic map for different values of r initiated by $v_0 = 0.7$.

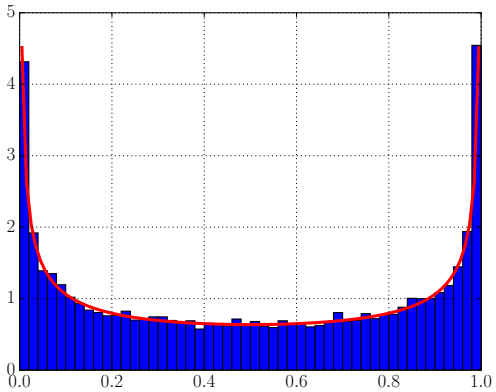


Figure 1.4: Normalized histogram taken of a 10000 iterates sample. The red curve shows the probability density function of the logistic map for $r = 4$.

The quantity $\mathrm{d}P(v)/\mathrm{d}v$ is called the *probability density function* of the logistic map for $r = 4$. It is also displayed in figure 1.4 and obviously the normalized histogram follows this distribution. We close the section and keep this example in mind when turning to the statistical description of turbulence.

1.2.2 Statistical description of turbulence

In the previous section we have presented an example that exhibits a highly fluctuating behavior, but the statistical properties are reproducible. [Fri95] also shows measurement data taken in a wind tunnel as a motivation (c.f. Section 3.1 in his book).

The statistical description here should only give a rough and minimal outline of the concepts. For a more profound treatment refer to [Fri95, Les08]. A *random variable* ϖ is defined by (I) the set of its possible values, sometimes called the phase space and (II) the distribution of probability over this set. The distribution of probability may be described by the *cumulative distribution function* (CDF) or by the *probability density function* (PDF). To give a concrete example related to turbulence, consider the Navier-Stokes system with initial condition as a (vector-valued) random variable ϖ:

$$\begin{aligned} \rho\partial_t v - \mu\Delta v + \rho(v\cdot\nabla)v + \nabla p &= f \\ \nabla\cdot v &= 0 \\ v = v_0 &= \varpi \ . \end{aligned}$$

Here the velocity is a vector-valued *random function* $v(x,t,\varpi)$ that depends on the position, time and the random initial condition. In this case the cumulative distribution function is calculated from the velocity components,

$$\mathrm{Prob}\{v_i(x,t,\varpi) \le y_i\} = P(y_i;x,t) \ , \quad i = 1,\dots,d \ , \tag{1.45}$$

and then the probability density function is given by $\mathrm{d}P(y_i;x,t)/\mathrm{d}y_i$. The PDF is normalized,

$$\int \frac{\mathrm{d}P(y_i;x,t)}{\mathrm{d}y_i}\mathrm{d}y_i = \int \mathrm{d}P = 1 \ . \tag{1.46}$$

Mean values, or generally ensemble averages, are derived from the PDF,

$$\langle v_i(x,t)\rangle = \int v_i(x,t,\varpi)\mathrm{d}P = \int y_i \frac{\mathrm{d}P(y_i;x,t)}{\mathrm{d}y_i}\mathrm{d}y_i \ , \quad i = 1,\dots,d \ , \tag{1.47}$$

and will be denoted by the $\langle\cdot\rangle$-brackets.

Moments of the velocity field - homogeneity, stationarity and isotropy

For the following section we assume (I) velocity fields defined on the whole Euclidean space $\mathbb{R}^d$ that decrease rapid enough towards infinity or (II) velocity fields within a box Ω with periodic boundaries.

Moments of the velocity field are higher-order ensemble averages than the mean value. A moment of order n is a tensor depending on n space-time variables,

$$\langle v_{i_1}(x_1,t_1)v_{i_2}(x_2,t_2)\cdots v_{i_n}(x_n,t_n)\rangle \ . \tag{1.48}$$

Note that here for example x_1 does not denote the first component of x, but the realization of a space variable. The second order moment $\langle v_{i_1}(x_1,t_1)v_{i_2}(x_2,t_2)\rangle$ is called the velocity correlation tensor at points x_1 and x_2 and at times t_1 and t_2. Computing ensemble averages is a daunting task when dealing with laboratory experiments or numerical simulations, as it requires to repeat experiments or numerical runs over and over. Fortunately, under certain assumptions to be described below, ensemble means can be estimated by taking averages over one realization.

- Homogeneity

  Turbulence is called *homogeneous* if all moments of the velocity field built with a set of n space-time points $(x_1,t_1),\ldots,(x_n,t_n)$ are invariant under any translation of the set $(x_1,\ldots,x_n)$. One has in particular

  $$\langle v_{i_1}(x_1,t_1)v_{i_2}(x_2,t_2)\cdots v_{i_n}(x_n,t_n)\rangle = \langle v_{i_1}(x_1+y,t_1)v_{i_2}(x_2+y,t_2)\cdots v_{i_n}(x_n+y,t_n)\rangle. \tag{1.49}$$

  This implies that the velocity correlation tensor only depends on the separation between two points x_1 and x_2. Therefore we write the velocity correlation tensor as a function of the spatial separation, and particularly at same time snapshots as

  $$\mathrm{Corr}(v_i,v_j)(x,t) = \langle v_i(y,t)v_j(x+y,t)\rangle \ . \tag{1.50}$$

  Now the *ergodic theorem* allows to replace the ensemble average by a spatial average. For velocity fields defined on the whole $\mathbb{R}^d$ this results for the correlation tensor in

  $$\mathrm{Corr}(v_i,v_j)(x,t) = \lim_{V\to\infty}\frac{1}{|V|}\int_V v_i(y,t)v_j(x+y,t)\mathrm{d}y \ . \tag{1.51}$$

  If the velocity field is periodic within a box Ω, then (1.51) can be replaced by

  $$\mathrm{Corr}(v_i,v_j)_{ij}(x,t) = \frac{1}{|\Omega|}\int_\Omega v_i(y,t)v_j(x+y,t)\mathrm{d}y \tag{1.52}$$

  where $x+y$ is identified by its equivalent position inside the box Ω.

  The descriptions in the following sections are primarily based on homogeneous turbulence.

- Stationarity

  Turbulence is called *stationary* if all moments of the velocity field involving n space-time points $(x_1,t_1),\ldots,(x_n,t_n)$ are invariant under any translation of $(t_1,\ldots,t_n)$. In particular

  $$\langle v_{i_1}(x_1,t_1)v_{i_2}(x_2,t_2)\cdots v_{i_n}(x_n,t_n)\rangle = \langle v_{i_1}(x_1,t_1+\tau)v_{i_2}(x_2,t_2+\tau)\cdots v_{i_n}(x_n,t_n+\tau)\rangle. \tag{1.53}$$

  This implies that the velocity correlation tensor only depends on the time difference between t_1 and t_2.

- Isotropy

  Turbulence is called statistically *isotropic* if all moments of the velocity field are invariant under simultaneous rotation of the coordinates. The first immediate consequence is that the mean value under mirroring of the coordinate becomes

$$\langle v(x,t)\rangle = 0 \; .$$

1.2.3 Kinetic energy, helicity, enstrophy and scalar spectra

One property of turbulent flows is the broad spectrum in frequency and wave space. In this section we want to briefly recapitulate the necessary mathematical tool and present the quantities of interest in turbulence to analyze. For simplicity we assume, as in the previous section, periodic velocity fields or velocity fields defined on the whole Euclidean space that decrease rapidly enough to infinity.

Spectral tools

The Fourier transform of a velocity field $v(x,t)$ is defined as

$$(\mathcal{F}v)(k,t) = \frac{1}{(2\pi)^d}\int_{\mathbb{R}^d} v(x,t)\exp(-ik\cdot x)\mathrm{d}x \; . \tag{1.54}$$

Note that with this convention the inverse Fourier transform writes

$$v(x,t) = \int_{\mathbb{R}^d} (\mathcal{F}v)(k,t)\exp(ik\cdot x)\mathrm{d}x \; . \tag{1.55}$$

Differentiation in physical space becomes multiplication of the Fourier transform with k in wave space,

$$\begin{aligned} (\mathcal{F}(\nabla v))(k,t) &= i(\mathcal{F}v)(k,t)\otimes k \\ (\mathcal{F}(\nabla\times v))(k,t) &= ik\times(\mathcal{F}v)(k,t) \; . \end{aligned} \tag{1.56}$$

An important equation is the Fourier representation of the Dirac delta distribution:

$$\delta(k) = \frac{1}{(2\pi)^d}\int_{\mathbb{R}^d}\exp(ik\cdot x)\mathrm{d}x \; . \tag{1.57}$$

For possibly complex-valued functions $f(x), g(x) : \mathbb{R}^d \to \mathbb{C}$ that possess the same regularity assumption as for the velocity field above, their cross-correlation is defined as

$$(f\star g)(x) = \int_{\mathbb{R}^d} \overline{f(y)}g(x+y)\mathrm{d}y \; .$$

Now, the *cross-correlation theorem* states that correlation between two functions in physical space transforms to multiplication in wave space, i.e.

$$(\mathcal{F}(f\star g))(k) = (2\pi)^d\,\overline{(\mathcal{F}f)(k)}\,(\mathcal{F}g)(k) \; . \tag{1.58}$$

Kinetic energy spectrum

Starting from the normalized kinetic energy

$$E(t) = \frac{1}{|\Omega|}\frac{1}{2}\int_\Omega v(x,t)\cdot v(x,t)\mathrm{d}x \ , \tag{1.59}$$

the kinetic energy spectrum is defined as a function $E(k,t) : \mathbb{R}_+ \times (0,T] \to \mathbb{R}_+$ such that $E(t) = \int_0^\infty E(k,t)\mathrm{d}k$. $E(k,t)$ corresponds to the kinetic energy density in Fourier space integrated on a sphere. It is always real and positive and specifies the amount of energy contained in the wavenumber interval $[k, k+\mathrm{d}k]$. The kinetic energy can be brought into connection with the velocity correlation tensor,

$$E(t) = \frac{1}{2}\sum_{i=1}^{d}\mathrm{Corr}(v_i, v_i)(0,t) \ .$$

Based on this formula $E(k,t)$ can be numerically computed by taking the Fourier transform of the right-hand side, see Appendix B.1. Turbulent flows are expected to have a long-range kinetic energy spectrum. Theories of turbulence, as we will see in Section 1.2.4, aim at describing the dependency of the spectrum on the wavenumber k, amongst other quantities.

Helicity spectrum

We recall that helicity is a concept only defined in three dimensions. Similarly as for the kinetic energy, with the mean helicity

$$H(t) = \frac{1}{2\,|\Omega|}\int_\Omega v(x,t)\cdot\zeta(x,t)\mathrm{d}x \ ,$$

the helicity spectrum is defined as function $H(k,t) : \mathbb{R}_+ \times (0,T] \to \mathbb{R}$ such that $H(t) = \int_0^\infty H(k,t)\mathrm{d}k$. We can also write the helicity as

$$H(t) = \frac{1}{2}\sum_{j=1}^{d}\mathrm{Corr}(v_j, \zeta_j)(0,t) \ .$$

The helicity spectrum can be numerically computed by taking the Fourier transform of the velocity-vorticity correlation tensor. As for the kinetic energy spectrum refer to appendix B.2 for details. The helicity spectrum is also real-valued, but it cannot be determined by the kinetic energy spectrum $E(k,t)$. In fact the Fourier transform of the velocity correlation tensor in three dimensions can be decomposed into a real and imaginary part, [Les08] Eq. (5.84). The real part consists of an orthogonal projection onto the transverse plane with respect to k scaled by the kinetic energy spectrum. The imaginary part consists of the anti-symmetric operator representing a cross product with the wave vector scaled by the helicity spectrum.

Enstrophy spectrum

The enstrophy spectrum is defined as a function $\mathcal{E}(k,t) : \mathbb{R}_+ \times (0,T] \to \mathbb{R}_+$ such that $\mathcal{E}(t) = \int_0^\infty \mathcal{E}(k,t)\mathrm{d}k$, with the enstrophy

$$\mathcal{E}(t) = \frac{1}{|\Omega|}\frac{1}{2}\int_\Omega \zeta(x,t)\cdot\zeta(x,t)\,\mathrm{d}x\ .$$

Similar to the kinetic energy and helicity the Fourier transform of the vorticity correlation tensor is taken. The transformation of the curl operator in wave space, equation (1.56), implies that the enstrophy spectrum is completely determined by the kinetic energy spectrum, $\mathcal{E}(k,t) = k^2 E(k,t)$. This is not the case for the helicity spectrum.

Scalar spectrum

The scalar spectrum is defined as a function $E_\theta(k,t) : \mathbb{R}_+ \times (0,T] \to \mathbb{R}_+$ such that $E_\theta(t) = \int_0^\infty E_\theta(k,t)\mathrm{d}k$ holds. Since $E_\theta(t)$ is the analog of the kinetic energy its spectrum can be derived in the same way.

1.2.4 Kolmogorov's 1941 theory

With the spectral tools and quantities introduced we are now ready to recapitulate theories of turbulence. Kolmogorov's 1941 theory is one of the milestones in turbulence theory in still humankind's little understanding of turbulence up to today. It describes the behavior of the *kinetic energy cascade* - the redistribution of energy among the various scales of motion through the convective term that we have already mentioned. Kolmogorov assumed before that the kinetic energy spectrum is described by a power law and in his 1941 work he determined the 5/3 exponent. This theory can also be formulated in physical space where it is equivalent to the two-thirds law.

In the following Kolmogorov's 1941 theory will be abbreviated by the acronym K41. We summarize the presentation from [Les08, Fri95] and refer to it for a more profound treatment. K41 is a theory of homogeneous and isotropic turbulence in three space dimensions. It assumes that the kinetic energy spectrum at wavenumbers in the range $k_i < k < k_d$ (k_i and k_d to be explained below) only depends on the dissipation rate ϵ and k. A dimensional analysis yields

$$E(k,t) = C_K \epsilon(t)^{2/3} k^{-5/3} \tag{1.60}$$

where C_K is a universal constant called the *Kolmogorov constant*.

Characteristic scales

Turbulent flows - not only in the setting of Kolmogorov - can be separated into three regimes. We therefore mention those.

Integral range At length scale l that describes the largest scale motions, the geometry of the flow domain and the specific forcing are the essential determinants of the flow field. To the characteristic length scale l, one can associate the injection wavenumber k_i, $l \sim 1/k_i$, which is the smallest wavenumber for the power spectrum to hold.

Inertial range The inertial range is described by the interval $k_i < k < k_d$ where the kinetic energy spectrum follows a power law, $E(k,t) = C_K \epsilon(t)^{2/3} k^{-5/3}$. In this range, energy cascades from the larger scales where it was injected ultimately to the dissipation range.

Dissipation range In the dissipation range, the effects of finite viscosity come into play. At wavenumbers $k > k_d$ viscous dissipation damps the velocity perturbations. k_d is called the dissipation wavenumber or the Kolmogorov wavenumber. It is given by

$$k_d = \left(\frac{\epsilon}{\nu^3}\right)^{\frac{1}{4}} \tag{1.61}$$

and therewith one can associate the Kolmogorov dissipative scale $l_d = 1/k_d$. In this range the kinetic energy spectrum will rapidly (possibly exponentially) drop to negligible values.
A schematic plot of the characteristic spectral energy distribution is shown in figure 1.5.

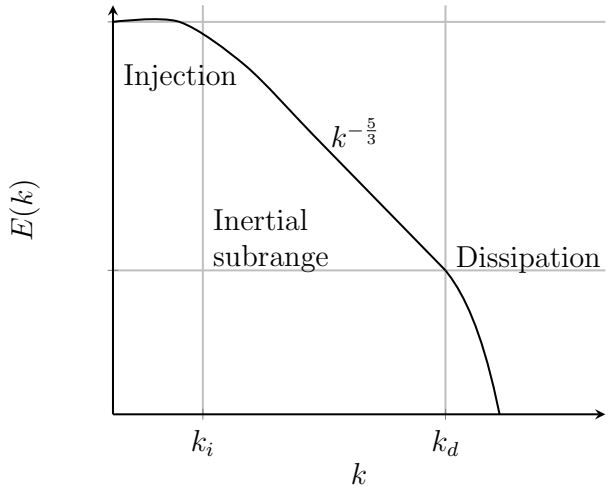


Figure 1.5: Schematic kinetic energy spectrum in the Kolmogorov cascade.

Another relation between the energy dissipation rate, the scale l and the characteristic velocity u can be established. A simple dimensional analysis yields

$$\epsilon \sim \frac{u^3}{l} \, . \tag{1.62}$$

Note that in this relation there is no dependency on the viscosity. In fact this law has a wider applicability than K41, it only uses the finiteness of energy dissipation (in the limit $\nu \to 0$) which has been also verified experimentally. From Equation (1.6) we also have $\epsilon = -\mathrm{d}E/\mathrm{d}t \sim -\mathrm{d}u^2/\mathrm{d}t$ which leads to the kinetic energy decay law over time:

$$\frac{E(t)}{E(t_0)} = \left(\frac{t}{t_0}\right)^{-\alpha} , \quad t \geq t_0 . \tag{1.63}$$

Kolmogorov 1941 in physical space

Kolmogorov's 1941 theory in physical space relies on the velocity increment and gives a description of its statistical moments. The velocity increment is

$$\delta v = v(x+y,t) - v(y,t) \tag{1.64}$$

for spatial lags $x \in \mathbb{R}^d$. The *longitudinal velocity increment* is defined by projection onto the separation direction. For it, let x be a unit vector and r the scalar spatial lag. Without loss of generality we can assume $r \geq 0$. The longitudinal velocity increment writes

$$\delta v_\parallel = (v(y+rx,t) - v(y,t)) \cdot x . \tag{1.65}$$

Statistical moments of the (longitudinal) velocity increment are called structure functions, in particular

$$\langle (\delta v_\parallel)^n \rangle \tag{1.66}$$

denotes the n-th order longitudinal structure function which is a function of the spatial separation distance r. To give a connection to the formulation in wave space, consider the second order velocity structure function $\langle \|\delta v\|_2^2 \rangle$. Expansion therein of the velocity increment and using homogeneity on the second order moments gives

$$\begin{aligned}
\langle \|\delta v\|_2^2 \rangle &= \langle v(x+y,t)\cdot v(x+y,t) - 2\, v(x+y,t)\cdot v(y,t) + v(y,t)\cdot v(y,t) \rangle \\
&= 2\langle v(y,t)\cdot v(y,t)\rangle - 2\langle v(x+y,t)\cdot v(y,t)\rangle \\
&\overset{(1.50)}{=} 2\sum_{i=1}^{d} \mathrm{Corr}(v_i,v_i)(0,t) - 2\sum_{i=1}^{d} \mathrm{Corr}(v_i,v_i)(x,t) .
\end{aligned}$$

Next we use inverse Fourier transform formula (1.55):

$$= 2\sum_{i=1}^{d} \int_{\mathbb{R}^d} (\mathcal{F}(\mathrm{Corr}(v_i,v_i)))(k,t)(1-\exp(ik\cdot x))\mathrm{d}k .$$

The integral is transformed to spherical coordinates. For isotropic velocity fields the Fourier transform of the correlation tensor only depends on separation distance. The angular integrals (in three dimensions) need only be performed with the integrand $(1 - \exp(ik \cdot x))$. The radial remainder contains the kinetic energy spectrum,

$$= 4\int_0^\infty E(k,t)\left(1 - \frac{\sin(kr)}{kr}\right)\mathrm{d}k$$

where k now stands for the modulus of the wave vector and $r = \|x\|_2$ is the separation distance. Substitute the kinetic energy spectrum to follow $E(k,t) \propto k^{-m}$ from $k_i \gg r^{-1}$. Note at first that the integral $\int_0^\infty k^{-m}\,(1-\sin(kr)/(kr))\,\mathrm{d}k$ is finite in the vicinity of zero k if $m < 3$ and finite towards the limit $k \to \infty$ if $1 < m$. Based on the primitive of the substitution $E(k,t) \propto k^{-m}$ from $k_i \gg r^{-1}$ we can express the final outcome of the transformation to wave space:
When the kinetic energy spectrum follows a power law

$$E(k,t) \propto k^{-m} \text{ for } 1 < m < 3\ ,$$

the second order velocity structure function has also a power law of the form

$$\langle \|\delta v\|_2^2 \rangle \propto r^{m-1}\ . \tag{1.67}$$

Isotropy of the velocity field further implies that $\langle (\delta v_\parallel)^2 \rangle$ and $\langle \|\delta v\|_2^2 \rangle$ underlie the same power law.

Kolmogorov's two-thirds law states that the second order longitudinal velocity structure function behaves as

$$\langle (\delta v_\parallel)^2 \rangle \propto \epsilon(t)^{2/3} r^{2/3}\ . \tag{1.68}$$

Based on the derivation above, it is immediate to see that the two-thirds law is equivalent to the statement that the energy spectrum $E(k,t)$ is proportional to $\epsilon(t)^{2/3}k^{-5/3}$. With K41 it can be shown in general that the statistical moments of the longitudinal velocity increment should scale as

$$\langle (\delta v_\parallel)^n \rangle \propto \epsilon(t)^{n/3} r^{n/3}\ . \tag{1.69}$$

For $n = 3$ the four-fifths law is obtained which is an exact result on the constant in the scaling law for the 3rd-order moment:

$$\langle (\delta v_\parallel)^3 \rangle = -4/5\,\epsilon(t) r\ . \tag{1.70}$$

In his 1941 papers Kolmogorov made explicit predictions only for second and third order structure functions. Accurate measurements obtained for the second order structure function didn't detect any discrepancy in the 2/3 exponent. However those experiments (taken in the 1980's-90's) indicate a deviation from the $n/3$-law for high order structure functions. A behavior that is related to the phenomenon of *intermittency*. The results suggest that statistical moments follow power laws

$$\langle (\delta v_\parallel)^n \rangle \propto r^{\xi_n}\ . \tag{1.71}$$

The ξ_n's are called exponents of the structure functions and are specified depending on the intermittency theory. A clearly arranged overview of intermittency models can be found in [KPCS06], Table 1. A non-intermittent model as K41 falls back to $\xi_n = n/3$.

1.2.5 Degrees of freedom of turbulence

In a numerical simulation the number of degrees of freedom per direction is given by the extent of the domain divided by the number of cells in this direction. Similarly, the ratio l/l_d gives the number of degrees of freedom (DOFs) which are needed to describe the motion (from dissipative scales under which the motion is quickly damped by viscosity, to large-scale energy-containing eddies) in each direction.

Three-dimensional turbulence In three dimensions l is proportional to the characteristic velocity cubed as stated in equation (1.62). Together with the formula for l_d one can derive

$$\left(\frac{l}{l_d}\right) \sim (\mathrm{Re})^{\frac{3}{4}} \text{ per direction in 3D}$$
$$\Rightarrow \left(\frac{l}{l_d}\right)^3 \sim (\mathrm{Re})^{\frac{9}{4}} \text{ in 3D.} \tag{1.72}$$

Thus the number of DOFs to perform a DNS in three dimensions scales with $\mathrm{Re}^{9/4}$. Obviously we have

$$\frac{k_d}{k_i} \sim \frac{l}{l_d}$$

and hence $k_d/k_i \sim \mathrm{Re}^{3/4}$.

Two-dimensional turbulence Some more preparations are required to describe the scaling on the Reynolds number as two-dimensional turbulence is inherently different than its three-dimensional counterpart. We have already pointed out the differences between 2D and 3D vorticity transport equation and accordingly the evolution of enstrophy density. From the differential form (1.36), we can pass on to the integral form and use Reynolds transport theorem.

$$\frac{\mathrm{d}\mathcal{E}(t)}{\mathrm{d}t} = -\nu \int_\Omega \nabla\zeta(x,t) : \nabla\zeta(x,t)\mathrm{d}x$$

Recall the integrand on the right-hand side is referred to as palinstrophy density which is solely responsible for the enstrophy rate of change in two dimensions. As for the kinetic energy define the enstrophy dissipation rate as

$$\beta(t) = \int_\Omega \nu\nabla\zeta(x,t) : \nabla\zeta(x,t)\mathrm{d}x \; . \tag{1.73}$$

Finally, we present the *enstrophy cascade*, a major conclusion of the two-dimensional Navier-Stokes equations

$$\frac{\mathrm{d}E(t)}{\mathrm{d}t} = -\epsilon(t) = -2\nu\mathcal{E}(t) \tag{1.74}$$
$$\frac{\mathrm{d}\mathcal{E}(t)}{\mathrm{d}t} = -\beta(t) \; . \tag{1.75}$$

This similar result for the enstrophy will form the basis of two-dimensional turbulence theory.

In a hand-waving manner - which can be justified when explicating the theory - kinetic energy cascade and kinetic energy dissipation rate may be replaced by enstrophy cascade and enstrophy dissipation rate, respectively. The kinetic energy spectrum - by dimensional analysis - is in proportion to

$$E(k,t) \propto \beta(t)^{2/3} k^{-3} \; . \tag{1.76}$$

As before we have the injection wavenumber k_i but in contrast the enstrophy dissipation wavenumber

$$k_d = \left(\frac{\beta}{\nu^3} \right)^{\frac{1}{6}} \; . \tag{1.77}$$

Also a counterpart to (1.62) can be derived: $u \sim \beta^{1/3}/k_i$. Then, the total number of DOFs to perform a DNS in two dimensions,

$$\left(\frac{l}{l_d} \right)^2 = \left(\frac{k_d}{k_i} \right)^2 \sim \frac{u}{\nu k_i} = \mathrm{Re} \; , \tag{1.78}$$

is of order Re.

1.3 Surface renewal

The outline of this section can be summarized as follows: In 1.3.1 we give an introduction to surface renewal process based on [TPUSSS, KPCS06]. In 1.3.2 we develop the tools utilized as detection of ramps in measurements and estimation of their characteristics. In 1.3.3 we present surface renewal models that specify the shape of ramps, especially the termination during the ejection phase. Finally in 1.3.4 we test the detection methods on synthetic time series before applying those on a direct numerical simulation in Chapter 4.

1.3.1 Introduction to surface renewal process

The concept of surface renewal was originally developed in the chemical engineering literature. Higbie (1935) described the transport of gases to liquids. Liquid fresh elemental volumes, with some scalar value characteristic of the outer liquid layers, were hypothesized to come into contact with the surface-gas interface for fixed time intervals. During this contact time (or residence time), molecular diffusion transferred the gas into the liquid which was finally removed from the surface and replaced by fresh fluid from above.

Explanations on turbulence in atmosphere-subsurface coupling are based on the dominance of *turbulent coherent structures*, analogous to such structures reported in the engineering literature. In the following turbulent coherent structures are referred to as *coherent structures* for brevity. In common with engineering flows over both smooth and rough surfaces, coherent structures are characterized by repeated temporal and spatial

patterns of the velocity and scalar field. Ejections and sweeps are features of such coherent motion. In *stable* atmospheric conditions a slow drop of temperature/concentration would be terminated by an ejection where low field values instantaneously rise upward. In *unstable* atmospheric conditions however a slow rise of temperature/concentration would be terminated by an ejection where high field values instantaneously rise upward. In both conditions the ejection is followed by a sweeping motion where the near-surface region is filled by fresh parcels. Near a surface, where the fluid is assumed to reach zero velocity, a shear zone must be created if the fluid is moving in respect to the surface. In the turbulent boundary layer *hairpin vortices* evolve that are the dominant kinematic structure responsible for surface renewal, [HSLR99].

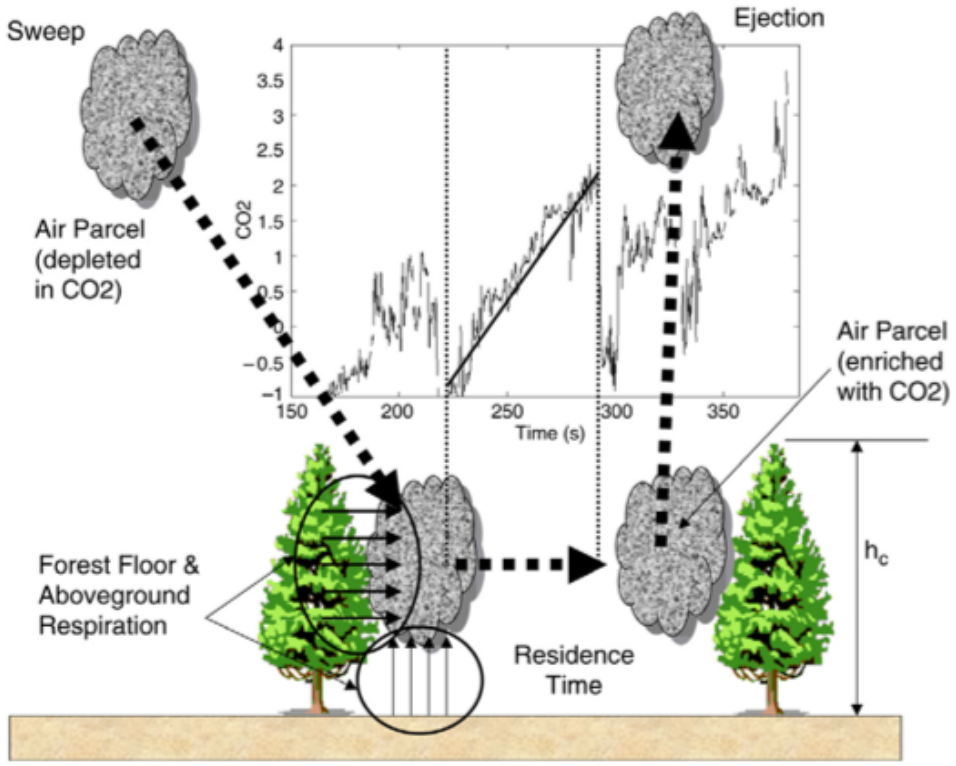


Figure 1.6: Demonstration of the surface renewal effect in [KPCS06]. Shown are the definition of the canopy height h_c in a loblolly pine forest and the mechanisms of sweep, residence and ejection under unstable conditions. Note that the corresponding excerpt of the CO_2 time series is normalized to have zero mean and unit variance.

1.3.2 Detection of coherent structures - Structure functions

Van Atta (VA) suggested in [VA77] the use of temperature structure functions to detect coherent and random parts of a time series. Repeated patterns as ramps in the scalar trace create special shapes in structure functions. For the upcoming development of the

detection method consider for simplicity, concentration only dependent on time, $c(t)$. The steps can be repeated with temperature as an active scalar, $\theta(t)$.

Concentration is decomposed as $c(t) = c_{ch}(t) + c_{rd}(t)$ into a coherent (ch) and random (rd) part. The coherent part models repeated ramp patterns in the surface renewal process. The random part consists of the turbulent fluctuations. Structure functions have already been encountered for spatial lags, c.f. Kolmogorov's two-thirds law in Section 1.2.4, and were successful in describing turbulence. Here the analysis is based on the concentration increment

$$\delta c = c(t) - c(t - \Delta t) \tag{1.79}$$

for a time lag Δt. Obviously the decomposition of the concentration translates to the increment,

$$\delta c = \delta c_{ch} + \delta c_{rd} \ . \tag{1.80}$$

Temporal average over a duration τ is denoted by the brackets $\langle \cdot \rangle_\tau$. This averaging procedure applied to the concentration increment leads to structure functions. In explicit

$$\langle (\delta c)^n \rangle_\tau \tag{1.81}$$

denotes the n-th order structure function that is only a function of the time lag Δt. Analysis of the ramp characteristics is based on higher order structure functions. Therefore with respect to the decomposition of $c(t)$, the following three assumptions are required:

1. The first order structure function of the coherent part vanishes,

$$\langle (\delta c_{ch}) \rangle_\tau = 0 \ .$$

2. The turbulent fluctuating signal is locally isotropic, so that

$$\langle (\delta c_{rd})^n \rangle_\tau = 0 \text{ for } n \text{ odd} \ .$$

3. The two parts are statistically independent,

$$\langle (\delta c_{ch})^n (\delta c_{rd})^m \rangle_\tau = \langle (\delta c_{ch})^n \rangle_\tau \ \langle (\delta c_{rd})^m \rangle_\tau \ .$$

Based on these assumptions higher order structure functions of the time series can be separated into its coherent and random part. In particular the 2nd-, 3rd- and 5th-order structure functions simplify to

$$\langle (\delta c)^2 \rangle_\tau = \langle (\delta c_{ch})^2 \rangle_\tau + \langle (\delta c_{rd})^2 \rangle_\tau \ , \tag{1.82}$$

$$\langle (\delta c)^3 \rangle_\tau = \langle (\delta c_{ch})^3 \rangle_\tau \ , \tag{1.83}$$

$$\langle (\delta c)^5 \rangle_\tau = \langle (\delta c_{ch})^5 \rangle_\tau + 10 \langle (\delta c_{ch})^3 \rangle_\tau \ \langle (\delta c_{rd})^2 \rangle_\tau \ . \tag{1.84}$$

It is important to note that the 3rd-order structure function is completely determined by the coherent part, Equation (1.83). Detection of ramp characteristics is therefore primarily focused on the 3rd-order structure function.

1.3.3 Surface renewal models

Van Atta's model for surface renewal

[VA77] assumed that ramps were regular patterns of fixed geometry that had instantaneous terminations for unstable conditions. In the VA model the coherent part $c_{ch}(t)$ is described as a signal with period τ that consists of a quiescent period $t_s > 0$ followed by a sawtooth ramp of amplitude M.

The temporal dependence of c_{ch} is displayed in figure 1.7. Temporal averaging is

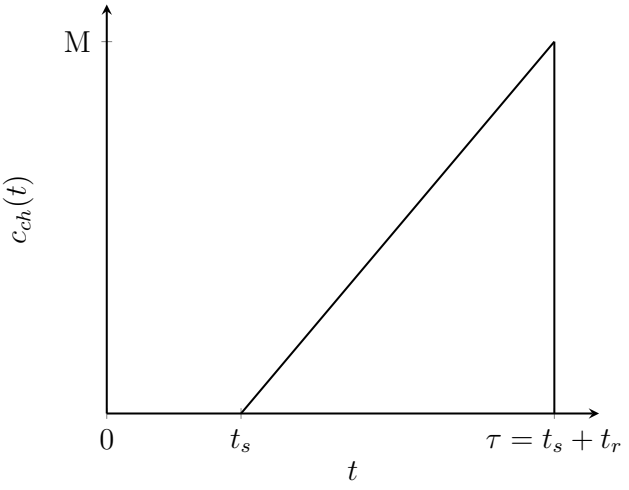


Figure 1.7: Ramp model of VA showing definitions of $c_{ch}(t)$, M, t_s, t_r and τ.

sufficient to take over a single period τ,

$$\langle (\delta c_{ch})^n \rangle_\tau = \frac{1}{\tau} \int_0^\tau (\delta c_{ch})^n \mathrm{d}t \ .$$

The concentration increment is considered for time lags in the range $0 \leq \Delta t < t_s$. This results in

$$\delta c_{ch} = \begin{cases} -\frac{M}{t_r}(t - \Delta t + \tau - t_s), & 0 \leq t \leq \Delta t \\ 0, & \Delta t \leq t \leq t_s \\ \frac{M}{t_r}(t - t_s), & t_s \leq t \leq t_s + \Delta t \\ \frac{M \Delta t}{t_r}, & t_s + \Delta t \leq t \leq \tau \end{cases} , \tag{1.85}$$

and the first-order structure function vanishes, $\langle \delta c_{ch} \rangle_\tau = 0$. Similarly, the integrals can be carried out to derive expressions for the 2nd-, 3rd- and 5th-order structure function:

$$\langle (\delta c_{ch})^2 \rangle_\tau = \frac{M^2 \Delta t}{\tau} \left[1 - \frac{1}{3} \left(\frac{\Delta t}{t_r} \right)^2 \right] \tag{1.86}$$

$$\langle (\delta c_{ch})^3 \rangle_\tau = -\frac{M^3 \Delta t}{\tau} \left[1 - \frac{3}{2} \left(\frac{\Delta t}{t_r} \right) + \frac{1}{2} \left(\frac{\Delta t}{t_r} \right)^3 \right] \tag{1.87}$$

$$\langle(\delta c_{ch})^5\rangle_\tau = -\frac{M^5\Delta t}{\tau}\left[1-\frac{5}{2}\left(\frac{\Delta t}{t_r}\right)+\frac{10}{3}\left(\frac{\Delta t}{t_r}\right)^2-\frac{5}{2}\left(\frac{\Delta t}{t_r}\right)^3+\frac{2}{3}\left(\frac{\Delta t}{t_r}\right)^5\right] \quad (1.88)$$

for $0 \leq \Delta t < t_s$. If $\Delta t \ll t_r$ we obtain the *linearized VA model*:

$$\langle(\delta c_{ch})^n\rangle_\tau = (-1)^n\frac{M^n\Delta t}{\tau} \text{ for } n \geq 2\,. \quad (1.89)$$

Based on the linearized VA model the following observations are recorded:

1. $\langle(\delta c_{ch})^n\rangle_\tau$ depends linearly on Δt for $\Delta t \ll t_r$.
2. The n-th order structure function divided by Δt has a finite value in the small time lag limit,
$$\lim_{\Delta t\to 0}\frac{\langle(\delta c_{ch})^n\rangle_\tau}{\Delta t} = (-1)^n\frac{M^n}{\tau}\,. \quad (1.90)$$

M can be estimated in the VA model by solving real roots of the cubic equation

$$M^3 + Mp + q = 0 \quad (1.91)$$

where

$$p = 10\langle(\delta c)^2\rangle_\tau - \frac{\langle(\delta c)^5\rangle_\tau}{\langle(\delta c)^3\rangle_\tau} \quad \text{and} \quad q = 10\langle(\delta c)^3\rangle_\tau\,. \quad (1.92)$$

The choice of coefficients p, q can be understood by using the decompositions (1.82) - (1.84) therein and then inserting p, q into (1.91). This only leaves structure functions of the coherent part in the cubic equation without any approximation. In the small time lag limit the left-hand side of equation (1.91) vanishes which explains the key idea in this approach. In a second step the ramp duration τ is computed with the help of (1.83) and (1.89),

$$\tau = -\frac{M^3\Delta t}{\langle(\delta c)^3\rangle_\tau}\,. \quad (1.93)$$

A model alike Van Atta with no quiescent period

In this subsection we want to derive an analog of VA's model which exhibits instantaneous termination, but has no quiescent period. The derivation needs to be done separately to the VA model since above $0 \leq \Delta t < t_s$ was assumed but here the quiescent period is missing, $t_s = 0$. As in the VA model temporal averages are calculated as

$$\langle(\delta c_{ch})^n\rangle_\tau = \frac{1}{\tau}\int_0^\tau(\delta c_{ch})^n\mathrm{d}t\,.$$

The concentration increment equals

$$\delta c_{tr} = \begin{cases}\frac{M}{\tau}(\Delta t-\tau), & 0 \leq t \leq \Delta t\\ \frac{M\Delta t}{\tau}, & \Delta t \leq t \leq \tau\end{cases} \quad (1.94)$$

and obviously the first-order structure function vanishes. The 2nd-, 3rd- and 5th-order structure function read

$$\langle(\delta c_{ch})^2\rangle_\tau = \frac{M^2\Delta t}{\tau^2}(\tau-\Delta t) \tag{1.95}$$

$$\langle(\delta c_{ch})^3\rangle_\tau = -\frac{M^3\Delta t}{\tau^3}(\tau-\Delta t)(\tau-2\Delta t) \tag{1.96}$$

$$\langle(\delta c_{ch})^5\rangle_\tau = -\frac{M^5\Delta t}{\tau^5}(\tau-\Delta t)\left[\tau^3-4\tau^2\Delta t+6\tau\Delta t^2-4\Delta t^3\right] . \tag{1.97}$$

$\langle(\delta c_{ch})^2\rangle_\tau$ is a polynomial of degree two in Δt with roots at $\Delta t = 0$ and $\Delta t = \tau$. $\langle(\delta c_{ch})^3\rangle_\tau$ and $\langle(\delta c_{ch})^5\rangle_\tau$ are polynomials of degree three and five in Δt, respectively, with roots at $\Delta t = 0$, $\Delta t = \tau/2$ and $\Delta t = \tau$. The expressions can be linearized in the limit $\Delta t \ll \tau$ which yields

$$\langle(\delta c_{ch})^n\rangle_\tau = (-1)^n\frac{M^n\Delta t}{\tau} \text{ for } n \geq 2 . \tag{1.98}$$

The linearized structure functions exhibit the same behavior as in the linearized VA model. Therefore the observations stated above - (1) $\langle(\delta c_{ch})^n\rangle_\tau$ depends linearly on Δt for $\Delta t \ll \tau$ and (2) $\lim_{\Delta t\to 0}\frac{\langle(\delta c_{ch})^n\rangle_\tau}{\Delta t} = (-1)^n\frac{M^n}{\tau}$ - also apply here. Furthermore the parameters M and τ can be estimated through equations (1.91) - (1.93) as well.

A surface renewal model with finite microfront time

It was observed by [GSPU89] and [UBC+92] the existence of microfronts separating air sweeping into the canopy from within-canopy air. Measurements for example in [VA77] and [CNBL97a] indicate a departure from the linear law in the 3rd-order structure function for small time lags. Whereas the linearized theory is applicable in an intermediate range of Δt.

[CNBL97a] considered ramps of total duration τ that have non-instantaneous terminations with no spacing in between. In this model an additional parameter t_f is required to specify the width of the finite microfront. [CNBL97a] did a rigorous analysis on this signal that is summarized here. In their article structure functions were developed to estimate the ramp dimensions. They also noted that the results do not change significantly with a quiescent period included, but did not present equations in their subsequent publication [CNBL97b] either.

The temporal dependence of c_{ch} is displayed in figure 1.8. As usual temporal averages are calculated as

$$\langle(\delta c_{ch})^n\rangle_\tau = \frac{1}{\tau}\int_0^\tau(\delta c_{ch})^n\mathrm{d}t .$$

The concentration increment can be separated into three regimes with respect to Δt, c.f. figure 1.9. By carrying out the integrations one arrives at the 1st-order structure function,

$$\langle\delta c_{ch}\rangle_\tau(\Delta t) = \begin{cases}\frac{1}{2}\frac{M\tau}{\tau-t_f}\left(1-\frac{1}{t_f}\right)\Delta t^2, & 0\leq\Delta t\leq t_f\\ 0, & t_f\leq\Delta t\leq\tau-t_f\end{cases} . \tag{1.99}$$

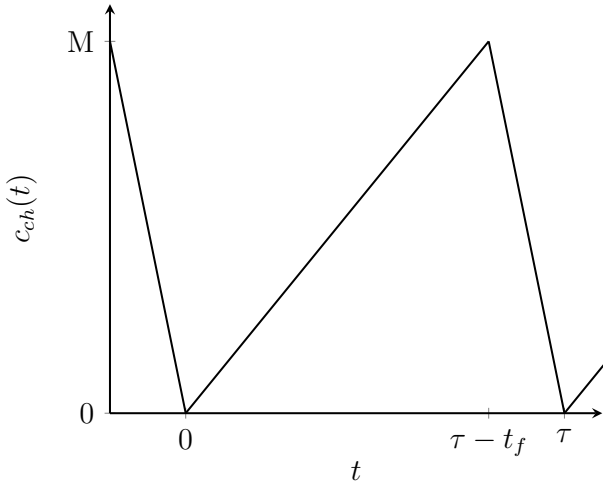


Figure 1.8: Finite microfront model showing definitions of $c_{ch}(t)$, M, t_f and τ.

The same way formulas for the 2nd-, 3rd- and 5th-order structure function divided by Δt are derived:

$$\frac{\langle(\delta c_{ch})^n\rangle_\tau}{\Delta t} = \begin{cases} (-1)^n \frac{M^n}{\tau}\left[\left(\frac{\Delta t}{t_f}\right)^{n-1}\left(1+\frac{t_f-\Delta t}{\tau-t_f}\right) - \frac{(n-1)}{(n+1)}\left(\frac{\Delta t}{t_f}\right)^2\left(1-\frac{t_f-\Delta t}{\tau-t_f}\right)\right], & 0 \le \Delta t \le t_f \\ (-1)^n \frac{M^n}{\tau}\left[\frac{(\tau-\Delta t)^n+(-1)^n(\tau-\Delta t)\Delta t^{n-1}}{(\tau-t_f)^n} - \frac{(n-1)t_f\tau^n f_n}{(n+1)\Delta t(\tau-t_f)^n}\right], & t_f < \Delta t \le \tau - t_f \\ \frac{M^n}{\tau}\left[\frac{(\tau-\Delta t)^n}{t_f^{n-1}\Delta t} - \frac{(n-1)(\tau-\Delta t)^{n+1}}{(n+1)t_f^n\Delta t}\right], & \tau - t_f < \Delta t \le \tau \end{cases} \tag{1.100}$$

with

$$f_n = \begin{cases} 1, & \text{for } n = 2 \\ 1 - 2\left(\frac{\Delta t}{\tau}\right), & \text{for } n = 3 \\ 1 - \frac{9}{2}\left(\frac{\Delta t}{\tau}\right) + \frac{15}{2}\left(\frac{\Delta t}{\tau}\right)^2 - 5\left(\frac{\Delta t}{\tau}\right)^3, & \text{for } n = 5 \end{cases}.$$

If $\Delta t \ll t_f \ll \tau$ we can truncate powers of $\Delta t/t_f$ to obtain the small time lag behavior

$$\frac{\langle(\delta c_{ch})^2\rangle_\tau}{\Delta t} = \frac{M^2}{\tau}\left(\frac{\Delta t}{t_f}\right), \tag{1.101}$$

$$\frac{\langle(\delta c_{ch})^3\rangle_\tau}{\Delta t} = -\frac{M^3}{2\tau}\left(\frac{\Delta t}{t_f}\right)^2, \tag{1.102}$$

$$\frac{\langle(\delta c_{ch})^5\rangle_\tau}{\Delta t} = \frac{2M^5}{3\tau}\left(\frac{\Delta t}{t_f}\right)^2. \tag{1.103}$$

We can now make the following observations (recall the observations with no finite microfront):

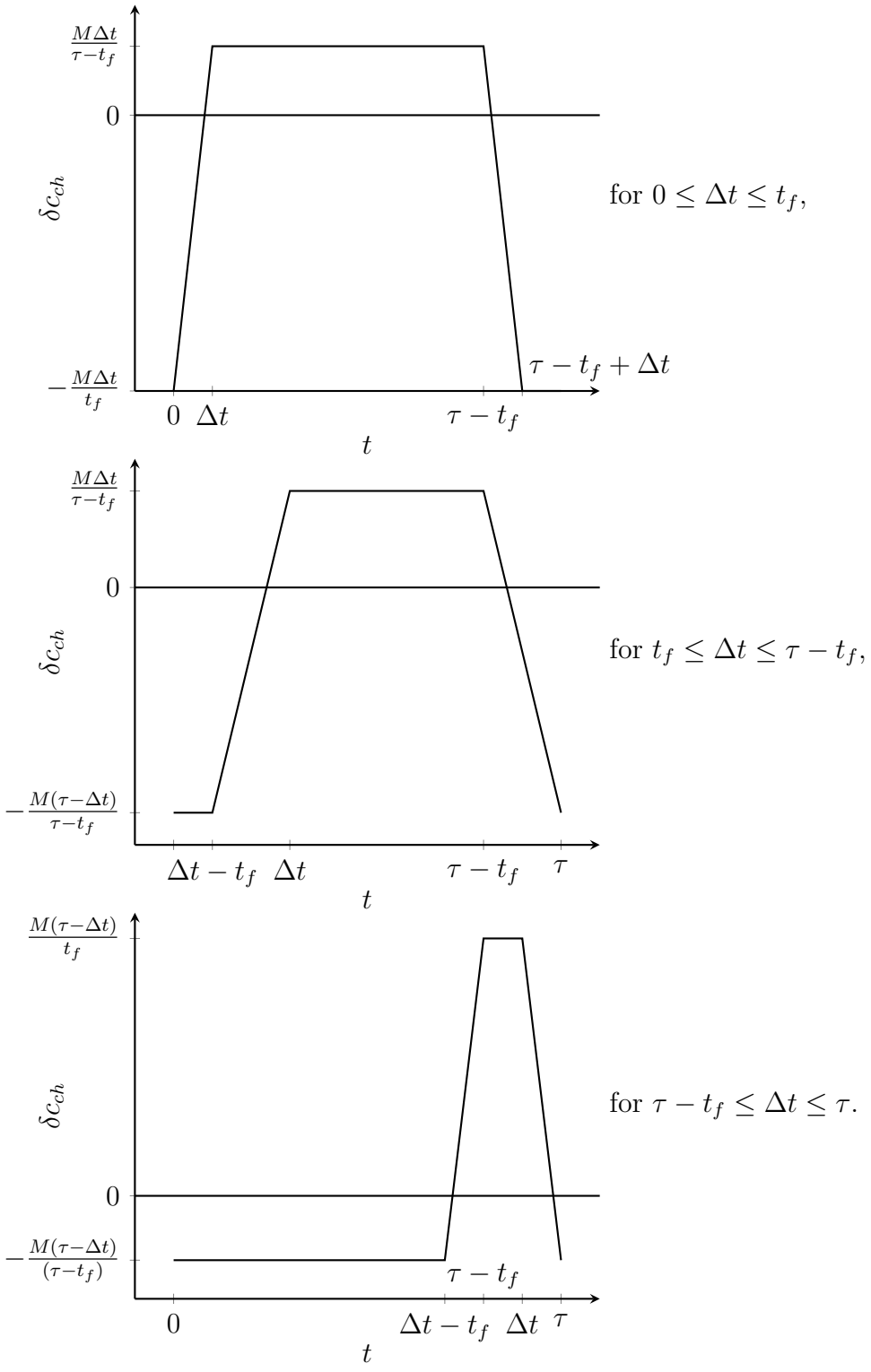


Figure 1.9: Concentration increment of the finite microfront model in the three regimes with respect to Δt shown on the right.

1. $\langle(\delta c_{ch})^n\rangle_\tau$ does not depend linearly on Δt as $\Delta t \to 0$.

2. The n-th order structure function divided by Δt approaches zero as Δt approaches zero,
$$\lim_{\Delta t \to 0} \frac{\langle(\delta c_{ch})^n\rangle_\tau}{\Delta t} = 0 \,. \tag{1.104}$$

In the following we want to restrict ourselves to the 3rd-order structure function as the corresponding superpositioned time series is completely determined by the coherent part, Equation (1.83). Therefore let M, τ, t_f be fixed and consider $\langle(\delta c_{ch})^3\rangle_\tau/\Delta t$ as a function of Δt.
Obviously $\langle(\delta c_{ch})^3\rangle_\tau/\Delta t$ has a root $\Delta t = 0$. It is negative in the first range $0 \leq \Delta t \leq t_f$ and takes the value $-M^3/(2\tau)$ at $\Delta t = t_f$ which coincides to the value given by the lowest order approximation (1.102). In the intermediate range $t_f < \Delta t \leq \tau - t_f$ a root is located at $\Delta t = \tau/2$ and afterwards the function is positive. In the last range $\tau - t_f < \Delta t \leq \tau$ there is a root of $\langle(\delta c_{ch})^3\rangle_\tau/\Delta t$ at $\Delta t = \tau$. An inflection point can be found in the first range if $\tau \leq 7t_f$. Moreover a minimum if $\tau \leq 2.5t_f$. Hence for practical ratios of t_f/τ the minimum is taken in the intermediate range besides to the root at $\Delta t = \tau/2$. The position of the extremal value can be further specified such that $\Delta t \gtrapprox 1.5t_f$ for $t_f \leq 0.1\tau$.

After clarification of the curve shape the ramp dimensions can be estimated. We pick an appropriate part of data to perform a least squares fit to minimize the functional in the intermediate range, c.f. Equation (1.100). [CNBL97a] also noted the linearized VA model is applicable away from the different behavior of the finite microfront model. Thereby ramp amplitude and total ramp duration can be estimated for time lags $\Delta t \geq \Delta t_m$, where Δt_m corresponds to the separation such that $-\langle(\delta c_{ch})^3\rangle_\tau/\Delta t$ takes its maximum value. Especially those estimates can be used as an initial guess in the least squares fit.

1.3.4 Analysis of synthetic data

We perform an *ideal surface renewal* analysis that only requires the temporal trace of a scalar as input. A simple model example from [KPCS06] is taken. The time series is generated from 1024 samples over the interval $[0, 2]$. For the coherent part we either take a sawtooth shape without microfront or with microfront width of $t_f = 0.1\tau$. Both signals have a total duration of $\tau = 0.1$ and an amplitude of $M \approx 3.463$. For the random part we take a fractional Brownian motion (fBm) time series with Hurst exponent $H = 1/3$ to model K41 turbulence. The Hurst parameter is obtained by evaluating the exponent of the structure functions in turbulence (1.71) at $n = 1$. All signals are normalized to have zero mean and unit variance. In total the two purely coherent signals are analyzed plus a convex combination of each with the fBm-series. The parameter in the convex combination is denoted by α,

$$c = \alpha c_{rd} + (1 - \alpha) c_{ch}$$

and set to $\alpha = 0.6$. A display of all signals can be found in figure 1.10. In both superpositions on the right ramp patterns are observable, mainly such repeating patterns in the time window from $t \sim 1.1$ to $t \sim 1.6$. Moreover note in between, existing time windows around $t \sim 1.0$ or at the beginning/end of the sampling interval where the iterates seem to be completely random. Such time windows of randomness and repeating ramp shapes, respectively, will be rediscovered in measurements taken within a direct numerical simulation.

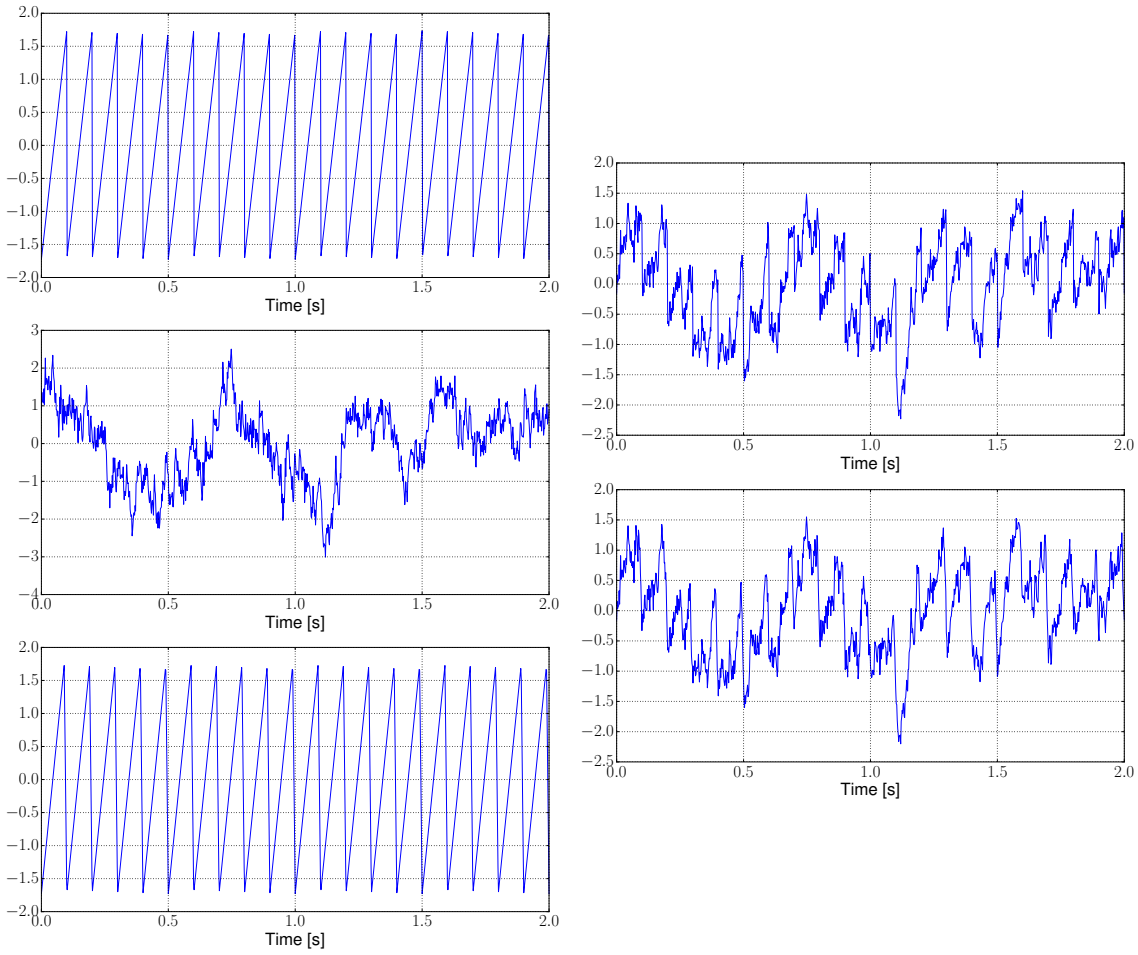


Figure 1.10: Collection of time series used for structure function analysis.
Left column shows the sawtooth shape (top), fractional Brownian motion (middle) and sawtooth shape with finite microfront (bottom).
Right column shows the corresponding superposition given by a convex combination with parameter $\alpha = 0.6$.

Approximations to the n-th order structure function are the discrete truncated version

$$\widehat{\langle(\delta c)^n\rangle}_{tr}(j\Delta t) = \frac{1}{m-j}\sum_{i=j}^{m-1}\big(c(i\Delta t) - c((i-j)\Delta t)\big)^n \tag{1.105}$$

and the discrete periodic version

$$\widehat{\langle(\delta c)^n\rangle}_{pe}(j\Delta t) = \frac{1}{m}\sum_{i=0}^{m-1}\big(c(i\Delta t) - c((i-j)\mathrm{mod}\, m\,\Delta t)\big)^n\ . \tag{1.106}$$

In the two formulas m denotes the number of sampling points, j the integer lag and Δt the inverse sampling frequency. The averaging window in (1.106) is $m\Delta t$ for all lags since the discrete periodic version assumes perfectly periodic signals and thereby $i-j$ is identified within the set $\{0,\ldots,m-1\}$. In contrast the truncated version (1.105) exhibits truncation errors for lags approaching the sampling length due to less points collected. We prefer nevertheless the truncated version because it is applicable for non-periodic signals, too.

The computed 2nd-, 3rd- and 5th-order structure function of the two coherent signals are displayed in figure 1.11. Note the repeated patterns created: The dip in the second order structure function that resembles an ejection event. The change of sign in the third and fifth order structure function every half-integral multiple of the ramp duration. Further in the proximity of ejection events, progression of third and fifth order structure function squeezed to a double bend clearly visible in the finite microfront model. In figure 1.12 the behavior of the 3rd-order structure function in the small lag limit is demonstrated. The top row shows the negative function in a log-log plot together with a linear law. It can be clearly seen that the instantaneous termination model depends linearly whereas the finite microfront model reveals a steeper descent in log-log axes. The bottom row shows the negative 3rd-order structure function divided by the integer lag. Again the finite limit in the instantaneous termination model predicted by equation (1.90) is evident. In contrast the limit tends to zero for the finite microfront model as predicted by equation (1.104).

Next the ramp parameters for the two coherent signals are estimated from the truncated third order structure function. In the linearized VA model we always choose the results of (1.91)- (1.93) for which $-\widehat{\langle(\delta c)^3\rangle}_{tr}(j\Delta t)/j$ takes its maximum value.

Instantaneous termination ramp In the model alike Van Atta with no quiescent period it is only possible to apply the linearized VA model. The maximum value criterion yields the following estimates of the ramp amplitude M and ramp duration τ:

$$(\hat{M}_{VA}, \hat{\tau}_{VA}) = (3.3876, 0.09955)\ .$$

Note that both values are slightly lower than the correct values $(M,\tau) = (3.463, 0.1)$. This can be explained since the zero lag limit is not available and the remainder in the analytic version (1.96) is a parabola $(\tau-\Delta t)(\tau-2\Delta t)$ that we can also observe in the bottom left of figure 1.12. The same analysis based on the discrete periodic structure function does not have an impact on the digits presented above.

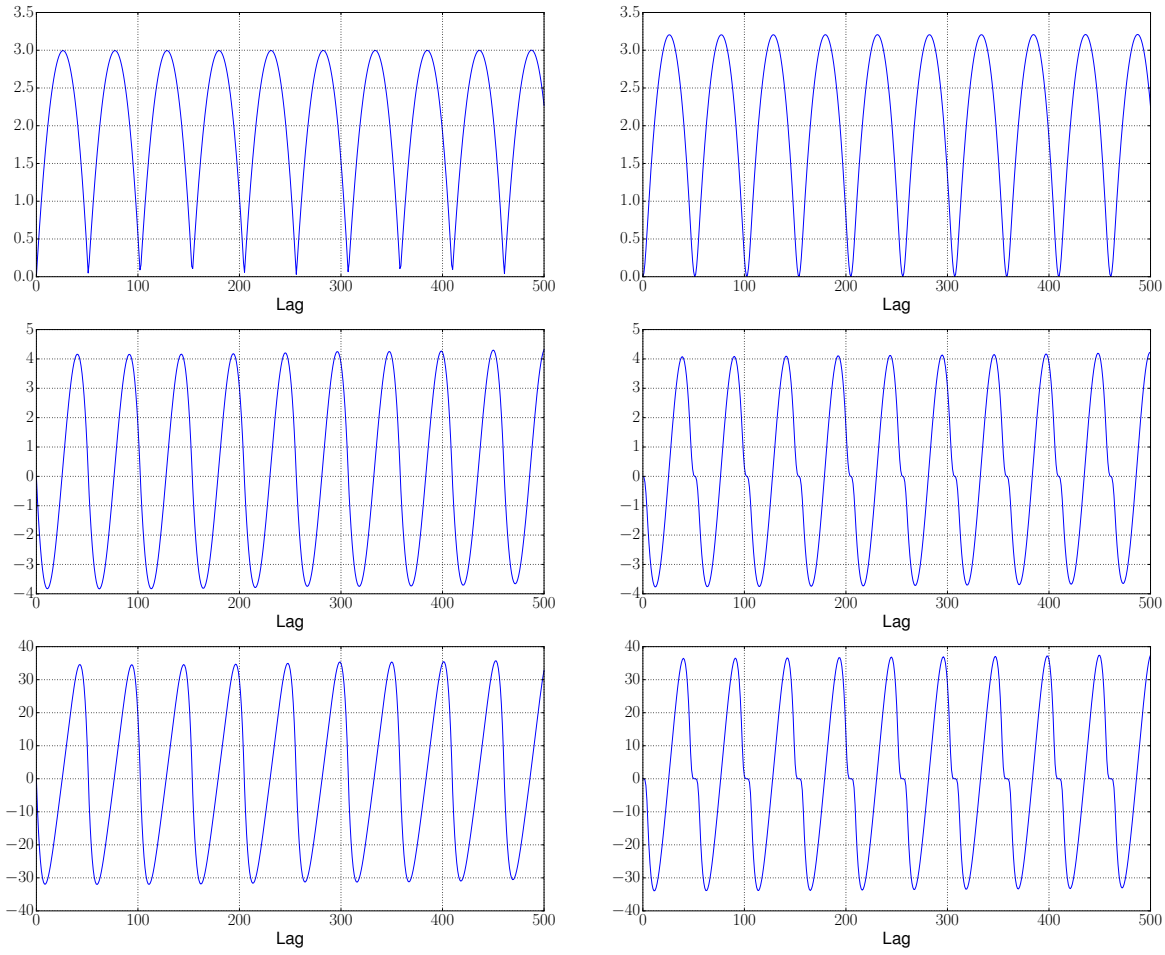


Figure 1.11: Plot of discrete structure functions. Left column belongs to sawtooth shape and right column to sawtooth shape with finite microfront. Top row shows second order structure functions. Middle row shows third order structure functions. Bottom row shows fifth order structure functions.

Finite microfront ramp In the model suggested by [CNBL97a] both the linearized VA model and the corresponding finite microfront model can be applied. The maximum value of $-\widehat{\langle(\delta c)^3\rangle}_{tr}(j\Delta t)/j$ is located at $j = 7$. The linearized VA model for this lag yields the following estimates of M and τ only:

$$(\hat{M}_{VA}, \hat{\tau}_{VA}) = (2.6939, 0.09765) \; .$$

Especially the ramp amplitude is much lower than the correct value as the largest value picked for the evaluation at lag $j = 7$ is further away from lag zero than the instantaneous

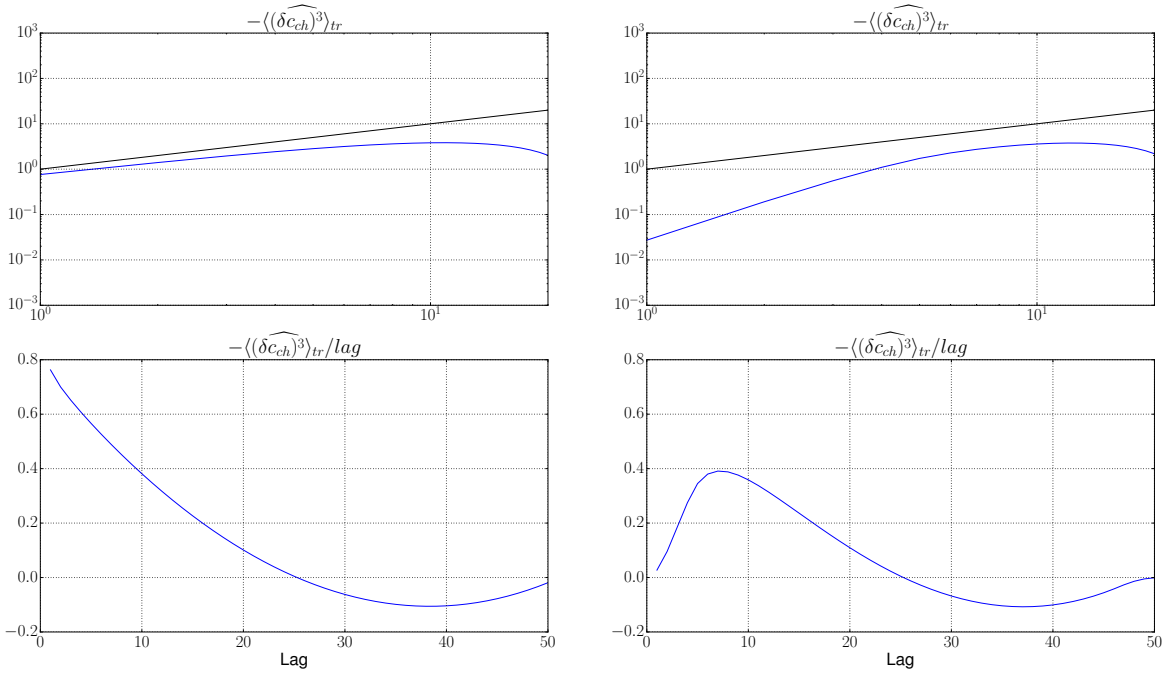


Figure 1.12: Discrete negative third order structure function behavior in the small lag limit. Left column belongs to sawtooth shape and right column to sawtooth shape with finite microfront. Top row shows a log-log plot together with linear law. Bottom row shows negative structure function divided by integer lag.

termination ramp. For successively larger lags both results agree well until deviations for the lag approaching the ramp duration, c.f. bottom row of figure 1.12.

In the finite microfront model the least squares functional in the intermediate range is minimized, Equation (1.100) for $t_f < \Delta t \leq \tau - t_f$. Therefore recall the discussion on the curve shape given by Equation (1.100) when M, τ, t_f are parameters. Knowing the maximum position in terms of the integer lag and the corresponding result from the linearized VA model, the least squares fit can be carried out. The obtained ramp amplitude, ramp duration and microfront width are

$$\begin{aligned} \hat{M}_{mf} &= 3.4166 \pm 8.7 \cdot 10^{-4} \\ \hat{\tau}_{mf} &= 0.09906 \pm 6.0 \cdot 10^{-5} \\ \hat{t}_{f,\,mf} &= 0.00982 \pm 2.3 \cdot 10^{-5} \,. \end{aligned}$$

All three values are slightly lower than the correct $(M, \tau, t_f) = (3.46388, 0.1, 0.01)$ which can be explained by errors from the truncated version of the structure function. The same analysis based on the discrete periodic structure function gives

$$\hat{M}_{mf} = 3.464008 \pm 4.6 \cdot 10^{-6}$$

$$\hat{\tau}_{mf} = 0.1001034 \pm 3.2 \cdot 10^{-7}$$
$$\hat{t}_{f,\,mf} = 0.0100103 \pm 1.2 \cdot 10^{-7}$$

which delivers a better estimate of the ramp dimensions. The fit of the intermediate range to the data is found figure 1.14, left part. The graph parametrized by the correct values (M, τ, t_f) has been omitted as it would have led to a indistinguishable curve with respect to the fit and the data points.

Finally the third order structure function of superposed signals $c = \alpha c_{rd} + (1-\alpha)c_{ch}$ are studied. The left column of figure 1.13 shows the result of the instantaneously terminated ramp as coherent part, the right column the result of the finite microfront ramp as coherent part. Note that the range up to intermediate lags is well reproduced

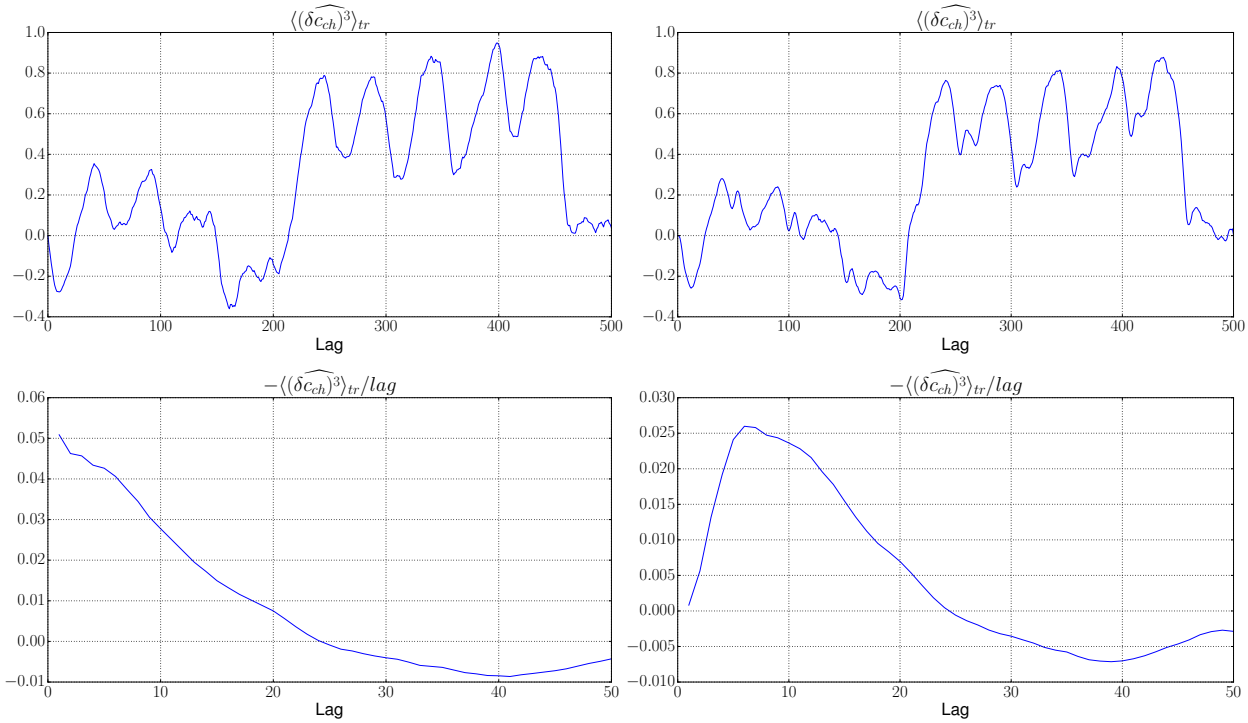


Figure 1.13: Computed third order structure function of the superposed signal $c = \alpha c_{rd} + (1-\alpha)c_{ch}$, $\alpha = 0.6$. Left column shows the curves with instantaneously terminated sawtooth as a coherent signal, right column the curves with finite microfront sawtooth as a coherent signal. Top row corresponds to third order structure function in terms of integer lag. Bottom row negative function divided by integer lag.

and resembles the respective coherent part, but contains wiggles and is not smooth. The signal is still dominated by noise which is weighted by the factor $\alpha = 0.6$ whereas the coherent part is weighted by the factor $1 - \alpha = 0.4$. For lags larger than the ramp duration, the curve is affected by statistical errors due to only 1024 iterates generated.

Nevertheless repeated patterns in the structure function related to ejection events are visible. As mentioned in previous discussions we are mainly interested in the small till intermediate lag behavior.

Instantaneous termination ramp + fBm(1/3) As in the corresponding coherent sample, only the linearized VA model is employed. The estimates of the ramp amplitude M and ramp duration τ are

$$(\hat{M}_{VA}, \hat{\tau}_{VA}) = (1.3978, 0.10486)$$

which approximates well the correct value $(M, \tau) = (1.3852, 0.1)$. Note that the correct value of the amplitude in the superponed signals are rescaled by $1 - \alpha = 0.4$ whereas the durations remain unchanged.

Finite microfront ramp + fBm(1/3) As above both the linearized VA model and the finite microfront model are applied. The maximum value of $-\widehat{\langle(\delta c)^3\rangle}_{tr}(j\Delta t)/j$ is located at $j = 6$. The linearized VA model for this lag gives the following $\hat{M}$ and $\hat{\tau}$ only:

$$(\hat{M}_{VA}, \hat{\tau}_{VA}) = (1.0891, 0.09712)\ .$$

For the reasons already mentioned in the coherent part, the ramp amplitude is much lower than the correct value, in contrast to the ramp duration. The finite microfront model returns estimated ramp amplitude, ramp duration and microfront width equal to

$$\begin{aligned} \hat{M}_{mf} &= 1.3773 \pm 0.0042 \\ \hat{\tau}_{mf} &= 0.09693 \pm 0.00072 \\ \hat{t}_{f,\,mf} &= 0.00944 \pm 0.00019 \end{aligned}$$

that are slightly lower than the correct $(M, \tau, t_f) = (1.385552, 0.1, 0.01)$, but approximate the triple well. A comparison with the discrete periodic structure function gives

$$\begin{aligned} \hat{M}_{mf} &= 1.3819 \pm 0.0039 \\ \hat{\tau}_{mf} &= 0.09624 \pm 0.00065 \\ \hat{t}_{f,\,mf} &= 0.00951 \pm 0.00017 \end{aligned}$$

which does not reveal a significant difference with respect to the results from the truncated version. Though only the prescribed value of M lies within the error bound of $\hat{M}_{mf}$. The error estimates of $\hat{M}_{mf}, \hat{\tau}_{mf}, \hat{t}_{f\,mf}$ for the discrete and periodic structure function have an overlap. This is different to the purely coherent signal where moreover the standard deviation of each parameter in the periodic identification is two orders of magnitude lower. The fit of $-\widehat{\langle(\delta c)^3\rangle}_{tr}(j\Delta t)/j$ in the intermediate range to the data is depicted in figure 1.14, right part. The graph parametrized by the correct values (M, τ, t_f) is added in dashed line style.

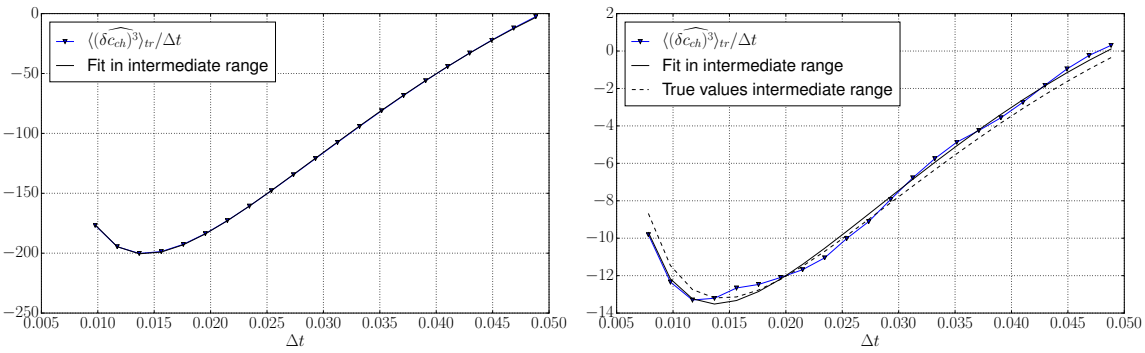


Figure 1.14: Third order structure function plotted over the time lag. Left picture shows the result coming from the finite microfront ramp, right picture superposition with the fBm(1/3) time series. Also included is a least squares fit to the intermediate range in the finite microfront model.

1.4 Vortex identification in Computational Fluid Dynamics

We have pointed out that turbulent flows can be separated into three regimes. In the inertial range the only relevant process is vortex dynamics. This is already described by the *Richardson cascade* that has been discovered before K41 and offers basis assumptions in Kolmogorov's phenomenology. In fact the cascade picture brings *vortices encompassing smaller vortices with still smaller vortices and so on all the way down to the Kolmogorov dissipative scale.* For the upcoming summary on vortex identification methods we refer to [JH95]. Difficulties in the identification arise for flow fields in three dimensions which is the starting point here.

1.4.1 Intuitive measures on v and p

Local pressure minima are commonly used to extract vortex regions. The concept of a local pressure minimum in three dimensions requires clarification because pressure may have minimum in all three directions in a point, or it may have a minimum only in a plane perpendicular to the vortex axis. Here the latter condition (i.e. minimum in a plane) is considered. The definition of a pressure minimum in a vortex region may be misleading as for example it can occur due to the unsteady straining even though the flow has no vortex or it can be eliminated due to viscous effects in a flow with vortical motion. Also [JH95] mentions the example where pressure is inherently of a larger scale than vorticity in vortex regions which highlights the point of carefully choosing the threshold for the minimum. It is furthermore stated that an isopressure surface in mixing layer turbulence fails to capture the ribs and rolls simultaneously. Within a rib pressure is much higher than at the roll center and therefore enlarging the value does not give a clear identification of the rolls.

Pathlines and streamlines are identifications based on the velocity field that have been

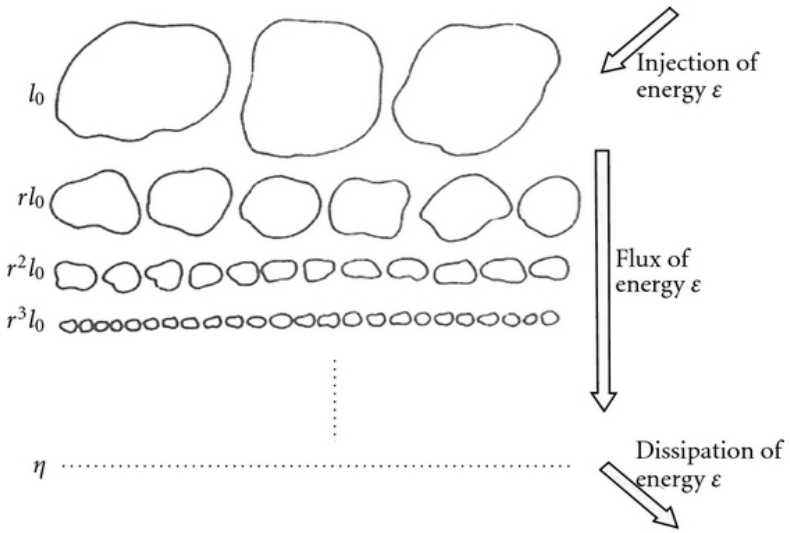


Figure 1.15: Illustration on Richardson cascade, [Fri95]. In this picture l_0 denotes the integral scale l, $0 < r < 1$ is a parameter and η denotes the Kolmogorov dissipative scale l_d.

proposed to use as vortex regions extraction. However closed or spiral pathlines may not capture a vortex accurately as it can undergo pairing, stretching or breakdown. Closed or spiral streamlines used as identification are also problematic because they are not Galilean invariant.

1.4.2 Vorticity and helicity

Vorticity magnitude $\|\zeta\|_2$ or components of the vorticity have been widely used to represent vortex regions and cores. This approach is fairly successful in free shear flows, however in shear flows vorticity does not only identify vortex cores. For example in a turbulent boundary layer, extremal values for the vorticity magnitude as well as for a vorticity component occur at the wall. However, the flow near the wall is characterized by shear and exhibits by no means vortical motion.

Helicity can be used to identify vortices. At first it was discovered by [Mor61, Mof69] that for inviscid flows $H(t)$ is a conserved quantity. The potential for vortex extraction has been recognized in [DLS90]. Therein besides the mean helicity $H(t)$,

- the helicity density $H_d = v \cdot (\nabla \times v)$
- and the normalized helicity $H_n = \frac{v \cdot (\nabla \times v)}{\|v\|_2 \|\nabla \times v\|_2}$

are taken as criteria. Vortex regions can be detected by considering absolute values of the helicity density above a threshold. The aim of the normalized helicity is to extract vortex core lines. The underlying assumption is, that near vortex core regions the angle

between v and $\nabla \times v$ is small. The limiting case is $v \parallel \nabla \times v \Rightarrow H_n = \pm 1$ inside the core regions.

1.4.3 Q-criterion

Let us consider Galilean invariant definitions of a vortex based on the velocity Jacobian ∇v. Several criteria can be constructed from the characteristic polynomial of ∇v. By Q the second invariant in the characteristic equation of the velocity Jacobian is denoted and accordingly the Q-criterion is based upon. It is useful to split the velocity Jacobian into its symmetric and anti-symmetric part, c.f. Equations (1.19) and (1.20),

- strain tensor $S = \frac{1}{2}(\nabla v + \nabla v^T)$,
- vorticity tensor $\mathrm{Z} = \nabla v - S = \frac{1}{2}(\nabla v - \nabla v^T)$.

Then Q can be expressed as

$$Q = \frac{1}{2}(\mathrm{Z} : \mathrm{Z} - S : S) \; . \tag{1.107}$$

Q represents thus the local balance between vorticity magnitude and shear strain rate. It can be easily shown that Q vanishes at a wall, unlike $\|\zeta\|_2$. Hence the Q-criterion defines vortex regions as

$$\text{connected regions of } Q > 0 \tag{1.108}$$

meaning that over there vorticity magnitude is stronger than shear strain rate. This definition is free from the problem associated with $\|\zeta\|_2$ which fails to properly recognize vortical motion near a wall.

Q can also be interpreted as the source term in the Poisson equation for the pressure. From the right-hand side $\rho \nabla v^T : \nabla v$ in (1.32), insert the decomposition into S and Z to obtain

$$\Delta p = 2\rho Q \; .$$

From the maximum principle, [Eva10], we can conclude that if $Q > 0$ the pressure maximum only occurs at the boundary and if $Q < 0$ the pressure minimum only occurs at the boundary. However, $Q > 0$ does not necessarily imply that the pressure minimum occurs within the region, it can also take its minimum on the boundary of $Q > 0$. Thus there is no explicit connection between a region with $Q > 0$ and region of a pressure minimum.

1.4.4 λ_2-criterion

We have summarized that a pressure minimum cannot be used as a general detection criterion for a vortex region because it can be created by unsteady straining without involving a vortex or it can be eliminated by viscous effects in a flow with vortical motion. Nevertheless it provides a starting point for a new definition by simply discarding those effects. Information on local extremal pressure values are contained in the Hessian $\nabla^2 p$.

The acceleration is denoted here in shorthand by $a = \frac{\mathrm{d}v}{\mathrm{d}t}$. The derivation of the λ_2-criterion starts by taking the gradient of the momentum equation,

$$\rho \nabla a = \mu \nabla \Delta v - \nabla^2 p \, .$$

The acceleration Jacobian on the left-hand side can be decomposed into symmetric and anti-symmetric parts as follows

$$\rho \left[\frac{\mathrm{d}S}{\mathrm{d}t} + S^2 + \mathrm{Z}^2 \right] + \rho \left[\frac{\mathrm{d}\mathrm{Z}}{\mathrm{d}t} + \mathrm{Z}S + S\mathrm{Z} \right] = \mu \nabla \Delta v - \nabla^2 p \, .$$

The third-order derivatives on the right-hand side can be written in terms of the strain tensor and vorticity tensor:

$$\rho \left[\frac{\mathrm{d}S}{\mathrm{d}t} + S^2 + \mathrm{Z}^2 \right] + \underbrace{\rho \left[\frac{\mathrm{d}\mathrm{Z}}{\mathrm{d}t} + \mathrm{Z}S + S\mathrm{Z} \right] = \mu \Delta \mathrm{Z}} + \mu \Delta S - \nabla^2 p \, .$$

The terms embraced by the bracket constitute the vorticity transport equation (1.33). The remainder is the symmetric part

$$\rho \frac{\mathrm{d}S}{\mathrm{d}t} - \mu \Delta S + \rho (S^2 + \mathrm{Z}^2) = -\nabla^2 p.$$

The first term on the left-hand side represents unsteady straining, the second term viscous effects. As motivated above, both terms will not be considered which leaves $\rho(S^2 + \mathrm{Z}^2)$. The occurrence of a local pressure minimum in a plane requires two positive eigenvalues of $\nabla^2 p$. This in turn leads to the definition based on two negative eigenvalues of the matrix $S^2 + \mathrm{Z}^2$. Now $S^2 + \mathrm{Z}^2$ is symmetric,

$$(S^2 + \mathrm{Z}^2)^T = S^T S^T + \mathrm{Z}^T \mathrm{Z}^T = S^2 + (-1)^2 \mathrm{Z}^2 = S^2 + \mathrm{Z}^2$$

and is so diagonalizable with real eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3$. The sufficient condition for the pressure to take a minimum in a plane translates to

$$\text{connected regions of } \lambda_2 < 0 \tag{1.109}$$

which is the λ_2-criterion.

Connection between Q-criterion and λ_2-criterion

The quantity Q can also be related to the eigenvalues of $S^2 + \mathrm{Z}^2$:

$$-\frac{1}{2}\mathrm{tr}(S^2 + \mathrm{Z}^2) = -\frac{1}{2}\mathrm{tr}(S^T S) + \frac{1}{2}\mathrm{tr}(\mathrm{Z}^T \mathrm{Z}) = \frac{1}{2}(\mathrm{Z} : \mathrm{Z} - S : S) = Q$$

and on the other hand $-\frac{1}{2}\mathrm{tr}(S^2 + \mathrm{Z}^2) = -\frac{1}{2}(\lambda_1 + \lambda_2 + \lambda_3)$. This points out that those two definitions do not necessarily agree. They are the same indicators if $\lambda_2 < 0$ and $\sum_i \lambda_i < 0$ or if $\lambda_2 > 0$ and $\sum_i \lambda_i > 0$ but different for $\lambda_2 < 0$ and the sum being positive or vice versa.

1.4.5 Extension to two-dimensional flows

So far the review on the identification techniques assumed three-dimensional flow fields. We now want to point out how those measures can be extended to two dimensions.

At first, recall the discussion in 1.1.3, helicity cannot be used in 2D for extraction because it is identically zero.

Obviously the Q-criterion remains unchanged in 2D. The λ_2-criterion comes from the 2×2 matrix $S^2 + \Omega^2$ with eigenvalues $\lambda_1 \geq \lambda_2$. As in three dimensions take connected regions of $\lambda_2 < 0$ for the criterion. Moreover a simple calculation shows that in 2D $S^2 + \Omega^2 = \lambda_2 I$ and $\lambda_1 = \lambda_2$. So $Q = -\lambda_2$, both definitions agree and also correspond to a local pressure minimum.

2 Development of a simulation method for incompressible fluid flow

An efficient Navier-Stokes solver holds the key to solving general multiphysics problems that have an incompressible fluid model as a component. In this section we present the spatial discretization of the Navier-Stokes system with an interior penalty DG method taken from [GRMW04]. The convective term is discretized using the Vijayasundaram flux, [PMB18]. Operator splitting methods for solving the transient Navier-Stokes equations are employed such that the splitting is between incompressibility and dynamics. There are three classes of such time discretizations: (I) pressure-correction schemes (II) velocity-correction schemes (III) consistent splitting schemes, [GMS06, KIO91]. Here we concentrate on the pressure-correction schemes and consistent splitting schemes.

2.1 Discontinuous Galerkin discretization of the incompressible Navier-Stokes equations

The instationary incompressible Navier-Stokes equations in an open and bounded domain $\Omega \subset \mathbb{R}^d$ $(d = 2, 3)$ and time interval $(0, T]$ with velocity v and pressure p as unknowns for given right-hand side f, viscosity μ and density ρ are given by

$$\rho\partial_t v - \mu\Delta v + \rho(v \cdot \nabla)v + \nabla p = f \qquad \text{in } \Omega \times (0, T] \tag{2.1a}$$

$$\nabla \cdot v = 0 \qquad \text{in } \Omega \times (0, T] \tag{2.1b}$$

$$v = v_0 \qquad \text{for } t = 0\ . \tag{2.1c}$$

Either Dirichlet boundary condition for the velocity:

$$v = g \qquad \text{on } \Gamma_D = \partial\Omega, t \in (0, T] \tag{2.1d}$$

together with

$$\int_\Omega p \mathrm{d}x = 0 \qquad \text{for all } t \in (0, T] \tag{2.1e}$$

or mixed boundary conditions:

$$v = g \qquad \text{on } \Gamma_D \neq \{\partial\Omega, \emptyset\} \tag{2.1f}$$

$$\mu\nabla v n - pn - \rho\beta(v \cdot n)_- v = 0 \qquad \text{on } \Gamma_N = \partial\Omega \setminus \Gamma_D \tag{2.1g}$$

are supplemented with the system. For pure Dirichlet boundary conditions g is required to satisfy the compatibility condition $\int_{\partial\Omega} g \cdot n \mathrm{d}s = 0$. On Γ_N, $(v \cdot n)_- = \min(0, v \cdot n)$ denotes the negative part of the flux across the boundary. The parameter β can take the values $\beta = 0$ (classical do-nothing, CDN) or $\beta = \frac{1}{2}$ (directional do-nothing, DDN), [BM14]. In the numerical examples below we will also consider periodic boundary conditions in addition. Under appropriate assumptions the Navier-Stokes problem in weak form has a solution in $(H^1(\Omega))^d \times L^2(\Omega)$ for $t \in (0, T]$, [GR86, Té87]. In absence of do-nothing conditions the pressure is only determined up to a constant and is in the space $L^2_0(\Omega) = \{q \in L^2(\Omega) \mid \int_\Omega q \mathrm{d}x = 0\}$.

For the discretization let $\mathcal{E}_h$ be a quadrilateral mesh (in dimension $d = 2$) or a hexahedral mesh (in dimension $d = 3$) with maximum diameter h. We denote by Γ_h^{int} the set of all interior faces, by Γ_h^D the set of all faces intersecting with the Dirichlet boundary Γ_D and by Γ_h^N the set of all faces intersecting with the mixed boundary Γ_N. We set $\Gamma_h = \Gamma_h^{\text{int}} \cup \Gamma_h^D \cup \Gamma_h^N$. To an interior face $e \in \Gamma_h^{\text{int}}$ shared by elements E_e^1 and E_e^2 we define an orientation through its unit normal vector n_e pointing from E_e^1 to E_e^2. The jump and average of a scalar-valued function ϕ on a face is then defined by

$$\begin{aligned} [\phi] &= \phi\,|_{E_e^1} - \phi\,|_{E_e^2} \quad = \phi^{\text{int}} - \phi^{\text{ext}}, \\ \{\phi\} &= \frac{1}{2}\phi\,|_{E_e^1} + \frac{1}{2}\phi\,|_{E_e^2} = \frac{1}{2}\phi^{\text{int}} + \frac{1}{2}\phi^{\text{ext}}\ . \end{aligned} \tag{2.2}$$

Note that the definition of jump and average can be extended in a natural way to vector and matrix-valued functions. If $e \in \partial\Omega$ then n_e corresponds to the outer normal vector n. Below we make heavy use of the identities and notation, respectively:

$$\begin{aligned} [uv] &= [u]\{v\} + \{u\}[v]\ , & (u,v)_{0,\omega} &= \int_\omega uv \,\mathrm{d}x\ , & &(u, v \text{ scalar-valued}) \\ [u \cdot v] &= [u] \cdot \{v\} + \{u\} \cdot [v]\ , & (u,v)_{0,\omega} &= \int_\omega u \cdot v \,\mathrm{d}x\ , & &(u, v \text{ vector-valued}) \\ [u : v] &= [u] : \{v\} + \{u\} : [v]\ , & (u,v)_{0,\omega} &= \int_\omega u : v \,\mathrm{d}x\ , & &(u, v \text{ matrix-valued}). \end{aligned} \tag{2.3}$$

where $\omega \subset \Omega$ is a d-dimensional subset together with the d-dimensional measure $\mathrm{d}x$. The same shorthand notation holds for the hypersurface measure $\mathrm{d}s$ when integrating over codimension one subsets as (parts of) the boundary or possible collection of faces. The DG discretization on hexahedral meshes is based on the non-conforming finite element space of polynomial degree p

$$Q_h^p = \{v \in L^2(\Omega) \mid v|_E = q \circ \mu_E^{-1}, q \in \mathbb{Q}_{p,d}, E \in \mathcal{E}_h\} \tag{2.4}$$

where $\mu_E : \hat{E} \to E$ is the transformation from the reference cube $\hat{E}$ to E and $\mathbb{Q}_{p,d}$ is the set of polynomials of maximum degree p in d variables. The approximation spaces for velocity and pressure are then

$$X_h^p \times M_h^{p-1} = (Q_h^p)^d \times (Q_h^{p-1} \cap L^2_0(\Omega)) \qquad \text{(Dirichlet b. c.)}, \tag{2.5a}$$

$$X_h^p \times M_h^{p-1} = (Q_h^p)^d \times Q_h^{p-1} \qquad \text{(mixed b. c.)}. \tag{2.5b}$$

We make use of the following mesh-dependent forms defined on $X_h^p \times X_h^p$, $X_h^p \times M_h^{p-1}$, X_h^p and M_h^{p-1}, respectively:

$$a(u,v) = d(u,v) + J_0(u,v), \text{ where} \tag{2.6a}$$

$$d(u,v) = \sum_{E\in\mathcal{E}_h} (\mu\nabla u, \nabla v)_{0,E} - \sum_{e\in\Gamma_h^{\text{int}}} (\mu\{\nabla u\}n_e, [v])_{0,e} - \sum_{e\in\Gamma_h^D} (\mu\nabla u^{\text{int}} n_e, v^{\text{int}})_{0,e}, \tag{2.6b}$$

$$\begin{aligned} J_0(u,v) = {} & \epsilon \sum_{e\in\Gamma_h^{\text{int}}} (\mu\{\nabla v\}n_e, [u])_{0,e} + \epsilon \sum_{e\in\Gamma_h^D} (\mu\nabla v^{\text{int}} n_e, u^{\text{int}})_{0,e} \\ & + \sum_{e\in\Gamma_h^{\text{int}}} \mu\frac{\sigma}{h_e}([u],[v])_{0,e} + \sum_{e\in\Gamma_h^D} \mu\frac{\sigma}{h_e}(u^{\text{int}}, v^{\text{int}})_{0,e}, \end{aligned} \tag{2.6c}$$

$$b(v,q) = -\sum_{E\in\mathcal{E}_h} (\nabla\cdot v, q)_{0,E} + \sum_{e\in\Gamma_h^{\text{int}}} ([v]\cdot n_e, \{q\})_{0,e} + \sum_{e\in\Gamma_h^D} (v^{\text{int}}\cdot n, q^{\text{int}})_{0,e}, \tag{2.6d}$$

$$l(v;t) = \sum_{E\in\mathcal{E}_h} (f(t), v)_{0,E} + \epsilon \sum_{e\in\Gamma_h^D} (\mu\nabla v^{\text{int}} n_e, g(t))_{0,e} + \sum_{e\in\Gamma_h^D} \mu\frac{\sigma}{h_e}(g(t), v^{\text{int}})_{0,e}, \tag{2.6e}$$

$$r(q;t) = \sum_{e\in\Gamma_h^D} (g(t)\cdot n, q^{\text{int}})_{0,e}\ . \tag{2.6f}$$

Here we made the time dependence of the right hand side functionals explicit. For ease of writing this will be omitted mostly below. In the interior penalty parameter σ/h_e, the denominator accounts for the mesh dependence. The formula for h_e,

$$h_e = \begin{cases} \frac{\min\left(\left|E^{\text{int}}(e)\right|, \left|E^{\text{ext}}(e)\right|\right)}{|e|} & , E^{\text{int}}(e)\cap E^{\text{ext}}(e) = e \\ \frac{\left|E^{\text{int}}(e)\right|}{|e|} & , E^{\text{int}}(e)\cap \Gamma_D = e \end{cases},$$

has been stated in [HH08] where it was proven that this choice ensures coercivity of the bilinear form for anisotropic meshes. For σ we choose $\sigma = \alpha p(p+d-1)$ as in [BBS12] with α a user-defined parameter to be chosen $\alpha = 3$ in the computations reported below. In J_0 the *Symmetric Interior Penalty Galerkin* (SIPG) ($\epsilon = -1$) method is preferred since the matrix of the linear system in absence of the convection term is then symmetric. Other choices are the NIPG ($\epsilon = 1$) or IIPG ($\epsilon = 0$) method.

[Hil13] presents a rigorous analysis on the optimal penalty parameter where exact bounds from the trace inverse inequality for triangles, tetrahedra, quadrilaterals, hexahedra, wedges and pyramids are derived. See Table 3.1 in his doctoral dissertation. There are two conditions mentioned to ensure coercivity, Equation (3.22) and Equation (3.23). [Hil13] also verifies that an optimal penalty parameter is not sharply confined by these equations. The condition expressed in (3.22) is used in [KFWK17] and a related series of publications. The condition expressed in (3.23) gives the same mesh dependence on the penalty parameter as the formula for h_e. It is cheaper than the former condition which requires to iterate over all faces in the adjacent elements for each face. Further in Equation (3.23) setting the number of faces per element (n_e in his notation where e denotes an element) equal to $2d$ for quadrilaterals and hexahedra, and our choice of

$\alpha = 3$, yields a good agreement compared to the penalty parameter σ/h_e introduced above.

A first discretization of the nonlinear term in the Navier-Stokes equations is the standard (or centered) discretization,

$$c(v,\varphi) = \sum_{E\in\mathcal{E}_h} ((v\cdot\nabla)v,\varphi)_{0,E}, \tag{2.7}$$

which is applicable only for small Reynolds numbers. For higher Reynolds numbers the convective term is treated with an upwind scheme already introduced in [PMB18] that will be reviewed in Section 2.1.1 below:

$$c(v,\varphi) = -\sum_{E\in\mathcal{E}_h} (F(v),\nabla\varphi)_{0,E} + \sum_{e\in\Gamma_h^{\text{int}}} (\hat{F}_e(v,n_e),[\varphi])_{0,e} + \sum_{e\in\Gamma_h^D\cup\Gamma_h^N} (\hat{F}_e(v,n_e),\varphi)_{0,e}\ , \tag{2.8}$$

with the numerical fluxes

$$\hat{F}_e(v,n_e) = \begin{cases} \max(0,\{v\}\cdot n_e)v^{\text{int}} + \min(0,\{v\}\cdot n_e)v^{\text{ext}} & ,e\in\Gamma_h^{\text{int}} \\ \max(0,v^{\text{int}}\cdot n_e)v^{\text{int}} + \min(0,v^{\text{int}}\cdot n_e)g & ,e\in\Gamma_h^D \\ \max(0,v^{\text{int}}\cdot n_e)v^{\text{int}} & ,e\in\Gamma_h^N \end{cases}.$$

On the outflow boundary the variational form of the DDN contribution is

$$s_o(u,v) = \sum_{e\in\Gamma_h^N} ((u\cdot n)_- u, v)_{0,e}\ . \tag{2.9}$$

The discrete in space, continuous in time formulation of the Navier-Stokes problem (2.1) now seeks to find $v_h(t) : (0,T] \to X_h^p$, $p_h(t) : (0,T] \to M_h^{p-1}$:

$$\rho(\partial_t v_h,\varphi)_{0,\Omega} + a(v_h,\varphi) + \rho c(v_h,\varphi) - \rho\beta s_o(v_h,\varphi) + b(\varphi,p_h) = l(\varphi;t), \tag{2.10a}$$

$$b(v_h,q) = r(q;t), \tag{2.10b}$$

for all $(\varphi,q) \in X_h^p \times M_h^{p-1}$. The following observation will be used in several circumstances below.

Remark 1. The bilinear form $b(v,q)$ has the equivalent representation

$$b(v,q) = \sum_{E\in\mathcal{E}_h} (v,\nabla q)_{0,E} - \sum_{e\in\Gamma_h^{\text{int}}} (\{v\}\cdot n_e,[q])_{0,e} - \sum_{e\in\Gamma_h^N} (v\cdot n,q)_{0,e}. \tag{2.11}$$

This holds true for Dirichlet and mixed boundary conditions (in the former case just set $\Gamma_h^N = \emptyset$).

Proof. Follows from integration by parts and (2.3).

$$\begin{aligned} b(v,q) &= -\sum_{E\in\mathcal{E}_h}(\nabla\cdot v,q)_{0,E} + \sum_{e\in\Gamma_h^{\text{int}}}([v]\cdot n_e,\{q\})_{0,e} + \sum_{e\in\Gamma_h^D}(v\cdot n,q)_{0,e} \\ &= \sum_{E\in\mathcal{E}_h}[(v,\nabla q)_{0,E} - (v\cdot n_E,q)_{0,\partial E}] + \sum_{e\in\Gamma_h^{\text{int}}}([v]\cdot n_e,\{q\})_{0,e} + \sum_{e\in\Gamma_h^D}(v\cdot n,q)_{0,e} \\ &= \sum_{E\in\mathcal{E}_h}(v,\nabla q)_{0,E} + \sum_{e\in\Gamma_h^D}[(v\cdot n_e,q)_{0,e} - (v\cdot n_e,q)_{0,e}] - \sum_{e\in\Gamma_h^N}(v\cdot n,q)_{0,e} \\ &\quad + \sum_{e\in\Gamma_h^{\text{int}}}\left[([v]\cdot n_e,\{q\})_{0,e} - ([v]\cdot n_e,\{q\})_{0,e} - (\{v\}\cdot n_e,[q])_{0,e}\right] \\ &= \sum_{E\in\mathcal{E}_h}(v,\nabla q)_{0,E} - \sum_{e\in\Gamma_h^{\text{int}}}(\{v\}\cdot n_e,[q])_{0,e} - \sum_{e\in\Gamma_h^N}(v\cdot n_e,q)_{0,e}. \end{aligned} \tag{2.12}$$

□

As a corollary we obtain the following local mass conservation property by testing (2.10b) with $q=\chi_E$, the characteristic function of element E, and using Remark 1:

$$\sum_{e\in\Gamma_h^{\text{int}}\cap\partial E}(\{v\}\cdot n_e,1)_{0,e} + \sum_{e\in\Gamma_h^N\cap\partial E}(v\cdot n_e,1)_{0,e} + \sum_{e\in\Gamma_h^D\cap\partial E}(g\cdot n,1)_{0,e} = 0. \tag{2.13}$$

It is helpful to split the variational forms into volume, interface and boundary face contributions:

$$\begin{aligned} a(v_h,\varphi) &= \sum_{E\in\mathcal{E}_h}a_E(v_h,\varphi) + \sum_{e\in\Gamma_h^{\text{int}}}a_{\text{int},e}(v_h,\varphi) + \sum_{e\in\Gamma_h^D}a_{\Gamma_h^D,e}(v_h,\varphi)\ , \\ c(v_h,\varphi) &= \sum_{E\in\mathcal{E}_h}c_E(v_h,\varphi) + \sum_{e\in\Gamma_h^{\text{int}}}c_{\text{int},e}(v_h,\varphi) + \sum_{e\in\Gamma_h^D}c_{\Gamma_h^D,e}(v_h,\varphi) + \sum_{e\in\Gamma_h^N}c_{\Gamma_h^N,e}(v_h,\varphi)\ , \\ b(\varphi,p_h) &= \sum_{E\in\mathcal{E}_h}b_E(\varphi,p_h) + \sum_{e\in\Gamma_h^{\text{int}}}b_{\text{int},e}(\varphi,p_h) + \sum_{e\in\Gamma_h^D}b_{\Gamma_h^D,e}(\varphi,p_h). \end{aligned}$$

The DDN contribution $-\rho\beta s_o(v_h,\varphi)$ has a corresponding obvious splitting.

2.1.1 Upwind discretization of the convective part

For higher Reynolds numbers we employ a suitable upwind discretization based on the Vijayasundaram numerical flux adapted from DG methods for inviscid compressible flow [FFS03, DF04].

Note that due to $\nabla\cdot v=0$ the convective term in the momentum equations can be written equivalently as $(v\cdot\nabla)v=\nabla\cdot(v\otimes v)$ where

$$F(v) = v\otimes v = [v_1 v,\ldots,v_d v] = [F_1(v),\ldots,F_d(v)]$$

is the convective flux matrix with columns $F_k(v) = v_k v$ and $\nabla_v F_k(v) = (v_k I + v \otimes e_k)$. I denotes the identity matrix and $(e_k)_i = \delta_{ik}$ are the coordinate unit vectors. In order to derive the upwinding we consider the first order system

$$\partial_t v + \nabla \cdot F(v) = 0$$

which is said to be hyperbolic if the matrix

$$P(v,n) = \sum_{k=1}^{d} n_k \nabla_v F_k(v) = (v \cdot n) I + v \otimes n$$

is real diagonalizable for all $v, n \in \mathbb{R}^d$ with $\|n\| = 1$ [Eva10]. This is indeed the case for $v \cdot n \neq 0$. When $v \cdot n = 0$, $P(v,n) = v \otimes n$ has d eigenvalues zero with a corresponding eigenspace $W_n^\perp = \{w : w \cdot n = 0\}$ of dimension $d-1$.

When discretizing the conservative form of the convective terms with DG one uses element-wise integration by parts to arrive at

$$\begin{aligned} c(v,\varphi) = (\nabla \cdot F(v), \varphi)_{0,\Omega} &= \sum_{E \in \mathcal{E}_h} (\nabla \cdot F(v), \varphi)_{0,E} = -\sum_{E \in \mathcal{E}_h} (F(v), \nabla\varphi)_{0,E} + \sum_{E \in \mathcal{E}_h} (F(v)n, \varphi)_{0,\partial E} \\ &= -\sum_{E \in \mathcal{E}_h} (F(v), \nabla\varphi)_{0,E} + \sum_{e \in \Gamma_h^{\text{int}}} ([F(v)n_e \cdot \varphi], 1)_{0,e} + \sum_{e \in \Gamma_h^D \cup \Gamma_h^N} (F(v)n, \varphi)_{0,e} \, . \end{aligned}$$

Now the flux in face normal direction $F(v)n_e$ needs to be replaced by a *consistent* and *conservative* numerical flux function $\hat{F}(v, n_e)$ which we now derive. Since $F_k(v) = v_k v$ is homogeneous of degree 2 (i.e. $F_k(\alpha v) = \alpha^2 F_k(v)$ for α a real number) it admits a representation

$$F_k(v) = \frac{1}{2} \nabla_v F_k(v) v$$

and therefore

$$F(v)n = \frac{1}{2} P(v,n) v = \frac{1}{2} [(v \cdot n) I + v \otimes n] v =: B_{\frac{1}{2}}(v) v \, .$$

Using the identity $(v \cdot n)v = (v \otimes n)v$ we see

$$F(v)n = B_\beta(v) v := [(1-\beta)(v \cdot n) I + \beta v \otimes n] v$$

for any $\beta \in [0,1]$. For $v \cdot n \neq 0$, $B_\beta(v,n)$ is real diagonalizable with eigenvalues $\lambda_{\beta,i} \in \mathbb{R}$ and a full set of right eigenvectors r_i, $\text{span}\{r_1, \ldots, r_{d-1}\} = W_n^\perp$, $r_d = v$, admitting the decomposition

$$B_\beta(v,n) = B_\beta^+(v,n) + B_\beta^-(v,n),$$

where $B_\beta^\pm(v,n) = T D_\beta^\pm T^{-1}$, $T = [r_1, \ldots, r_d]$, $D_\beta^\pm$ are diagonal matrices with $(D_\beta^+)_{ii} = \max(0, \lambda_{\beta,i})$ and $(D_\beta^-)_{ii} = \min(0, \lambda_{\beta,i})$ (all eigenvectors and eigenvalues depending on v and n).

Following [DF04], in the DG scheme we employ the *Vijayasundaram* numerical flux given by

$$\hat{F}_\beta(v, n_e) = B_\beta^+(\{v\}, n_e)v^{\text{int}} + B_\beta^-(\{v\}, n_e)v^{\text{ext}} \,. \tag{2.14}$$

Here the matrices $B_\beta^\pm(\{v\}, n_e)$ are *not* applied to $\{v\}$ and therefore $(\{v\}\cdot n)I$ and $\{v\}\otimes n$ act differently. The effect is shown by the following

Observation 1. Assume $\{v\} \cdot n \neq 0$. Then the numerical flux (2.14) satisfies

$$\begin{aligned}\hat{F}_\beta(v, n_e) = (1-\beta)\left[\max(0, \{v\}\cdot n_e)v^{\text{int}} + \min(0, \{v\}\cdot n_e)v^{\text{ext}}\right] \\ + \beta[H(\{v\}\cdot n_e)(v^{\text{int}}\cdot n_e) + H(-\{v\}\cdot n_e)(v^{\text{ext}}\cdot n_e)]\{v\} \,.\end{aligned}$$

where $H(x)$ is the Heaviside function.

Proof. We consider the interior part. The eigenvectors of $B_\beta(\{v\}, n_e)$ are $d-1$ vectors spanning $W_n^\perp$ and $\{v\}$ independent of $\beta \in [0,1]$. We can uniquely decompose

$$v^{\text{int}} = \left(v^{\text{int}} - \frac{v^{\text{int}}\cdot n_e}{\{v\}\cdot n_e}\{v\}\right) + \frac{v^{\text{int}}\cdot n_e}{\{v\}\cdot n_e}\{v\} = w + \alpha\{v\}$$

where $w \in W_n^\perp$. Now

$$\begin{aligned}B_\beta^+(\{v\}, n_e)v^{\text{int}} &= B_\beta^+(\{v\}, n_e)(w + \alpha\{v\}) \\ &= (1-\beta)\max(0, \{v\}\cdot n_e)w + \max(0, \{v\}\cdot n_e)\alpha\{v\} \\ &= (1-\beta)\max(0, \{v\}\cdot n_e)(v^{\text{int}} - \alpha\{v\}) + \max(0, \{v\}\cdot n_e)\alpha\{v\} \\ &= (1-\beta)\max(0, \{v\}\cdot n_e)v^{\text{int}} + \beta\frac{\max(0, \{v\}\cdot n_e)(v^{\text{int}}\cdot n_e)}{\{v\}\cdot n_e}\{v\} \,.\end{aligned}$$

$B_\beta^-(\{v\}, n_e)v^{\text{ext}}$ can be treated in the same way. □

The observation shows that for $\beta > 0$ the $v \otimes n_e$ part gives a contribution in the flux in the direction of $\{v\}$, i.e. a central flux which moreover might have the wrong sign since the signs of $\{v\}\cdot n_e$ and $v^{\text{int}}\cdot n_e$ or $v^{\text{ext}}\cdot n_e$ might differ since the DG velocity is not in $H(\text{div}; \Omega)$. (Note, however, that the new projection scheme to be described below improves significantly on this point). Also note that the upwind decision is based on the average velocity which is locally mass conservative due to (2.13).

For these reasons we propose to employ $\beta = 0$ in the numerical flux function, leading to the simple form:

$$\hat{F}_e(v, n_e) = \begin{cases} \max(0, \{v\}\cdot n_e)v^{\text{int}} + \min(0, \{v\}\cdot n_e)v^{\text{ext}} & , e \in \Gamma_h^{\text{int}} \\ \max(0, v^{\text{int}}\cdot n_e)v^{\text{int}} + \min(0, v^{\text{int}}\cdot n_e)g & , e \in \Gamma_h^D \\ \max(0, v^{\text{int}}\cdot n_e)v^{\text{int}} & , e \in \Gamma_h^N \end{cases}$$

and the upwind DG discretization of the convective term

$$\hat{c}(v, \varphi) = -\sum_{E\in\mathcal{E}_h}(F(v), \nabla\varphi)_{0,E} + \sum_{e\in\Gamma_h^{\text{int}}}(\hat{F}_e(v, n_e), [\varphi])_{0,e} + \sum_{e\in\Gamma_h^D\cup\Gamma_h^N}(\hat{F}_e(v, n_e), \varphi)_{0,e} \,. \tag{2.15}$$

In the following computations we drop the hat in the variational form $\hat{c}$ and use it in solving equation (2.10a).

2.2 Projection methods

2.2.1 Continuous Helmholtz decomposition

The Helmholtz decomposition takes a fundamental role in the construction of splitting methods for incompressible flows. It states that any vector field in $L^2(\Omega)^d$ can be decomposed into a divergence-free contribution and an irrotational contribution, see e.g. [Cho68, KS05, AE04, Sch13, BNPB13]. In order to define the decomposition boundary conditions on the pressure need to be enforced which are not part of the underlying Navier-Stokes equations. The choice and consequence of these boundary conditions is a delicate issue in projection methods [Ran92, EL95, EL96]. Before turning to the Helmholtz decomposition in the discrete setting of DG methods we recall the Helmholtz decomposition in the weak continuous setting.

First consider Dirichlet boundary conditions (2.1d), (2.1e). Let us denote the space of weakly divergence free functions by

$$H(\Omega) := \{v \in L^2(\Omega)^d \mid (v, \nabla f)_{0,\Omega} - (g \cdot n, f)_{0,\Gamma_D} = 0 \; \forall f \in H^1(\Omega)\} \tag{2.16}$$

where $\Gamma_D = \partial\Omega$. This definition is motivated by the identity $(\nabla \cdot v, f)_{0,\Omega} = -(v, \nabla f)_{0,\Omega} + (g \cdot n, f)_{0,\Gamma_D} = 0$ which holds true for $v \in H(\mathrm{div}; \Omega) = \{u \in L^2(\Omega)^d \mid \nabla \cdot u \in L^2(\Omega)\}$. In that case the normal component of v can be prescribed on the boundary. In addition, we employ the pressure space

$$\Psi_D(\Omega) := \{q \in H^1(\Omega) \mid (q, 1)_{0,\Omega} = 0\}. \tag{2.17}$$

in the following decomposition.

Theorem 1 (Helmholtz decomposition, Dirichlet boundary conditions)**.** For any $w \in L^2(\Omega)^d$ there are unique functions $v \in H(\Omega)$ and $\psi \in \Psi_D(\Omega)$ such that

$$w = v + \nabla\psi.$$

Proof. Define $\psi \in \Psi_D(\Omega)$ by

$$(\nabla\psi, \nabla q)_{0,\Omega} = (w, \nabla q)_{0,\Omega} - (g \cdot n, q)_{0,\Gamma_D} \qquad \forall q \in \Psi_D(\Omega). \tag{2.18}$$

According to the Lax-Milgram theorem this problem has a unique solution. Since any $f \in H^1(\Omega)$ can be written as $f = q + c$ with $q \in \Psi_D(\Omega)$ and c a constant function, equation (2.18) holds also true for all test functions in $H^1(\Omega)$ (Note the compatibility condition on g). Now set $v = w - \nabla\psi$ and verify that $(v, \nabla f)_{0,\Omega} - (g \cdot n, f)_{0,\Gamma_D} = 0$ for all $f \in H^1(\Omega)$. □

Remark 2. 1) Note that equation (2.18) is the weak formulation of a Poisson equation with homogeneous Neumann boundary conditions.

2) The map $\mathcal{P} : L^2(\Omega)^d \to H(\Omega)$ given by $\mathcal{P}w = w - \nabla\psi$ is a projection since the right hand side of (2.18) is zero for $w \in H(\Omega)$. $\mathcal{P}$ is called the *continuous Helmholtz projection*.

3) The construction above can be equivalently written as

$$(v, \varphi)_{0,\Omega} + (\mathbb{\nabla}\psi, \varphi)_{0,\Omega} = (w, \varphi)_{0,\Omega} \qquad \forall \varphi \in L^2(\Omega)^d \tag{2.19a}$$

$$(v, \mathbb{\nabla} q)_{0,\Omega} = (g \cdot n, q)_{0,\Gamma_D} \qquad \forall q \in \Psi_D(\Omega) \tag{2.19b}$$

since from the first equation we get $v = w - \mathbb{\nabla}\psi$ and inserting in the second equation yields (2.18).

4) In Chorin's classical projection scheme [Cho68] the (divergence-free) velocity v^{k+1} and pressure p^{k+1} at time t^{k+1} are computed from a tentative velocity w^{k+1} by the system

$$\begin{aligned} \frac{v^{k+1} - w^{k+1}}{\Delta t} + \mathbb{\nabla} p^{k+1} &= 0 \\ \nabla \cdot v^{k+1} &= 0 \end{aligned}$$

in strong form. Setting $\psi^{k+1} = \Delta t p^{k+1}$ this is equivalent to

$$\begin{aligned} v^{k+1} + \mathbb{\nabla}\psi^{k+1} &= w^{k+1} \\ \nabla \cdot v^{k+1} &= 0 \end{aligned}$$

which is the strong form of (2.19). Thus, $\psi/\Delta t$ from the Helmholtz decomposition is the new pressure from Chorin's projection scheme. □

In the case of mixed boundary conditions (2.1f), (2.1g) the space $\Psi_D(\Omega)$ is replaced by

$$\Psi_M(\Omega) := \{q \in H^1(\Omega) \mid q = 0 \text{ a.e. on } \Gamma_N\} \tag{2.20}$$

employing homogeneous Dirichlet boundary conditions on Γ_N. This can be understood from (2.1g) which implies $p \approx 0$ for small μ, i.e. large Reynolds number. The irrotational part is defined as in (2.18) with $\Psi_D(\Omega)$ replaced by $\Psi_M(\Omega)$, meaning that ψ satisfies homogeneous Neumann conditions on Γ_D and homogeneous Dirichlet conditions on Γ_N. Again, $v \in H(\Omega)$ is uniquely defined (observe that now $\Gamma_D \subset \partial\Omega$ in $H(\Omega)$).

2.2.2 Discrete Helmholtz decomposition

We now seek discrete versions $\mathcal{P}_h : X_h^p \to X_h^p$ of the Helmholtz projection operator $\mathcal{P}$. A direct reconstruction of the weakly divergence free velocity as $v = w - \mathbb{\nabla}\psi$ in DG splitting schemes is reported to be unstable when the spatial mesh is coarse and the time step is small [SSL13, JDS+16, KFWK17] and several local postprocessing techniques are discussed in the literature. Here we propose a new postprocessing technique based on $H(\mathrm{div})$ reconstruction which is popular in porous media flows [BR03, ENV07]. These reconstructions are element-local, easy to compute and provide a locally mass conservative projected velocity, a property not shared by the reconstructions in [SSL13, KFWK17]. [JDS+16] takes into account inter-element continuity in a regularized least-squares sense but does not provide a projection. The construction presented here is easier to compute, provides exact local mass conservation, satisfies the discrete continuity equation exactly and provides a projection.

Standard projection

For any given tentative velocity $w_h \in X_h^p$ the straightforward translation of the Helmholtz decomposition (2.19) in the DG setting reads

$$(v_h, \varphi)_{0,\Omega} + (\nabla_h \psi_h, \varphi)_{0,\Omega} = (w_h, \varphi)_{0,\Omega} \qquad \forall \varphi \in X_h^p, \tag{2.21a}$$

$$b(v_h, q) = r(q) \qquad \forall q \in M_h^{p-1}. \tag{2.21b}$$

Note that the second equation requires the projected velocity to satisfy the discrete form of the continuity equation (2.10b) at fixed time (hence silently dropping the time dependence from now). From the first condition (2.21a) we get $v_h + \nabla_h \psi_h = w_h \Leftrightarrow v_h = w_h - \nabla_h \psi_h$ since all involved functions are in X_h^p. Inserting this into (2.21b) yields an equation for ψ_h:

$$b(\nabla \psi_h, q) = b(w_h, q) - r(q) \quad \forall q \in M_h^{p-1}.$$

Using Remark 1 on the left hand side we get

$$b(\nabla \psi_h, q) = \sum_{E \in \mathcal{E}_h} (\nabla \psi_h, \nabla q)_{0,E} - \sum_{e \in \Gamma_h^{\text{int}}} (\{\nabla \psi_h\} \cdot n_e, [q])_{0,e} - \sum_{e \in \Gamma_h^N} (\nabla \psi_h \cdot n_e, q)_{0,e}. \tag{2.22}$$

This is part of the standard SIPG formulation of Poisson's equation with homogeneous Neumann boundary conditions on Γ_h^D with the stabilization terms missing. In order to stabilize, we define

$$\begin{aligned} j_0(\psi_h, q) = &- \sum_{e \in \Gamma_h^{\text{int}}} (\{\nabla q\} \cdot n_e, [\psi_h])_{0,e} + \sum_{e \in \Gamma_h^{\text{int}}} \frac{\sigma}{h_e} ([q], [\psi_h])_{0,e} \\ &- \sum_{e \in \Gamma_h^N} (\nabla q \cdot n_e, \psi_h)_{0,e} + \sum_{e \in \Gamma_h^N} \frac{\sigma}{h_e} (q, \psi_h)_{0,e}. \end{aligned} \tag{2.23}$$

and solve the *stabilized version*

$$\psi_h \in M_h^{p-1} : \quad \alpha(\psi_h, q) = b(w_h, q) - r(q) \quad \forall q \in M_h^{p-1} \tag{2.24}$$

where

$$\alpha(\psi_h, q) = b(\nabla \psi_h, q) + j_0(\psi_h, q).$$

Note that this system naturally corresponds to homogeneous Neumann conditions on Γ_D and homogeneous Dirichlet conditions on Γ_N (which might be empty). Now we may define the first projection scheme.

Algorithm 2.1 The standard projection $\mathcal{P}_h^{\text{std}}$ is given by the following algorithm:

---

i) For any tentative velocity $w_h \in X_h^p$ and fixed t solve

$$\psi_h \in M_h^{p-1} : \quad \alpha(\psi_h, q_h) = b(w_h, q_h) - r(q_h; t) \quad \forall q_h \in M_h^{p-1}. \tag{2.25}$$

ii) Set $\mathcal{P}_h^{\text{std}} w_h = v_h$ where v_h solves

$$(v_h, \varphi_h)_{0,\Omega} = (w_h, \varphi_h)_{0,\Omega} - (\nabla \psi_h, \varphi_h)_{0,\Omega} \quad \forall \varphi_h \in X_h^p. \tag{2.26}$$

This requires the solution of a mass matrix which is block-diagonal. Choosing an orthogonal basis it can even be diagonal and thus the computation is cheap. Note also that this implies $v_h = w_h - \nabla_h \psi_h$ since $\nabla_h \psi_h \in X_h^p$.

Unfortunately, this projection is reported to be unstable in the small time step limit [SSL13] and we also observed this behavior. Part of the problem is that $\mathcal{P}_h^{\text{std}}$ is actually not a projection, i.e. $(\mathcal{P}_h^{\text{std}})^2 \neq \mathcal{P}_h^{\text{std}}$.

Div-div projection

In order to overcome the stability problem the authors in [KFWK17] suggested to stabilize the projection by an additional term in (2.26):

Algorithm 2.2 The div-div projection $\mathcal{P}_h^{\text{div-div}}$ is given by the following algorithm:

---

i) For any tentative velocity $w_h \in X_h^p$ and fixed t solve (same as before)

$$\psi_h \in M_h^{p-1} : \quad \alpha(\psi_h, q_h) = b(w_h, q_h) - r(q; t) \quad \forall q_h \in M_h^{p-1}.$$

ii) Set $\mathcal{P}_h^{\text{div-div}} w_h = v_h$ where v_h solves

$$(v_h, \varphi_h)_{0,\Omega} + \tau_D (\nabla \cdot v_h, \nabla \cdot \varphi_h)_{0,\Omega} = (w_h, \varphi_h)_{0,\Omega} - (\nabla \psi_h, \varphi_h)_{0,\Omega} \quad \forall \varphi_h \in X_h^p \tag{2.27}$$

where τ_D is a user-supplied constant.

Again this requires the solution of an element-local system which is not diagonal. As reported in [KFWK17] and the examples below this gives good results with quite small point-wise divergence. However, the projected velocity does not satisfy a local mass conservation property and $(\mathcal{P}_h^{\text{div-div}})^2 \neq \mathcal{P}_h^{\text{div-div}}$

Pressure Poisson Raviart-Thomas projection

The aim of this subsection is to reconstruct $-\nabla \psi_h$ in the Raviart-Thomas space of degree k [BF91] on hexahedral meshes given by

$$\text{RT}_h^k = \{v \in H(\text{div}; \Omega) \mid v|_E \in \text{RT}_E^k \, \forall E \in \mathcal{E}_h\} \tag{2.28}$$

with the Raviart-Thomas space on element E given by

$$\text{RT}_E^k = \{v \in H(\text{div}; E) \mid v = T_E(\hat{v}), (\hat{v})_i = \sum_{\{\alpha | 0 \leq \alpha_j \leq k + \delta_{ij}\}} c_{i,\alpha} \hat{x}^\alpha\} \tag{2.29}$$

where we made use of the Piola transformation to the element $E \in \mathcal{E}_h$, i.e. for $\mu_E(\hat{x}) : \hat{E} \to E$ defined as

$$T_E(\hat{v})(x) = \frac{1}{|\det \nabla \mu_E(\hat{x})|} \nabla \mu_E(\hat{x}) \hat{v}(\hat{x}).$$

For $k > 0$ the construction needs also the space

$$\Psi_E^k = \{v \in H(\mathrm{div}; E) \mid v = T_E(\hat{v}), (\hat{v})_i = \sum_{\{\alpha | 0 \le \alpha_j \le k - \delta_{ij}\}} c_{i,\alpha} \hat{x}^\alpha\}. \tag{2.30}$$

Note that in contrast to (2.29) the polynomial degree in direction i in component i is decreased instead of increased.

Assume that $\psi_h \in M_h^{p-1}$ solves (2.24) as before. Following [ENV07] we now compute $\gamma_h = G_h \psi_h \in \mathrm{RT}_h^k$, $k = p-1$, as reconstruction of $-\nabla \psi_h$ as follows. On element $E \in \mathcal{E}_h$ with faces $e \in \partial E$ define

$$(\gamma_h \cdot n_e, q)_{0,e} = (-\{\nabla \psi_h\} \cdot n_e + \frac{\sigma}{h_e}[\psi_h], q)_{0,e} \qquad e \in \Gamma_h^{\mathrm{int}}, q \in Q_e^k, \tag{2.31a}$$

$$(\gamma_h \cdot n_e, q)_{0,e} = (-\nabla \psi_h \cdot n_e + \frac{\sigma}{h_e}\psi_h, q)_{0,e} \qquad e \in \Gamma_h^N, q \in Q_e^k, \tag{2.31b}$$

$$(\gamma_h \cdot n_e, q)_{0,e} = 0 \qquad e \in \Gamma_h^D, q \in Q_e^k, \tag{2.31c}$$

and for $k > 0$ define in addition

$$(\gamma_h, r)_{0,E} = -(\nabla \psi_h, r)_{0,E} + \frac{1}{2} \sum_{e \in \partial E \cap \Gamma_h^{\mathrm{int}}} (r \cdot n_e, [\psi_h])_{0,e} + \sum_{e \in \partial E \cap \Gamma_h^N} (r \cdot n_e, \psi_h)_{0,e}, \quad \forall r \in \Psi_E^k. \tag{2.31d}$$

With this we can define our final projection step:

Algorithm 2.3 The pressure Poisson RT projection $\mathcal{P}_h^{\mathrm{RT}}$ is given by the following algorithm:

---

i) For any tentative velocity $w_h \in X_h^p$ and fixed t solve

$$\psi_h \in M_h^{p-1} : \quad \alpha(\psi_h, q_h) = b(w_h, q_h) - r(q_h; t) \quad \forall q_h \in M_h^{p-1}.$$

ii) Reconstruct $\gamma_h = G_h \psi_h \in \mathrm{RT}_h^{p-1}$.

iii) Set $\mathcal{P}_h^{\mathrm{RT}} w_h = v_h$ where v_h solves

$$(v_h, \varphi_h)_{0,\Omega} = (w_h, \varphi_h)_{0,\Omega} + (G_h \psi_h, \varphi_h)_{0,\Omega} \quad \forall \varphi_h \in X_h^p.$$

This requires the solution of a (block-) diagonal system.

The reconstruction G_h defined above satisfies the following important property on affine hexahedral meshes.

Lemma 1. Let $\psi_h \in M_h^{p-1}$ solve $\alpha(\psi_h, q) = l(q)$ for all $q \in M_h^{p-1}$ and any linear right hand side functional l. Let furthermore $\gamma_h = G_h\psi_h \in \mathrm{RT}_h^{p-1}$ be the reconstruction defined above. Then for every $q \in Q_h^{p-1}$ and χ_E the characteristic function of element $E \in \mathcal{E}_h$ we have

$$(\nabla \cdot \gamma_h, q\chi_E)_{0,E} = l(q\chi_E). \tag{2.32}$$

Proof. Straightforward extension of Theorem 3.1 in [ENV07] from simplicial to hexahedral elements. Essential steps are the special definition of the right-hand side in (2.31d) and that for any $q \in Q_h^{p-1} \Rightarrow \mathbb{\nabla} q|_E \in \Psi_E^{p-1}$ where the condition on an affine transformation is only required. The calculation in detail:

$$\begin{aligned}(\nabla \cdot \gamma_h, q\chi_E)_{0,E} &= -(\gamma_h, \mathbb{\nabla} q\chi_E)_{0,E} + \sum_{e\in\partial E} (\gamma_h \cdot n_e, q\chi_E)_{0,e} \\ &\overset{(2.31d)}{=} \alpha(\psi_h, q\chi_E) = l(q\chi_E)\ .\end{aligned}$$

□

And with this lemma we can prove the following theorem.

Theorem 2. The projected velocity $\mathcal{P}_h^{\mathrm{RT}} w_h$ satisfies the discrete continuity equation exactly, i.e.

$$b(\mathcal{P}_h^{\mathrm{RT}} w_h, q) = r(q) \qquad \forall q \in M_h^{p-1}. \tag{2.33}$$

Proof. The characteristic functions form a partition of unity, i.e. for any $q \in Q_h^{p-1}$ we have $q = \sum_{E\in\mathcal{E}_h} q\chi_E$. Inserting into the definition of b, observing that $[\gamma_h] \cdot n_e = 0$ since $\gamma_h \in H(\mathrm{div}; \Omega)$ as well as $\gamma_h \cdot n_e = 0$ due to (2.31c) and using Lemma 1 gives:

$$\begin{aligned}b(\mathcal{P}_h^{\mathrm{RT}} w_h, q) &= b(w_h, q) + b(G_h\psi_h, q) \\ &= b(w_h, q) - \sum_{E\in\mathcal{E}_h} (\nabla \cdot \gamma_h, q\chi_E)_{0,E} + \sum_{e\in\Gamma_h^{\mathrm{int}}} ([\gamma_h] \cdot n_e, \{q\})_{0,e} + \sum_{e\in\Gamma_h^D} (\gamma_h \cdot n_e, q)_{0,e} \\ &= b(w_h, q) - \sum_{E\in\mathcal{E}_h} l(q\chi_E) = b(w_h, q) - \sum_{E\in\mathcal{E}_h} [b(w_h, q\chi_E) - r(q\chi_E)] \\ &= b(w_h, q) - b(w_h, q) + r(q) = r(q)\end{aligned} \tag{2.34}$$

□

Remark 3. As corollaries we have

1) The projected velocity $v_h = \mathcal{P}_h^{\mathrm{RT}} w_h$ satisfies the discrete conservation property (2.13) (use the fact $\chi_E \in M_h^{p-1}$ and Theorem 2). Note that this discrete conservation property can be achieved with reconstruction in Raviart-Thomas space with degree $k \leq p-1$.

2) $(\mathcal{P}_h^{\mathrm{RT}})^2 = \mathcal{P}_h^{\mathrm{RT}}$ follows from Theorem 2 and the fact that $l(q) = b(\mathcal{P}_h^{\mathrm{RT}} w_h, q) - r(q) = 0, q \in M_h^{p-1}$, is the right-hand side in step i) of Algorithm 2.3. Therefore when applying $\mathcal{P}_h^{\mathrm{RT}}$ twice a zero correction is produced in the second application.

The discrete continuity equation does not imply that the divergence of the projected velocity vanishes point-wise. The following Lemma shows that the divergence in the interior of elements is controlled in an integral sense only by the jumps of the tentative velocity:

Lemma 2. The projected velocity $v_h = \mathcal{P}_h^{\mathrm{RT}} w_h$ satisfies for all $q \in M_h^{p-1}$, $E \in \mathcal{E}_h$ and $q_E = q\chi_E$:

$$(\nabla \cdot v_h, q_E)_{0,E} = \frac{1}{2} \sum_{e \in \Gamma_h^{\mathrm{int}} \cap \partial E} ([w_h] \cdot n_e, q_E)_{0,e} + \sum_{e \in \Gamma_h^D \cap \partial E} ((w_h - g) \cdot n_e, q_E)_{0,e}. \tag{2.35}$$

Proof. Using Lemma 1 we get

$$\begin{aligned}(\nabla \cdot v_h, q_E)_{0,E} &= (\nabla \cdot w_h, q_E)_{0,E} + (\nabla \cdot \gamma_h, q_E)_{0,E} = (\nabla \cdot w_h, q_E)_{0,E} + l(q_E) \\ &= (\nabla \cdot w_h, q_E)_{0,E} + b_h(w_h, q_E) - r(q_E) \\ &= (\nabla \cdot w_h, q_E)_{0,E} - (\nabla \cdot w_h, q_E)_{0,E} \\ &\quad + \frac{1}{2} \sum_{e \in \Gamma_h^{\mathrm{int}} \cap \partial E} ([w_h] \cdot n_e, q_E)_{0,e} + \sum_{e \in \Gamma_h^D \cap \partial E} ((w_h - g) \cdot n_e, q_E)_{0,e} \,.\end{aligned} \tag{2.36}$$

□

Helmholtz-flux Raviart-Thomas projection

In this subsection we present a reconstruction of the Helmholtz-flux $w_h - \mathbb{\nabla}\psi_h$ in the Raviart-Thomas space of degree k that does not only satisfy the discrete continuity equation but is also pointwise divergence-free. The development of Helmholtz-flux Raviart-Thomas projection originated from studying *pressure-robust* discretizations of the incompressible Navier-Stokes equations. In pressure-robust discretizations the velocity does not depend on the pressure and remains unchanged under certain types of transformation - to be described below - as it would in the continuous case. It is understood that a violation of this so-called *invariance property* is connected to the Helmholtz decomposition such that discretely divergence-free vectors need not be divergence-free vector fields. Recently the following results were shown with respect to pressure robustness: [Lin14] presents a modified Crouzeix-Raviart element for the incompressible Navier-Stokes equations. [JLM$^+$17] derives a reconstruction operator for the mixed finite element pair on triangles consisting of conforming $\mathcal{P}_2$ space for velocity enriched with bubble functions and discontinuous $\mathcal{P}_1$ for pressure. [PELS16] presents a higher-order reconstruction operator on simplicial meshes for a discontinuous method where the cell-based unknowns are eliminated by static condensation. [LLMS17] develops a reconstruction for pressure-robust Stokes discretizations with continuous pressure finite elements. However to the best of our knowledge, no results have been presented first of all for discontinuous velocity and pressure spaces, on quadrilateral/hexahedral meshes.

Divergence-preserving reconstruction operator In order to present a divergence-preserving operator for both discontinuous velocity and pressure spaces, let us introduce the

discrete divergence operator B_h. It is a map $B_h : X_h^p \to Q_h^{p-1}$ satisfying

$$(B_h w_h, q_h)_{0,\Omega} = -b(w_h, q_h) + r(q_h) \quad \forall q_h \in Q_h^{p-1} . \tag{2.37}$$

The kernel of B_h is called the set of *discretely divergence-free* vector fields. It is an affine linear subspace of X_h^p denoted by

$$X_{h,div}^p := \{v_h \in X_h^p \mid b(v_h, q_h) = r(q_h) \; \forall q_h \in Q_h^{p-1}\} \tag{2.38}$$

with the belonging vector space

$$X_{h,div}^{p,hom} := \{\varphi_h \in X_h^p \mid b(\varphi_h, q_h) = 0 \; \forall q_h \in Q_h^{p-1}\} . \tag{2.39}$$

Next we introduce the reconstruction operator $\Pi_h^{\mathrm{RT}_h^{p-1}}$ that maps the velocity space X_h^p to the Raviart-Thomas space RT_h^{p-1} of degree $p-1$. Based on the results presented in [Lin14, JLM+17, PELS16] we define $v_h = \Pi_h^{\mathrm{RT}_h^{p-1}}(w_h)$, $w_h \in X_h^p$, as follows: On element $E \in \mathcal{E}_h$ with faces $e \in \partial E$ compute

$$(v_h \cdot n_e, q_h)_{0,e} = (\{w_h\} \cdot n_e, q_h)_{0,e} \qquad e \in \Gamma_h^{\mathrm{int}}, q \in Q_e^{p-1} , \tag{2.40}$$

$$(v_h \cdot n_e, q_h)_{0,e} = (g \cdot n_e, q_h)_{0,e} \qquad e \in \Gamma_h^{D}, q \in Q_e^{p-1} , \tag{2.41}$$

$$(v_h \cdot n_e, q_h)_{0,e} = (w_h \cdot n_e, q_h)_{0,e} \qquad e \in \Gamma_h^{N}, q \in Q_e^{p-1} , \tag{2.42}$$

and in addition for $p - 1 > 0$

$$(v_h, r_h)_{0,E} = (w_h, r_h)_{0,E} \qquad \forall r \in \Psi_E^{p-1} . \tag{2.43}$$

Obviously this defines a projection. Further its divergence is in Q_h^{p-1} and on affine hexahedral meshes we can show:

Theorem 3. The velocity reconstruction operator $\Pi_h^{\mathrm{RT}_h^{p-1}}$ defined through the equations (2.40)- (2.43) is divergence-preserving. In explicit it holds

$$\nabla \cdot \Pi_h^{\mathrm{RT}_h^{p-1}} = B_h . \tag{2.44}$$

Proof. Same line of argument as in proof of Lemma 1.

$$\begin{aligned}
(\nabla \cdot \Pi_h^{\mathrm{RT}_h^{p-1}}(w_h), q_h)_{0,\Omega} &= \sum_{E \in \mathcal{E}_h} (\nabla \cdot \Pi_h^{\mathrm{RT}_h^{p-1}}(w_h), q_h)_{0,E} \\
&= \sum_{E \in \mathcal{E}_h} -(\Pi_h^{\mathrm{RT}_h^{p-1}}(w_h), \nabla q_h)_{0,E} + \sum_{e \in \partial E} (\Pi_h^{\mathrm{RT}_h^{p-1}}(w_h) \cdot n_e, q_h)_{0,e} \\
&= -\sum_{E \in \mathcal{E}_h} (\Pi_h^{\mathrm{RT}_h^{p-1}}(w_h), \nabla q_h)_{0,E} + \sum_{e \in \Gamma_h} (\Pi_h^{\mathrm{RT}_h^{p-1}}(w_h) \cdot n_e, [q_h])_{0,e}
\end{aligned}$$

since $\left[\Pi_h^{\mathrm{RT}_h^{p-1}}(w_h)\right] \cdot n_e = 0$ for $e \in \Gamma_h^{\mathrm{int}}$ in the last term. Further

$$\begin{aligned}
&= -\sum_{E\in\mathcal{E}_h} (w_h, \nabla\!\!\!\!\nabla q_h)_{0,E} + \sum_{e\in\Gamma_h^N} (w_h \cdot n_e, q_h)_{0,e} \\
&\quad + \sum_{e\in\Gamma_h^{\mathrm{int}}} (\{w_h\} \cdot n_e, [q_h])_{0,e} + \sum_{e\in\Gamma_h^D} (g(t) \cdot n_e, q_h)_{0,e} \\
&= -b(w_h, q_h) + r(q_h) \qquad \text{using Remark 1} \\
&= (B_h w_h, q_h)_{0,\Omega} \, .
\end{aligned}$$

Finally, because $\nabla \cdot \Pi_h^{\mathrm{RT}_h^{p-1}}(w_h) \in Q_h^{p-1}$, we have $\nabla \cdot \Pi_h^{\mathrm{RT}_h^{p-1}}(w_h) = B_h w_h$. □

Lemma 3. The operator $\Pi_h^{\mathrm{RT}_h^{p-1}}$ maps discretely divergence-free vector fields to divergence-free vector fields. The image also satisfies the discrete continuity equation.

Proof. Let $v_h \in X_{h,div}^p$. Then according to Theorem 3 $\nabla \cdot \Pi_h^{\mathrm{RT}_h^{p-1}}(v_h) = B_h v_h = 0$. Writing this in the weak sense, we can add zeroes

$$\begin{aligned}
0 &= (\nabla \cdot \Pi_h^{\mathrm{RT}_h^{p-1}}(v_h), q_h)_{0,\Omega} \\
&= (\nabla \cdot \Pi_h^{\mathrm{RT}_h^{p-1}}(v_h), q_h)_{0,\Omega} - \sum_{e\in\Gamma_h^{\mathrm{int}}} (\left[\Pi_h^{\mathrm{RT}_h^{p-1}}(v_h)\right] \cdot n_e, \{q_h\})_{0,e} - \sum_{e\in\Gamma_h^D} (\Pi_h^{\mathrm{RT}_h^{p-1}}(v_h) \cdot n_e, q_h)_{0,e} \\
&\quad + \sum_{e\in\Gamma_h^D} (\Pi_h^{\mathrm{RT}_h^{p-1}}(v_h) \cdot n_e, q_h)_{0,e} \\
&= -b\left(\Pi_h^{\mathrm{RT}_h^{p-1}}(v_h), q_h\right) + r(q_h)
\end{aligned}$$

to show the second statement. □

Invariance property and pressure robustness A consequence of the Helmholtz decomposition is an invariance property of the Navier-Stokes equations that reads: A transformation of the forcing by $f \mapsto f + \nabla\!\!\!\!\nabla\psi$ changes the Navier-Stokes solution by $(v, p) \mapsto (v, p + \psi)$, i.e. the velocity field does not change and the additional forcing is balanced by the pressure gradient. A desirable property of a discretization scheme is to maintain an unchanged velocity field under such irrotational force translation.

As a simple example let us consider the mixed DG-discretization of the stationary incompressible Stokes equations. Find $(v_h, p_h) \in X_h^p \times M_h^{p-1}$ s.t.

$$\begin{aligned}
a(v_h, \varphi_h) + b(\varphi_h, p_h) &= l(\varphi_h) \quad \forall \varphi_h \in X_h^p \\
b(v_h, q_h) &= r(q_h) \quad \forall q_h \in M_h^{p-1} \, .
\end{aligned}$$

In [GRMW04, RG06] it was shown to be a inf-sup stable method. Employing a k-th polynomial order approximation $X_h^k \times M_h^{k-1}$ to the strong solution (v, p) of the Stokes problem, it was also shown

$$\left[\!\left[\|v - v_h\|\right]\!\right] \leq C h^k \left(|v|_{k+1} + \frac{1}{\mu} |p|_k \right)$$

with the mesh-dependent norm $|\!|\!| \cdot |\!|\!|$,

$$\begin{aligned}|\!|\!|v_h|\!|\!|^2 &:= \|v_h\|_{0,\Omega}^2 + |\!|v_h|\!|^2 \\ &= (v_h, v_h)_{0,\Omega} + \sum_{E\in\mathcal{E}_h} (\nabla v_h, \nabla v_h)_{0,E} + \sum_{e\in\Gamma_h^{\text{int}}\cup\Gamma_h^D} \frac{\sigma}{h_e}([v_h],[v_h])_{0,e} \,.\end{aligned}$$

The right-hand side in the a-priori estimate is a standard bound for classical mixed finite element methods. Observe that the velocity error depends on the pressure scaled by the inverse of viscosity which can become large if μ is small. Moreover it is not robust with respect to irrotational forces - that would be balanced solely by the pressure in the continuous case - and a discretization suffers spurious velocity oscillations. The pressure part $\frac{1}{\mu}h^k|p|_k$ originates for non-divergence-free methods and it is understood that the pressure term is indeed equivalent to the fact that such mixed methods do not fulfill the invariance property in a discrete sense. It can be avoided by divergence-free mixed methods as the Scott-Vogelius element for example. For instance in the mixed DG-discretization of the Stokes problem, vector fields satisfying (2.21) do not necessarily satisfy (2.19). Discretely divergence-free test functions lack a L^2-orthogonality to the irrotational fields. However with help of the introduced reconstruction operator, φ_h is replaced as in previous publications by the reconstructed φ_h in the source term. Motivation is just Lemma 3 which states discretely divergence-free functions are mapped to divergence-free functions. With such a variational modification we can prove that a discrete counterpart of the invariance property is established.

Theorem 4. Let $\Pi_h^{\mathrm{RT}_h^{p-1}}$ be the test function reconstruction operator defined through the equations (2.40)- (2.43) with zero g. Let $\pi_h^{Q_h^{p-1}} : L^2(\Omega) \to Q_h^{p-1}$ be the L^2-projection. The modified mixed scheme for the Stokes problem reads: Find $(v_h, p_h) \in X_h^p \times M_h^{p-1}$ s.t.

$$a(v_h, \varphi_h) + b(\varphi_h, p_h) = \tilde{l}(\varphi_h) \quad \forall \varphi_h \in X_h^p \tag{2.45}$$

$$b(v_h, q_h) = r(q_h) \quad \forall q_h \in M_h^{p-1} \tag{2.46}$$

with the modified right-hand side functional $\tilde{l}$

$$\tilde{l}(\varphi_h) - l(\varphi_h) = (f, \Pi_h^{\mathrm{RT}_h^{p-1}}(\varphi_h) - \varphi_h)_{0,\Omega} \,. \tag{2.47}$$

Then for a irrotational translation $f \mapsto f + \mathbb{\nabla}\psi$ where (1) $\psi \in \Psi_D(\Omega)$ if $\Gamma_D = \partial\Omega$ defined in (2.17) or (2) $\psi \in \Psi_M(\Omega)$ if $\Gamma_D \neq \{\emptyset, \partial\Omega\}$ defined in (2.20), the scheme fulfills a discrete counterpart to the invariance property: $(v_h, p_h) \mapsto (v_h, p_h + \pi_h^{Q_h^{p-1}}(\psi))$.

Proof. Same line of argument when showing the invariance property in the weak continuous case: The saddle point problem formulated in the affine linear subspace: Find $v_h \in X_{h,div}^p$ s.t.

$$a(v_h, \varphi_h) = \tilde{l}(\varphi_h) \quad \forall \varphi_h \in X_{h,div}^{p,hom} \,.$$

With this constrained problem it is shown that the discrete velocity stays unchanged. Leaving the trial space unchanged and testing in the orthogonal complement induced by the bilinear form a,

$$X_{h,div}^{p,\perp} := \{\varphi_h \in X_h^p \mid \forall v_h \in X_{h,div}^p : a(v_h, \varphi_h) = 0\},$$

gives $p_h \in M_h^{p-1}$ s.t.

$$b(\varphi_h, p_h) = \tilde{l}(\varphi) \quad \forall \varphi_h \in X_{h,div}^{p,\perp} .$$

With this it is shown the balancing of a irrotational translation by the discrete pressure gradient. When $f \mapsto f + \nabla\psi$, let $(v_{\psi,h}, p_{\psi,h})$ denote the discrete solution to the transformed source term. For $\varphi_h \in X_h^p$:

$$\begin{aligned}
(f + \nabla\psi, \varphi_h)_{0,\Omega} &= (f, \varphi_h)_{0,\Omega} + \sum_{E \in \mathcal{E}_h} (\nabla\psi, \varphi_h)_{0,E} \\
&= (f, \varphi_h)_{0,\Omega} + b(\varphi_h, \psi) \qquad \text{using Remark 1 .} \\
\Rightarrow (f + \nabla\psi, \Pi_h^{\mathrm{RT}_h^{p-1}}(\varphi_h))_{0,\Omega} &= (f, \Pi_h^{\mathrm{RT}_h^{p-1}}(\varphi_h))_{0,\Omega} + b(\Pi_h^{\mathrm{RT}_h^{p-1}}(\varphi_h), \psi) \\
&= (f, \Pi_h^{\mathrm{RT}_h^{p-1}}(\varphi_h))_{0,\Omega} - \sum_{E \in \mathcal{E}_h} (\nabla \cdot \Pi_h^{\mathrm{RT}_h^{p-1}}(\varphi_h), \psi)_{0,E} \\
&= (f, \Pi_h^{\mathrm{RT}_h^{p-1}}(\varphi_h))_{0,\Omega} - \sum_{E \in \mathcal{E}_h} (\nabla \cdot \Pi_h^{\mathrm{RT}_h^{p-1}}(\varphi_h), \pi_h^{Q_h^{p-1}}(\psi))_{0,E} \\
&= (f, \Pi_h^{\mathrm{RT}_h^{p-1}}(\varphi_h))_{0,\Omega} + b(\varphi_h, \pi_h^{Q_h^{p-1}}(\psi))
\end{aligned}$$

where we have used that a reconstructed test function has zero normal component on the Dirichlet boundary and that the operator is divergence-preserving.

$$\Rightarrow \tilde{l}(\varphi_h) \mapsto \tilde{l}(\varphi_h) + b(\varphi_h, \pi_h^{Q_h^{p-1}}(\psi)) .$$

The constrained problem with the transformed source term:

$$a(v_{\psi,h}, \varphi_h) = \tilde{l}(\varphi_h) + b(\varphi_h, \pi_h^{Q_h^{p-1}}(\psi)) = \tilde{l}(\varphi_h) \quad \forall \varphi_h \in X_{h,div}^{p,hom}$$

$\Rightarrow v_{\psi,h} = v_h$. Secondly

$$\begin{aligned}
b(\varphi_h, p_{\psi,h}) &= \tilde{l}(\varphi_h) + b(\varphi_h, \pi_h^{Q_h^{p-1}}(\psi)) \\
\Leftrightarrow b(\varphi_h, p_{\psi,h} - \pi_h^{Q_h^{p-1}}(\psi)) &= \tilde{l}(\varphi_h) \quad \forall \varphi_h \in X_{h,div}^{p,\perp}
\end{aligned}$$

$\Rightarrow p_h + \pi_h^{Q_h^{p-1}}(\psi) = p_{\psi,h}$ by inf-sup stability. □

The projection operator A motivation for the Helmholtz-flux Raviart-Thomas projection comes from Lemma 2 which states the result of the pressure Poisson Raviart-Thomas reconstruction not to be pointwise divergence-free. An improvement to the correction

$v_h = w_h + G_h\psi_h$ is to replace w_h. Based on the discussion above, a straightforward choice is to take the image of the divergence-preserving operator. This would lead to a velocity field that is pointwise divergence-free. So the proposition for the Helmholtz-flux Raviart-Thomas reconstruction thus consists of combining the reconstruction operator for the velocity field with the accurate flux reconstruction from the pressure Poisson equation: $v_h = \Pi_h^{\mathrm{RT}_h^{p-1}}(w_h) + G_h\psi_h$.
In explicit: On element $E \in \mathcal{E}_h$ with faces $e \in \partial E$ compute

$$(v_h \cdot n_e, q_h)_{0,e} = (\{w_h\} \cdot n_e, q_h)_{0,e} + (-\{\nabla\psi_h\} \cdot n_e + \frac{\sigma}{h_e}[\psi_h], q_h)_{0,e} \qquad e \in \Gamma_h^{\mathrm{int}}, q \in Q_e^{p-1} , \tag{2.48}$$

$$(v_h \cdot n_e, q_h)_{0,e} = (g \cdot n_e, q_h)_{0,e} \qquad e \in \Gamma_h^D, q \in Q_e^{p-1} , \tag{2.49}$$

$$(v_h \cdot n_e, q_h)_{0,e} = (w_h \cdot n_e, q_h)_{0,e} + (-\nabla\psi_h \cdot n_e + \frac{\sigma}{h_e}\psi_h, q_h)_{0,e} \qquad e \in \Gamma_h^N, q \in Q_e^{p-1} , \tag{2.50}$$

and in addition for $p - 1 > 0$

$$\begin{aligned}(v_h, r_h)_{0,E} = (w_h, r_h)_{0,E} - (\nabla\psi_h, r)_{0,E} \\ + \frac{1}{2}\sum_{e\in\partial E\cap\Gamma_h^{\mathrm{int}}} (r \cdot n_e, [\psi_h])_{0,e} + \sum_{e\in\partial E\cap\Gamma_h^N} (r \cdot n_e, \psi_h)_{0,e} \quad \forall r \in \Psi_E^{p-1} .\end{aligned} \tag{2.51}$$

The projection step is then defined as:

Algorithm 2.4 The Helmholtz-flux RT projection $\mathcal{P}_h^{\mathrm{RT}}$ is given by the following algorithm:

i) For any tentative velocity $w_h \in X_h^p$ and fixed t solve

$$\psi_h \in M_h^{p-1} : \quad \alpha(\psi_h, q_h) = b(w_h, q_h) - r(q_h; t) \quad \forall q_h \in M_h^{p-1} .$$

ii) Reconstruct $\Pi_h^{\mathrm{RT}_h^{p-1}}(w_h) + G_h\psi_h \in \mathrm{RT}_h^{p-1}$.

iii) Set $\mathcal{P}_h^{\mathrm{RT}} w_h = v_h$ where v_h solves

$$(v_h, \varphi_h)_{0,\Omega} = \left(\Pi_h^{\mathrm{RT}_h^{p-1}}(w_h) + G_h\psi_h, \varphi_h\right)_{0,\Omega} \quad \forall\varphi_h \in X_h^p .$$

This requires the solution of a (block-) diagonal system.

We are now able to prove the following theorem given on affine hexahedral meshes.

Theorem 5. The operator $\mathcal{P}_h^{\mathrm{RT}}$ is a projection. The image $v_h = \mathcal{P}_h^{\mathrm{RT}} w_h$ is pointwise divergence-free and satisfies the discrete continuity equation exactly.

Proof. First consider the divergence of v_h. Owing to Theorem 2 and 3

$$\begin{aligned}(\nabla \cdot v_h, q_h)_{0,\Omega} &= \big(\nabla \cdot \Pi_h^{\mathrm{RT}_h^{p-1}}(w_h) + \nabla \cdot G_h\psi_h, q_h\big)_{0,\Omega} \\ &= -b(w_h, q_h) + r(q_h) + b(w_h, q_h) - r(q_h) \\ &= 0 \quad \forall q_h \in Q_h^{p-1} \,.\end{aligned}$$

Or equivalently in shorthand: $\nabla \cdot v_h = \nabla \cdot \Pi_h^{\mathrm{RT}_h^{p-1}}(w_h) + \nabla \cdot \gamma_h = B_h w_h - B_h w_h = 0$. Using this and owing to the calculation in Lemma 3 we can further conclude

$$\begin{aligned}b(v_h, q_h) &= b\big(\Pi_h^{\mathrm{RT}_h^{p-1}}(w_h), q_h\big) + b(G_h\psi_h, q_h) \\ &= \sum_{e\in\Gamma_h^D} (g \cdot n_e, q_h)_{0,e} \;\; + 0 \qquad\qquad\qquad = r(q_h) \quad \forall q_h \in Q_h^{p-1} \,.\end{aligned}$$

A second application of $\mathcal{P}_h^{\mathrm{RT}}$ produces at first a zero Helmholtz correction and leaves the input velocity unchanged since $\Pi_h^{\mathrm{RT}_h^{p-1}}$ is a projection. □

In contrast to the sole pressure Poisson flux reconstruction those conservation properties cannot be achieved by reconstructing in a Raviart-Thomas space of degree $k < p-1$. For lower degrees especially divergence-preservation is lost and moreover the image of $\Pi_h^{\mathrm{RT}_h^k}$ where the tentative velocity would be mapped to, is too small compared to the size of the velocity ansatz space X_h^p.

We conclude this section that both Raviart-Thomas reconstructions can be employed on non-affine hexahedral meshes by evaluating the Piola transform pointwise. If μ_E is not affine linear on a cell E, the local space RT_E^k given by (2.29) may not form a reasonable approximation to $H(\mathrm{div}; E)$ as constant functions need not be contained. Though desirable properties are still satisfied exactly on elements that are the image of $\hat{E}$ under an affine transformation.

2.2.3 Pressure-correction schemes

For ease of writing the DDN-term is omitted in the presentation of the splitting method.

Incremental Pressure Correction Scheme (IPCS)

The IPCS is a straightforward way to split between incompressibility and dynamics. In the viscous substep the pressure is made explicit that we denote by $p_h^{\bigstar,k+1}$. In the second substep a pressure correction is computed to accordingly correct the velocity. The particular choice of the time discretization is not important. It is possible to use the implicit Euler time stepping or second order time stepping methods such as BDF2 or Alexander's second order strongly S-stable scheme [Ale77]. The semi-discretized in space splitting scheme then reads as follows: Given v_h^k, p_h^k at time t^k, compute v_h^{k+1}, p_h^{k+1} at time $t^{k+1} = t^k + \Delta t^{k+1}$ as follows:

1. Choose explicit extrapolation of pressure $p_h^{\bigstar,k+1}$ at time t^{k+1} and compute tentative velocity $\tilde{v}_h^{k+1}$ by temporal evolution of

$$\rho(\partial_t v_h, \varphi_h)_{0,\Omega} + a(v_h, \varphi_h) + \rho c(v_h, \varphi) + b(\varphi_h, p_h^{\bigstar,k+1}) = l(\varphi_h; t) \quad \forall \varphi_h \in X_h^p \, .$$

2. Projection step: Solve the pressure Poisson equation to compute correction δp_h^{k+1} (c.f. step i) in Algorithm 2.2, 2.3, 2.4):

$$\alpha(\delta p_h^{k+1}, q_h) = \frac{1}{\Delta t^{k+1}}(b(\tilde{v}_h^{k+1}, q_h) - r(q_h; t^{k+1})) \quad \forall q_h \in M_h^{p-1},$$

where

$$\alpha(\psi_h, q_h) = b(\nabla \psi_h, q_h) + j_0(\psi_h, q_h).$$

Then compute $v_h^{k+1} = \mathcal{P}_h(\tilde{v}_h^{k+1})$, $\mathcal{P}_h = \{\mathcal{P}_h^{\text{div-div}}, \mathcal{P}_h^{\text{RT}}\}$, specified by step ii) in Algorithm 2.2, 2.3, 2.4 where $\psi_h = \Delta t^{k+1} \delta p_h^{k+1}$.

3. Pressure update:

$$p_h^{k+1} = \delta p_h^{k+1} + p_h^{\bigstar,k+1}.$$

In general a $(q-1)$-th order pressure extrapolation can be employed,

$$p_h^{\bigstar,k+1} = \begin{cases} 0 & \text{or} \\ p_h^k & \text{, for } q = 1 \\ 2p_h^k - p_h^{k-1} & \text{, for } q = 2 \\ 3p_h^k - 3p_h^{k-1} + p_h^{k-2} & \text{, for } q = 3 \end{cases}. \tag{2.52}$$

However, the pressure extrapolations for $q > 1$, are not self-starting and only valid for constant time step sizes.

Usage of constant extrapolation $p_h^{\bigstar,k+1} = p_h^k$ gives the IPCS. The IPCS introduces the artificial boundary conditions for the pressure correction which lead to the series of equalities

$$\partial_n p_h^{k+1}\big|_{\Gamma_D} = \ldots = \partial_n p_h^1\big|_{\Gamma_D} = \partial_n p_h^0\big|_{\Gamma_D} \tag{2.53}$$

$$p_h^{k+1}\big|_{\Gamma_N} = \ldots = p_h^1\big|_{\Gamma_N} = p_h^0\big|_{\Gamma_N} \tag{2.54}$$

for the pressure itself over time.

The choice $p_h^{\bigstar,k+1} = 0$, implicit Euler as time stepping reproduces Chorin's projection method where those boundary conditions even become homogeneous. Due to the neglect of the pressure in the momentum equation this scheme has an irreducible splitting error of order $\mathcal{O}(\Delta t)$. Hence using a higher-order time stepping scheme does not improve the overall accuracy. These artificial boundary conditions induce a numerical boundary layer that prevents the scheme to be fully first-order on the velocity in the H_0^1-norm and on the pressure in the L^2-norm. For purely Dirichlet boundary conditions it can be shown that the error in these two norms converges as $\mathcal{O}(\sqrt{\Delta t})$, (c.f. [Ran92]).

The IPCS is in the purely Dirichlet case, i.e. $\Gamma_N = \emptyset$, fully first-order accurate even if the implicit Euler time stepping is used. But when $\Gamma_N \neq \emptyset$ the order of approximation of the velocity in the H_0^1-norm and of the pressure in the L^2-norm is degraded due to the homogeneous Dirichlet boundary conditions for the pressure. There is little improvement regarding the order of the scheme when a second order time stepping method is used. In the purely Dirichlet case the scheme is fully second order on the velocity in the L^2-norm but it stays first order on the velocity in the H_0^1-norm and on the pressure in the L^2-norm. For $\Gamma_N \neq \emptyset$ the approximation order even stays the same.

The constant extrapolation for the explicit pressure in the momentum equation implies that the scheme has an irreducible splitting error of $\mathcal{O}(\Delta t^2)$. Hence using a higher than second order time discretization does not improve the overall accuracy.

Rotational Incremental Pressure Correction Scheme (RIPCS)

One reason for the above scheme to have poor convergence properties especially when outflow boundary conditions are present is that the pressure boundary conditions stay constant over time. To overcome this difficulty it was first introduced by Timmermans, Minev and Van De Vosse [TMVDV96] to use the *rotational form* of the Laplacian, namely

$$-\Delta v = \nabla \times (\nabla \times v) - \nabla(\nabla \cdot v). \tag{2.55}$$

To understand why this modification performs better we consider for simplicity the momentum equation in classical form and insert the rotational form of the Laplacian:

$$\frac{\tilde{v}_h^{k+1} - v_h^k}{\Delta t^{k+1}} + \mu \nabla \times (\nabla \times \tilde{v}_h^{k+1}) + \nabla(p_h^{\bigstar,k+1} - \mu \nabla \cdot \tilde{v}_h^{k+1}) = f(t^{k+1}) \tag{2.56}$$

where $p_h^{\bigstar,k+1}$ is as before an approximation of $p(\cdot, t^{k+1})$. Eliminating the tentative velocity $\tilde{v}_h^{k+1} = v_h^{k+1} + \Delta t^{k+1} \nabla \delta p_h^{k+1}$ with the Helmholtz decomposition gives

$$\frac{v_h^{k+1} - v_h^k}{\Delta t^{k+1}} + \mu \nabla \times (\nabla \times \tilde{v}_h^{k+1}) + \nabla(\delta p_h^{k+1} + p_h^{\bigstar,k+1} - \mu \nabla \cdot \tilde{v}_h^{k+1}) = f(t^{k+1})\,. \tag{2.57}$$

Thus the quantity $\delta p_h^{k+1} + p_h^{\bigstar,k+1} - \mu \nabla \cdot \tilde{v}_h^{k+1}$ can be interpreted as an approximation of the pressure. Hence retaining the time step with the momentum equation the tables can be turned to obtain the *incremental pressure-correction scheme in rotational form*: Given v_h^k, p_h^k at time t^k, compute v_h^{k+1}, p_h^{k+1} at time $t^{k+1} = t^k + \Delta t^{k+1}$ as follows:

1. Choose explicit extrapolation of pressure $p_h^{\bigstar,k+1}$ at time t^{k+1} and compute tentative velocity $\tilde{v}_h^{k+1}$ by temporal evolution of

$$\rho(\partial_t v_h, \varphi_h)_{0,\Omega} + a(v_h, \varphi_h) + \rho c(v_h, \varphi) + b(\varphi_h, p_h^{\bigstar,k+1}) = l(\varphi_h; t) \quad \forall \varphi_h \in X_h^p\,.$$

2. Projection step: Solve the pressure Poisson equation to compute correction δp_h^{k+1} (c.f. step i) in Algorithm 2.2, 2.3, 2.4):

$$\alpha(\delta p_h^{k+1}, q_h) = \frac{1}{\Delta t^{k+1}} (b(\tilde{v}_h^{k+1}, q_h) - r(q_h; t^{k+1})) \quad \forall q_h \in M_h^{p-1},$$

where

$$\alpha(\psi_h, q_h) = b(\nabla\psi_h, q_h) + j_0(\psi_h, q_h).$$

Then compute $v_h^{k+1} = \mathcal{P}_h(\tilde{v}_h^{k+1})$, $\mathcal{P}_h = \{\mathcal{P}_h^{\text{div-div}}, \mathcal{P}_h^{\text{RT}}\}$, specified by step ii) in Algorithm 2.2, 2.3, 2.4 where $\psi_h = \Delta t^{k+1}\delta p_h^{k+1}$.

3. Pressure update with scaling factor ω:

$$(p_h^{k+1}, q_h) = (\omega\delta p_h^{k+1} + p_h^{\bigstar,k+1}, q_h) + \mu(b(\tilde{v}_h^{k+1}, q_h) - r(q_h; t^{k+1})) \quad \forall q_h \in M_h^{p-1} .$$

The scaling factor is usually set to $\omega = 1$ for first order time stepping schemes and to $\omega = \frac{3}{2}$ for second order time stepping schemes.

The contribution $\nabla \cdot \tilde{v}_h^{k+1}$ improves the accuracy of the scheme such that it is first order accurate for both Dirichlet and outflow boundary conditions. The use of a second order time stepping scheme improves the convergence rate on the velocity in the H_0^1-norm and on the pressure in the L^2-norm to $\frac{3}{2}$ when $\Gamma_N = \emptyset$. In the presence of outflow boundary conditions the convergence rate $\frac{3}{2}$ for the velocity in the L^2-norm is likely to be the best possible whereas the convergence rate in the H_0^1-norm for the velocity and in the L^2-norm for the pressure is limited to 1. As in the IPCS higher than second order time stepping schemes do not improve the overall accuracy.

2.2.4 Consistent Splitting Schemes

In this section we review the fractional step techniques introduced in Guermond and Shen [GS03]. The schemes are based on a weak form of the pressure Poisson equation derived from the momentum equation. However, unlike the pressure-correction schemes, these splitting schemes deliver full accuracy on the velocity and pressure for first and second order temporal approximations in case of purely Dirichlet boundary conditions. For ease of writing we only consider the instationary Stokes equations. The nonlinear convection term does not affect the derivation. Moreover we assume in these schemes a constant decomposition of the time interval with time step size Δt.

The key idea in the derivation

The derivation starts by considering (2.10a) and testing with $\nabla_h q_h, q_h \in M_h^{p-1}$ therein:

$$b(\nabla_h q_h, p_h) = l(\nabla_h q_h, t) - \rho(\partial_t v_h, \nabla_h q_h) - a(v_h, \nabla_h q_h). \tag{2.58}$$

Using Observation 1 on the left hand side we get

$$b(\nabla_h q_h, p_h) = \sum_{E\in\mathcal{E}_h} (\nabla q_h, \nabla p_h)_{0,E} - \sum_{e\in\Gamma_h^{\text{int}}} (\{\nabla q_h\}\cdot n_e, [p_h])_{0,e} - \sum_{e\in\Gamma_h^N} (\nabla q_h \cdot n, p_h)_{0,e}, \tag{2.59}$$

but importantly compared to (2.22) the role of test and ansatz functions are interchanged. Thus interpreting (2.59) as a part of the SIPG formulation the last two terms already correspond to the symmetrization of the interior penalty terms whereas the

consistency terms are missing. We add the missing terms exploiting symmetry of the bilinear form $b+j_0$ in order to replace $b(\nabla_h q_h, p_h)$ by $\alpha(p_h, q_h)$ in (2.58) with the boundary conditions $\partial_n p_h = 0$ on Γ_D and $p_h = 0$ on Γ_N.

If v_h is explicitly given then the pressure can be computed from the momentum equation by solving a Poisson equation. Hence the fractional step technique for the consistent splitting schemes can be formulated in two consecutive steps: First compute the velocity by making the pressure explicit, then we update the pressure with the Poisson equation derived from the momentum equation.

Standard Splitting Scheme (SSS)

The main focus in this section lies in the derivation of the right-hand side for the pressure Poisson equation in the consistent splitting schemes. On the right side of (2.58), second order derivatives of the test functions q_h are hidden in the term $a(v_h, \nabla q_h)$ that one wants to get rid of. In [GS03] this elimination begins by taking the L^2-inner product of the momentum equation (2.1) for $\nabla q, q \in H^1(\Omega)$ and is based on the assumption $\dot{v} \mid_{\partial\Omega} = 0$ when integrating the term $(\rho\partial_t v, \nabla q)_{0,\Omega}$ by parts. It is easy however to add the missing term for non-constant boundary conditions:

$$(\rho\partial_t v, \nabla q)_{0,\Omega} = \int_{\partial\Omega} \rho\partial_t v \cdot q n \mathrm{d}s \ . \tag{2.60}$$

We will redo this derivation for discontinuous test functions $q_h \in M_h^{p-1}$. Hence let us go back to (2.58) and forget about (2.59) for the moment. Assuming that $\nabla \cdot \partial_t v = \partial_t(\nabla \cdot v) = 0$, the term $(\rho\partial_t v, \nabla_h q_h)_{0,\Omega}$ can be integrated by parts as follows:

$$\begin{aligned}
\rho(\partial_t v, \nabla_h q_h)_{0,\Omega} &= \sum_{E\in\mathcal{E}_h} (\rho\partial_t v, \nabla q_h)_{0,E} \\
&= \sum_{E\in\mathcal{E}_h} (\rho, \nabla \cdot (q_h \partial_t v))_{0,E} \\
&= \sum_{E\in\mathcal{E}_h} (\rho\partial_t v, q_h n)_{0,\partial E} \\
&= \sum_{e\in\Gamma_h^{\mathrm{int}}} (\{\rho\partial_t v\}, [q_h] n_e)_{0,e} + \int_{\partial\Omega} \rho\partial_t v \cdot q_h n \mathrm{d}s \ .
\end{aligned} \tag{2.61}$$

Now approximate $\rho(\partial_t v_h, \nabla q_h)_{0,\Omega}$ in (2.58) by the right side of (2.61). This gives the first intermediate result of the pressure Poisson equation subsequent to (2.58),

$$\begin{aligned}
b(\nabla_h q_h, p_h) =& l(\nabla_h q_h, t) - \sum_{e\in\Gamma_h^{\mathrm{int}}} \int_e \{\rho\partial_t v_h\} \cdot [q_h] n_e \mathrm{d}s - \int_{\partial\Omega} \rho\partial_t v_h \cdot q_h n \mathrm{d}s \\
&- a(v_h, \nabla_h q_h).
\end{aligned} \tag{2.62}$$

Now we test with $\nabla_h q_h$ in the viscous substep (with explicit extrapolation of the pressure),

$$\rho(\partial_t v_h, \nabla_h q_h) + a(v_h, \nabla_h q_h) + b(\nabla_h q_h, p_h^{\bigstar, k+1}) = l(\nabla_h q_h, t).$$

Then considering $b(\mathbb{\nabla}_h q_h, p_h^{k+1} - p_h^{\bigstar,k+1})$, the source term and $a(\cdot,\cdot)$ cancel out. Equation (2.62) thus becomes

$$b(\mathbb{\nabla}_h q_h, p_h^{k+1} - p_h^{\bigstar,k+1}) = \rho(\partial_t v_h, \mathbb{\nabla}_h q_h)_{0,\Omega} - \sum_{e\in\Gamma_h^{\text{int}}} \int_e \{\rho\partial_t v_h\}\cdot[q_h] n_e \mathrm{d}s - \int_{\partial\Omega} \rho\partial_t v_h \cdot q_h n \mathrm{d}s, \tag{2.63}$$

which is the second intermediate version of the pressure Poisson equation. As discussed in 2.2.4, on the left-hand side $b(\mathbb{\nabla}_h q_h, p_h^{k+1} - p_h^{\bigstar,k+1})$ is replaced by $\alpha(p_h^{k+1} - p_h^{\bigstar,k+1}, q_h)$. The remaining puzzle piece towards the final version of the pressure Poisson equation is the computation of $\partial_t v_h$ on the right-hand side. Here we use the q-th order backward difference formula (BDF) to approximate $\partial_t v\,|_{t=t^{k+1}}$ by $Dv_h^{k+1}/\Delta t$, in explicit

$$Dv_h^{k+1} = \begin{cases} v_h^{k+1} - v_h^k & , \text{for } q = 1 \\ \frac{3}{2}v_h^{k+1} - 2v_h^k + \frac{1}{2}v_h^{k-1} & , \text{for } q = 2 \end{cases}. \tag{2.64}$$

The final version of the right-hand side in the pressure Poisson equation is abbreviated by $\ell(q_h)$:

$$\begin{aligned}\ell(q_h) := \sum_{E\in\mathcal{E}} \left(\frac{\rho}{\Delta t} Dv_h^{k+1}, \mathbb{\nabla} q_h\right)_{0,E} - \sum_{e\in\Gamma_h\cap\partial\Omega} \left(\frac{\rho}{\Delta t} Dv_h^{k+1}, q_h n\right)_{0,e} \\ - \sum_{e\in\Gamma_h^{\text{int}}} \left(\frac{1}{\Delta t}\{\rho Dv_h^{k+1}\}, n_e[q_h]\right)_{0,e}.\end{aligned} \tag{2.65}$$

Now one step of the SSS consists of the following problems: Given v_h^k, p_h^k at time t^k, compute v_h^{k+1}, p_h^{k+1} at time $t^{k+1} = t^k + \Delta t$ as follows:

1. Choose explicit extrapolation of pressure $p_h^{\bigstar,k+1}$ at time t^{k+1} and compute velocity v_h^{k+1} by temporal evolution of

$$\rho(\partial_t v_h, \varphi_h) + a(v_h, \varphi_h) + b(\varphi_h, p_h^{\bigstar,k+1}) = l(\varphi_h, t) \quad \forall \varphi_h \in X_h^p.$$

2. Pressure Poisson equation: Compute correction δp_h^{k+1}

$$\alpha(\delta p_h^{k+1}, q_h) = \ell(q_h) \quad \forall q_h \in M_h^{p-1}.$$

3. Pressure update:

$$p_h^{k+1} = p_h^{\bigstar,k+1} + \delta p_h^{k+1}.$$

In the first order SSS, implicit Euler time stepping is used together with the pressure extrapolation (2.52) and BDF approximation (2.64) of order $q = 1$. In the second order formulation of the SSS an appropriate time stepping is used (most likely BDF2) together with the pressure extrapolation (2.52) and BDF approximation (2.64) of order $q = 2$.

Note that if $q \geq 2$, the scheme is not self-starting and then for $1 \leq k \leq q-1$ the $(k-1)$-th order formulation is used for initialization.

As in the IPCS the scheme introduces the artificial boundary conditions for the pressure correction which lead to the series of equalities

$$\partial_n p_h^{k+1}\big|_{\Gamma_D} = \ldots = \partial_n p_h^1\big|_{\Gamma_D} = \partial_n p_h^0\big|_{\Gamma_D} \tag{2.66}$$

$$p_h^{k+1}\big|_{\Gamma_N} = \ldots = p_h^1\big|_{\Gamma_N} = p_h^0\big|_{\Gamma_N} \tag{2.67}$$

for the pressure itself over time. Thus the SSS has as poor approximation properties as the IPCS especially when $\Gamma_N \neq \emptyset$.

Consistent Splitting Scheme (CSS)

Similarly to the pressure-correction scheme the accuracy of the above splitting scheme can be improved by using the rotational form of the Laplacian. Following the same argumentation as in 2.2.3 an additional divergence correction in the pressure update appears whereas the viscous substep and the pressure Poisson equation for the pressure correction remain unchanged.

Given v_h^k, p_h^k at time t^k, compute v_h^{k+1}, p_h^{k+1} at time $t^{k+1} = t^k + \Delta t$ as follows:

1. Choose explicit extrapolation of pressure $p_h^{\bigstar,k+1}$ at time t^{k+1} and compute velocity v_h^{k+1} by temporal evolution of

$$\rho(\partial_t v_h, \varphi_h) + a(v_h, \varphi_h) + b(\varphi_h, p_h^{\bigstar,k+1}) = l(\varphi_h, t) \quad \forall \varphi_h \in X_h^p.$$

2. Pressure Poisson equation: Compute correction δp_h^{k+1}

$$\alpha(\delta p_h^{k+1}, q_h) = \ell(q_h) \quad \forall q_h \in M_h^{p-1}.$$

3. Pressure update:

$$(p_h^{k+1}, q_h) = (\delta p_h^{k+1} + p_h^{\bigstar,k+1}, q_h) + \mu(b(v_h^{k+1}, q_h) - r(q_h; t^{k+1})) \quad \forall q_h \in M_h^{p-1}.$$

As in the SSS, the first order formulation uses implicit Euler time stepping together with the pressure extrapolation (2.52) and BDF approximation (2.64) of order $q = 1$. The second order formulation of the CSS uses an appropriate time stepping (most likely BDF2) together with the pressure extrapolation (2.52) and BDF approximation (2.64) of order $q = 2$. Note that if $q \geq 2$, the scheme is not self-starting and then for $1 \leq k \leq q-1$ the $(k-1)$-th order formulation is used for initialization.

Numerical experiments indicate that the pressure approximation is no longer plagued by an artificial boundary condition. Consequently the scheme is truly q-th order for $q = 1$ as has been proved by [GS03] for the purely Dirichlet case and as numerical evidences show for the case $\Gamma_N \neq \emptyset$. Numerical experiments show that the scheme is truly q-th order for $q = 2$ in the purely Dirichlet case, but a rigorous proof of this fact is still elusive.

2.2.5 Temporal discretization of the viscous substep

The discrete in space viscous substep in the IPCS, RIPCS, SSS and CSS is equivalent to solving a system of ordinary differential equations

$$M_h \frac{\mathrm{d}z_h(t)}{\mathrm{d}t} + \mathcal{L}_h(z_h(t), t) = 0 \tag{2.68}$$

for the unknown coefficients $z_h(t)$ in the expansion $v_h(t) = \sum_{j=1}^{\dim(X_h^p)} z_h(t)_j \varphi_j$. The mass matrix and the nonlinear spatial residual vector have the entries

$$\begin{aligned}(M_h)_{ij} &= \rho(\varphi_j, \varphi_i)_{0,\Omega} \\ (\mathcal{L}_h(z_h(t),t))_i &= \sum_j a(\varphi_j, \varphi_i) z_h(t)_j + \rho c\big(\sum_j z_h(t)_j \varphi_j, \varphi_i\big) + b(\varphi_i, p_h^{\bigstar}(t)) \\ &\quad - \rho\beta s_o\big(\sum_j z_h(t)_j \varphi_j, \varphi_i\big) - l(\varphi_i; t).\end{aligned}$$

For the in-time evolution of (2.68) several classes of time stepping methods can be used.

Diagonal implicit Runge-Kutta methods The temporal evolution with the diagonal implicit Runge-Kutta methods is done in the *Shu-Osher representation*, [SO88, GKS11]. One time step in an s-stage scheme includes the following computations:

1. Set $z_h^{(0)} = z_h^k$.
2. For $i = 1, \ldots, s$ compute $z_h^{(i)}$ given by

$$\sum_{j=0}^{i} \Big[\frac{\alpha_{ij}}{\Delta t^{k+1}} M_h z_h^{(j)} + \beta_{ij} \mathcal{L}_h(z_h^{(j)}, t^k + \delta_j \Delta t^{k+1}) \Big] = 0. \tag{2.69}$$

3. Set $z_h^{k+1} = z_h^{(s)}$.

The rectangular matrices $\alpha, \beta \in \mathbb{R}^{s \times s+1}$ are lower Hessenberg and $\delta \in \mathbb{R}^{s+1}$. The representation through the Butcher tableaus becomes unique by setting $\alpha_{ii} = 1$. Note that diagonal implicit Runge-Kutta methods in the standard textbook formulation can be converted to the Shu-Osher representation if the last row in the $s \times s$ Butcher tableau is equal to the row vector of coefficients in the final linear combination of the vectors k_i. Though every explicit Runge-Kutta method can be transformed back and forth.

At each stage a possibly nonlinear system $R(z_h^{(i)}) = 0$ has to be solved where $R(z_h^{(i)})$ is defined by the left-hand side of (2.69). For the solution Newton's method is employed and according to this the residual $R(z_h^{(i)})$ is decomposed into a constant part with respect to the unknown $z_h^{(i)}$ and the non-constant part $\big(\Delta t^{k+1}\big)^{-1} \alpha_{ii} M_h z_h^{(i)} + \beta_{ii} \mathcal{L}_h(z_h^{(i)}, t^k + \delta_i \Delta t^{k+1})$. Several Butcher tableaus of diagonal implicit Runge-Kutta methods are listed in Appendix A.5.

Linear multistep methods In the linear m-step procedure z_h^{k+1} is computed as

$$\sum_{j=0}^{m} \frac{\alpha_{m-j}}{\Delta t} M_h z_h^{k+1-j} + \sum_{j=0}^{m} \beta_{m-j} \mathcal{L}_h(z_h^{k+1-j}, t^{k+1-j}) = 0 \,. \tag{2.70}$$

Here $\alpha \in \mathbb{R}^{m+1}, \beta \in \mathbb{R}^{m+1}$ and this system looks formally the same as one of the Shu-Osher stages, (2.69). A difference is that the coefficient vectors $z_h^{k-1}, \ldots, z_h^{k+1-m}$ of the previous time steps need to be explicitly remembered. The normalization $\alpha_m = 1$ is used such that the representation becomes unique. As above the residual $R(z_h^{k+1})$ is defined by the left-hand side of (2.70) and is decomposed into a constant and non-constant part with respect to the unknown z_h^{k+1}. Linear multistep methods are most easily applied for constant time step sizes. Second order temporal derivatives can also be discretized without decoupling into a first order system, an example is the the Newmark β-scheme [Zie77]. Butcher tableaus of several linear implicit multistep methods can be found in Appendix A.6.

IMEX Runge-Kutta methods In Implicit-Explicit (IMEX) Runge-Kutta methods the spatial residual vector is split into $\mathcal{L}_h = \mathcal{L}_{I,h} + \mathcal{L}_{E,h}$. The subscript I denotes the implicitly treated part in the time stepping and the subscript E denotes the explicitly treated part. A detailed description of such methods can be found in [ARS97]. Recall that the idea behind these methods is: If $\mathcal{L}_{E,h}$ is treated explicit the resulting systems should be easier to solve or if $\mathcal{L}_{I,h}$ is treated implicit the choice of the time step size should not be too restrictive. Similar to the diagonal implicit Runge-Kutta methods, IMEX methods can be brought to the Shu-Osher form. One time step in an s-stage scheme includes the following computations:

1. Set $z_h^{(0)} = z_h^k$.
2. For $i = 1, \ldots, s$ compute $z_h^{(i)}$ given by

$$\begin{aligned} \sum_{j=0}^{i} \frac{\alpha_{ij}}{\Delta t^{k+1}} M_h z_h^{(j)} &+ \sum_{j=0}^{i-1} (\beta_E)_{ij} \, \mathcal{L}_{E,h}(z_h^{(j)}, t^k + (\delta_E)_j \, \Delta t^{k+1}) \\ &+ \sum_{j=0}^{i} (\beta_I)_{ij} \, \mathcal{L}_{I,h}(z_h^{(j)}, t^k + (\delta_I)_j \, \Delta t^{k+1}) = 0 \,. \end{aligned} \tag{2.71}$$

3. Set $z_h^{k+1} = z_h^{(s)}$.

All the matrices are rectangular, $\alpha, \beta_I, \beta_E \in \mathbb{R}^{s \times s+1}$, the first two are lower Hessenberg and the latter is lower triangular. The vectors δ_I, δ_E for the intermediate times are in $\mathbb{R}^{s+1}$. Again we set $\alpha_{ii} = 1$ to obtain a unique representation. In particular for the viscous substep the diffusion is treated implicit and the nonlinear convection term is treated explicit:

$$(\, \mathcal{L}_{I,h}(z_h(t), t) \,)_i = \sum_j a(\varphi_j, \varphi_i) z_h(t)_j + b(\varphi_i, p_h^{\bigstar}(t)) - l(\varphi_i; t) \tag{2.72}$$

$$(\,\mathcal{L}_{E,h}(z_h(t),t)\,)_i = \rho c(\sum_j z_h(t)_j\varphi_j,\varphi_i) - \rho\beta s_o(\sum_j z_h(t)_j\varphi_j,\varphi_i)\;. \tag{2.73}$$

The system (2.71) is then linear and corresponds to a Helmholtz equation for the unknown $z_h^{(i)}$ at each stage. We list several Butcher tableaus of IMEX Runge-Kutta methods in Appendix A.7.

2.3 Numerical experiments

The numerical experiments start by cross-comparing the pointwise divergence and local mass conservation for the div-div projection and the pressure Poisson $H(\mathrm{div})$ reconstruction (summarized in Algorithm 2.3). Then the convergence properties of the IPCS, RIPCS, SSS and CSS for global Dirichlet boundary conditions 2.3.2 and mixed boundary conditions 2.3.3 are illustrated. For periodic boundary conditions 2.3.4 and in 3D using the Beltrami flow problem 2.3.5 we restrict ourselves to the IPCS and RIPCS. All schemes are tested in their second order formulation. Temporal convergence is analyzed for the Taylor-Hood-like DG-spaces $\mathcal{Q}_2/\mathcal{Q}_1$, $\mathcal{Q}_3/\mathcal{Q}_2$, $\mathcal{Q}_4/\mathcal{Q}_3$ and also local mass conservation - given as the left-hand side of (2.13) - is investigated.

2.3.1 Local mass conservation

We consider the Navier-Stokes equations on the domain $\Omega = (-1,1)^2$ and take the two dimensional Taylor-Green vortex which has been studied before by [TG37, Cho68, RB06]. In two dimensions the Taylor-Green vortex possesses the exact solution

$$\begin{aligned} v_1(x,y,t) &= -e^{-2\,\pi^2\,\nu\,t}\,\cos(\pi\,x)\,\sin(\pi\,y)\\ v_2(x,y,t) &= e^{-2\,\pi^2\,\nu\,t}\,\sin(\pi\,x)\,\cos(\pi\,y)\\ p(x,y,t) &= -0.25\,\rho\,e^{-4\,\pi^2\,\nu\,t}\,(\cos(2\,\pi\,y)+\cos(2\,\pi\,x))\;. \end{aligned} \tag{2.74}$$

The source term is given by $f = 0$. We set $\rho = 1$, $\mu = 1/100$ and $\nu = \mu/\rho$. Periodic boundary conditions are imposed in both the x and y directions. We do the computations on a 160×160 rectangular mesh. The discussion on the temporal convergence rates is postponed to Section 2.3.4.

We start the discussion on the choice of order in the Raviart-Thomas space for pressure Poisson $H(\mathrm{div})$ reconstruction. We have shown in Theorem 2 that for RT_h^{p-1} it holds: (I) $(\mathcal{P}_h^{\mathrm{RT}})^2 = \mathcal{P}_h^{\mathrm{RT}}$, (II) the reconstructed velocity satisfies the continuity equation and (III) is locally mass conservative. However a naive approach by looking at the dimension of the local function space of $\nabla_h M_h^{p-1}$ also accounts to possibly choose RT_h^{p-2}. As stated in Remark 3 local mass conservation can still be achieved with reconstruction in Raviart-Thomas space of degree $p-2$. This is demonstrated on the right of figure 2.2 and notably we get the same distribution with RT_h^{p-1}. Moreover numerical experiments with the power iteration applied to the operator $\mathcal{P}_h^{\mathrm{RT}}$ have shown that $(\mathcal{P}_h^{\mathrm{RT}})^2\tilde{v}_h^{k+1} = \mathcal{P}_h^{\mathrm{RT}}\tilde{v}_h^{k+1}$ also for RT_h^{p-2}. Table 2.2 - 2.3 compare the temporal accuracy between the discretizations

$\mathcal{Q}_2/\mathcal{Q}_1$ with reconstruction in RT_h^0 or RT_h^1. It can be seen that there is no significant difference on the error at final time. Reconstruction in the RT_h^{p-2} space provides thus to be a sufficient alternative in the splitting algorithm.

Next we want to cross-compare the temporal accuracy for the spatial discretizations $\mathcal{Q}_2/\mathcal{Q}_1$, $\mathcal{Q}_3/\mathcal{Q}_2$ with the div-div projection and $\mathcal{Q}_2/\mathcal{Q}_1$ with reconstruction in RT_h^0, $\mathcal{Q}_3/\mathcal{Q}_2$ with reconstruction in RT_h^1. Table 2.1 - 2.2 show the errors for the RIPCS $\mathcal{Q}_2/\mathcal{Q}_1$ with div-div projection and the RIPCS $\mathcal{Q}_2/\mathcal{Q}_1$ with reconstruction in RT_h^0 and table 2.4 - 2.5 the errors for the RIPCS $\mathcal{Q}_3/\mathcal{Q}_2$ with div-div projection and the RIPCS $\mathcal{Q}_3/\mathcal{Q}_2$ with reconstruction in RT_h^1, respectively. There is no significant difference in the temporal behavior for both pairs, a logarithmic plot of the errors would lead to indistinguishable curves. Thus for the upcoming investigation on the convergence properties we will use the div-div projection technique because it is an inexpensive alternative to the pressure Poisson $H(\mathrm{div})$ reconstruction which was at the time only implemented up to order one. Note that the errors in the tables 2.1, 2.4 are also contained in the figures of 2.7.

Table 2.1 Errors for the Taylor-Green vortex at final time T=2 obtained by RIPCS and $\mathcal{Q}_2/\mathcal{Q}_1$ with div-div projection.

dt	L^2 error v	H_0^1 error v	L^2 error p
2.000e-01	3.89336e-02	3.12769e-01	2.24301e-02
1.000e-01	1.00536e-02	7.86088e-02	6.49292e-03
5.000e-02	2.54833e-03	1.96963e-02	2.07180e-03
2.500e-02	6.40802e-04	5.00184e-03	7.85322e-04
1.250e-02	1.60275e-04	1.93785e-03	3.74214e-04
6.250e-03	3.99586e-05	1.67405e-03	2.24884e-04

Table 2.2 Errors for the Taylor-Green vortex at final time T=2 obtained by RIPCS and $\mathcal{Q}_2/\mathcal{Q}_1$ with pressure Poisson reconstruction in RT_h^0.

dt	L^2 error v	H_0^1 error v	L^2 error p
2.000e-01	3.89026e-02	3.06282e-01	2.24215e-02
1.000e-01	1.00444e-02	7.72283e-02	6.49243e-03
5.000e-02	2.54548e-03	1.96643e-02	2.07260e-03
2.500e-02	6.39597e-04	5.44801e-03	7.86167e-04
1.250e-02	1.59573e-04	2.63178e-03	3.75384e-04
6.250e-03	3.97373e-05	2.33037e-03	2.27288e-04

In figure 2.1 the pointwise divergence for $p = 2$ on each mesh element is presented. The element-local div-div projection leads to smaller pointwise divergence than obtained with the pressure Poisson $H(\mathrm{div})$ reconstruction. But it does not really cure the error on the local mass conservation. Compared to the standard L^2-projection the div-div projection reduces the values of the pointwise divergence and local mass conservation. The magnitude of the pointwise divergence from the pressure Poisson $H(\mathrm{div})$ reconstruction is in between the magnitudes from the standard L^2-projection and the stabilized variant,

it is not identically zero as predicted by Lemma 2. The distribution of the divergence error with $\mathcal{Q}_2/\mathcal{Q}_1$ and reconstruction in RT_h^1 is similar and has the same maximum.

Figure 2.2 shows the error on local mass conservation for $p = 2$. According to our discussion at the beginning of 2.3.1 this appealing conservation property is perfectly fulfilled for the RT_h^{p-1} and RT_h^{p-2} reconstructions of the Helmholtz correction.

Table 2.3 Errors for the Taylor-Green vortex at final time T=2 obtained by RIPCS and $\mathcal{Q}_2/\mathcal{Q}_1$ with pressure Poisson reconstruction in RT_h^1.

dt	L^2 error v	H_0^1 error v	L^2 error p
2.000e-01	3.88824e-02	3.03453e-01	2.24163e-02
1.000e-01	1.00387e-02	7.71449e-02	6.49254e-03
5.000e-02	2.54417e-03	1.96504e-02	2.07356e-03
2.500e-02	6.39432e-04	5.44560e-03	7.86989e-04
1.250e-02	1.59699e-04	2.63182e-03	3.75972e-04
6.250e-03	3.99134e-05	2.33056e-03	2.27698e-04

Table 2.4 Errors for the Taylor-Green vortex at final time T=2 obtained by RIPCS and $\mathcal{Q}_3/\mathcal{Q}_2$ with div-div projection.

dt	L^2 error v	H_0^1 error v	L^2 error p
2.000e-01	3.88929e-02	3.03675e-01	2.23165e-02
1.000e-01	1.00411e-02	7.70569e-02	6.38400e-03
5.000e-02	2.54558e-03	1.94734e-02	1.96199e-03
2.500e-02	6.40610e-04	4.89254e-03	6.72911e-04
1.250e-02	1.60670e-04	1.22611e-03	2.60387e-04
6.250e-03	4.02320e-05	3.06909e-04	1.11552e-04

Table 2.5 Errors for the Taylor-Green vortex at final time T=2 obtained by RIPCS and $\mathcal{Q}_3/\mathcal{Q}_2$ with pressure Poisson reconstruction in RT_h^1.

dt	L^2 error v	H_0^1 error v	L^2 error p
2.000e-01	3.88969e-02	3.03581e-01	2.23175e-02
1.000e-01	1.00423e-02	7.70650e-02	6.38402e-03
5.000e-02	2.54588e-03	1.94755e-02	1.96185e-03
2.500e-02	6.40684e-04	4.89306e-03	6.72797e-04
1.250e-02	1.60689e-04	1.22624e-03	2.60320e-04
6.250e-03	4.02364e-05	3.06934e-04	1.11516e-04

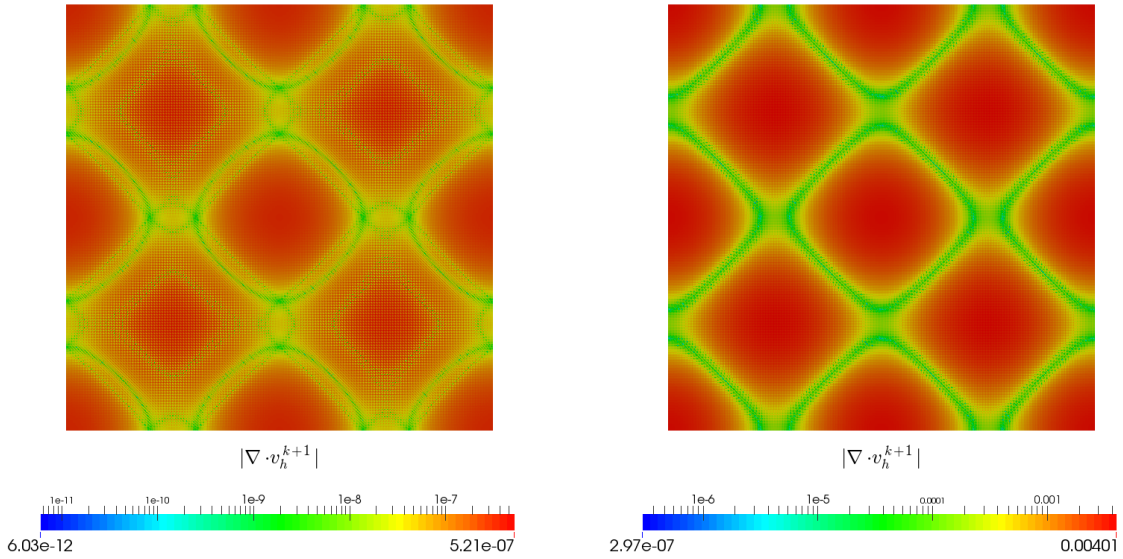


Figure 2.1: Pointwise divergence of the Taylor-Green vortex solution at time 1 with $\Delta t = 0.025$ obtained by the RIPCS.
Left part shows $\mathcal{Q}_2/\mathcal{Q}_1$ with div-div projection. Right part shows $\mathcal{Q}_2/\mathcal{Q}_1$ with pressure Poisson reconstruction in RT_h^0.

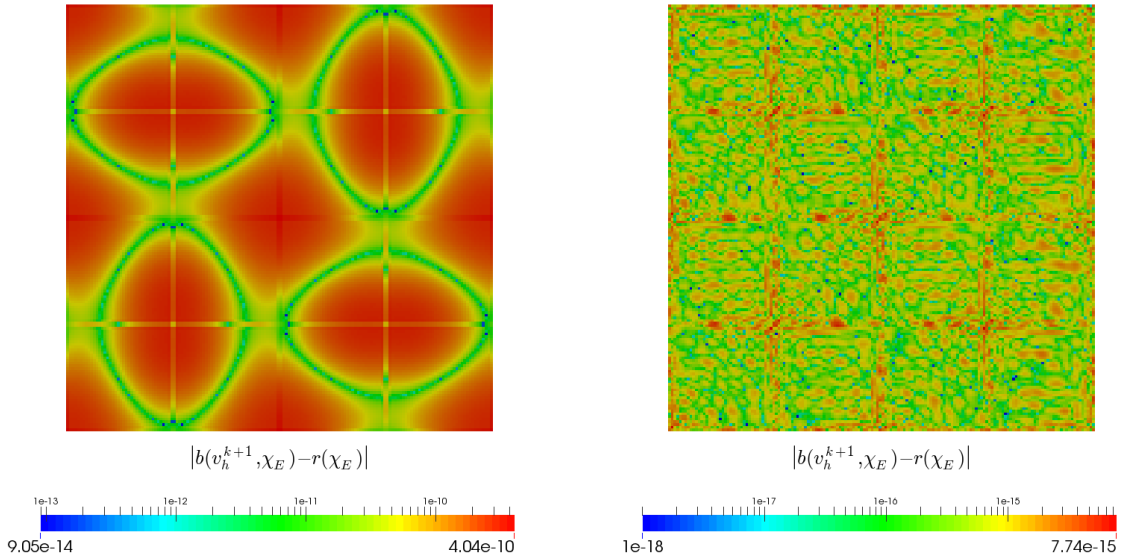


Figure 2.2: Local mass conservation of the Taylor-Green vortex solution at time 1 with $\Delta t = 0.025$ obtained by the RIPCS.
Left part shows $\mathcal{Q}_2/\mathcal{Q}_1$ with div-div projection. Right part shows $\mathcal{Q}_2/\mathcal{Q}_1$ with pressure Poisson reconstruction RT_h^0, identical to reconstruction in RT_h^1.

2.3.2 Global Dirichlet boundary conditions

We consider the Stokes equations on the domain $\Omega = (0,1)^2$ and take the exact solution to be

$$\begin{aligned} v_1(x,y,t) &= \sin(x+t)\,\sin(y+t) \\ v_2(x,y,t) &= \cos(x+t)\,\cos(y+t) \\ p(x,y,t) &= \sin(x-y+t)\ . \end{aligned} \tag{2.75}$$

The source term is given by $f = \partial_t v - \Delta v + \nabla p$. The density and viscosity are both set to $\rho = \mu = 1$. Computations were done on a 160×160 rectangular mesh.

Figure 2.3 shows the error and the convergence rates as function of Δt for the IPCS and RIPCS. The green curves show the L^2-error for the velocity, the red curves the H_0^1-error for the velocity and the blue curves the L^2-error for the pressure obtained by the polynomial degrees $p = 2, 3, 4$. The curves grouped by the same color are almost identical meaning that the splitting error is dominant in the measured range of Δt. Therefore we have left out the curves with $p = 4$ on the right for the sake of clarity. A transition towards smaller time steps causes earlier flattening out of the error curves the lower the spatial order is. This emerges at first for the H_0^1-error for the velocity and L^2-error for the pressure. This is demonstrated for the Taylor-Green vortex solution in section 2.3.4, c.f. right of figure 2.7.

Theory states that the solution of the second order IPCS satisfies the following error estimates: (I) L^2-velocity: $\mathcal{O}(\Delta t^2)$ (II) H_0^1-velocity, L^2-pressure: $\mathcal{O}(\Delta t)$. On the left of figure 2.3 it is observed that the velocity error in the L^2-norm is second order accurate, in the other two error measures the rate is 1.5 which is better than the prediction. Now the solution of the RIPCS satisfies the following error estimates: (I) L^2-velocity: $\mathcal{O}(\Delta t^2)$ (II) H_0^1-velocity, L^2-pressure: $\mathcal{O}(\Delta t^{\frac{3}{2}})$. The convergence rates on the right of figure 2.3 are consistent with the error estimates. Note that the L^2-errors on the velocity and pressure are almost identical to the results presented in [GS04]. The reason for the slight difference is likely to be the usage of BDF2 in [GS04] as time stepping.

A consideration of local mass conservation shows that it is well satisfied in the interior of the domain for the div-div projection. However the largest values $\sim 10^{-9}$ are located in the cells that share an edge with the boundary. This is due to the artificial boundary conditions on the pressure.

The situation is different for the pressure Poisson $H(\mathrm{div})$ reconstruction. In that case the distribution is similar to the right in figure 2.2 with $\max_E |b(v_h^{k+1}, \chi_E) - r(\chi_E)| \sim 5 \cdot 10^{-14}$.

The SSS and CSS are only considered for the polynomial degree $p = 2$. The BDF2 method is used for both schemes to march in time in order to compare with the results from [GS04]. Figure 2.4 shows the error and convergence rates as a function of Δt.

It can be shown that the second order SSS has similar convergence properties as the second order IPCS: (I) L^2-velocity: $\mathcal{O}(\Delta t^2)$ (II) H_0^1-velocity, L^2-pressure: $\mathcal{O}(\Delta t)$. The left of figure 2.4 shows that the error is second order accurate on the velocity in the L^2-norm. However the convergence rates in the H_0^1-norm and in the L^2-norm for the pressure behave rather like $\mathcal{O}(\Delta t^{\frac{3}{2}})$. These results are in agreement with the error plot

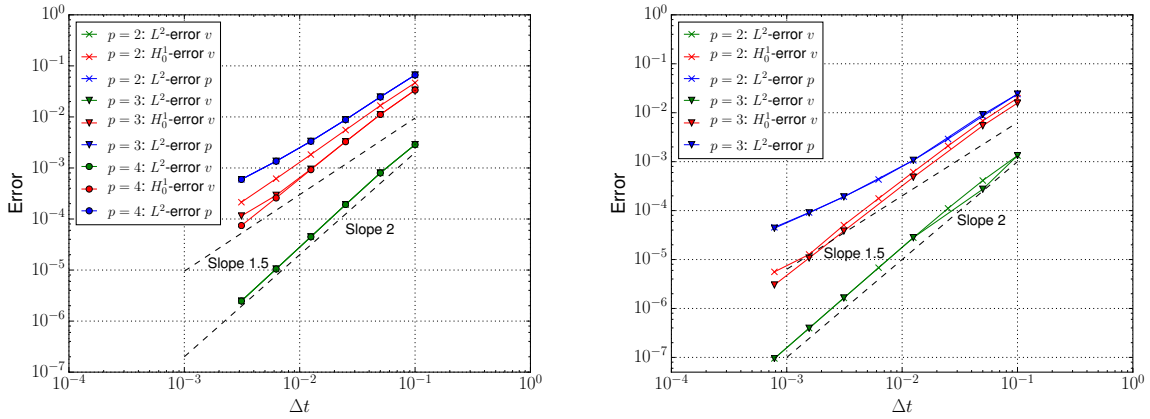


Figure 2.3: Errors and convergence rates at final time T=1 for the global Dirichlet problem and spatial discretizations $\mathcal{Q}_2/\mathcal{Q}_1$, $\mathcal{Q}_3/\mathcal{Q}_2$, $\mathcal{Q}_4/\mathcal{Q}_3$. Left part shows the IPCS. Right part shows the RIPCS.

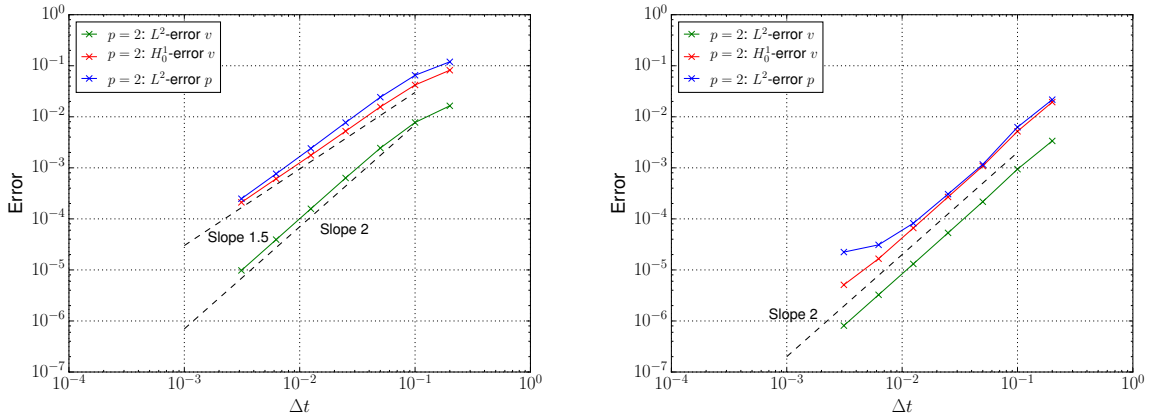


Figure 2.4: Errors and convergence rates at final time T=1 for the global Dirichlet problem and spatial discretization $\mathcal{Q}_2/\mathcal{Q}_1$. Left part shows the SSS. Right part shows the CSS.

presented in [GS03]. As for the SSS there is no proven error estimate for the second order CSS. But the right part of figure 2.4 indicates that the scheme is truly second order accurate. Note that the error in the L^2-norm for the pressure is saturated by the spatial discretization error for smaller time steps. The results lead to an identical error plot as presented in [GS03].

2.3.3 Mixed boundary conditions

We consider again the Stokes equations on the domain $\Omega = (0,1)^2$ and take the exact solution to be

$$\begin{aligned} v_1(x,y,t) &= \sin x \, \sin(y+t) \\ v_2(x,y,t) &= \cos x \, \cos(y+t) \\ p(x,y,t) &= \cos x \, \sin(y+t) \ . \end{aligned} \tag{2.76}$$

The source term is again given by $f = \partial_t v - \Delta v + \nabla p$. The density and viscosity are both set to $\rho = \mu = 1$. The outflow boundary is located at $\Gamma_N = \{(x,y) \in \partial\Omega \mid x = 0\}$. Computations were done on 160×160 rectangular mesh.

Figure 2.5 shows the error and the convergence rates as function of Δt for the IPCS and RIPCS. The green curves show the L^2-error for the velocity, the red curves the H_0^1-error for the velocity and the blue curves the L^2-error for the pressure obtained by the polynomial degrees $p = 2, 3, 4$. The curves grouped by the same color are almost identical meaning that the splitting error is dominant in the measured range of Δt. Therefore we have left out the curves with $p = 3$ on the left and $p = 4$ on the right for the sake of clarity. A transition towards smaller time steps causes earlier flattening out of the error curves the lower the spatial order is. This emerges at first for the H_0^1-error for the velocity and L^2-error for the pressure. This is demonstrated for the Taylor-Green vortex solution in section 2.3.4, c.f. right of figure 2.7.

The solution of the IPCS satisfies the following error estimates: (I) L^2-velocity: $\mathcal{O}(\Delta t)$ (II) H_0^1-velocity, L^2-pressure: $\mathcal{O}(\Delta t^{\frac{1}{2}})$ which are identical to the first order IPCS. The results on the left of figure 2.5 indeed show that the pressure approximation is poor due to the homogeneous Dirichlet boundary condition imposed on Γ_N. The RIPCS delivers improved error estimates in presence of mixed boundary conditions: (I) L^2-velocity: $\mathcal{O}(\Delta t^{\frac{3}{2}})$ (II) H_0^1-velocity, L^2-pressure: $\mathcal{O}(\Delta t)$. The convergence rates on right of figure 2.5 are consistent with those estimates. Furthermore the error on the velocity in the L^2-norm behaves like $\mathcal{O}(\Delta t^{\frac{5}{3}})$ which is also observed in Guermond, Minev and Shen [GMS06, GMS05]. The error in the H_0^1-norm is close to $\mathcal{O}(\Delta t^{\frac{5}{4}})$ which is higher than the rate $\mathcal{O}(\Delta t)$ predicted by theory. Note that [GMS06, GMS05] have used BDF2 as time stepping for this problem and therefore the error curves are almost identical.

Another consideration of local mass conservation shows that it is well satisfied in the interior of the domain. But due to the homogeneous Dirichlet boundary conditions for the pressure imposed on Γ_N, the largest errors $\sim 10^{-7}$ are located in the cells next to outflow boundary.

With the pressure Poisson $H(\mathrm{div})$ reconstruction we have $\max_E |b(v_h^{k+1}, \chi_E) - r(\chi_E)| \sim 5 \cdot 10^{-14}$ whereat the maximum also occurs in the boundary cells.

As in the previous section the SSS and CSS are briefly discussed. Figure 2.6 shows the error and convergence rates as function of Δt for the SSS.

The second order SSS has the same convergence properties as the IPCS: (I) L^2-velocity: $\mathcal{O}(\Delta t)$ (II) H_0^1-velocity, L^2-pressure: $\mathcal{O}(\Delta t^{\frac{1}{2}})$. Figure 2.6 confirms those rates and shows that pressure approximation is poor due to the homogeneous Dirichlet boundary condition imposed on Γ_N. There is no proven error estimate for the CSS in case of $\Gamma_N \neq \emptyset$

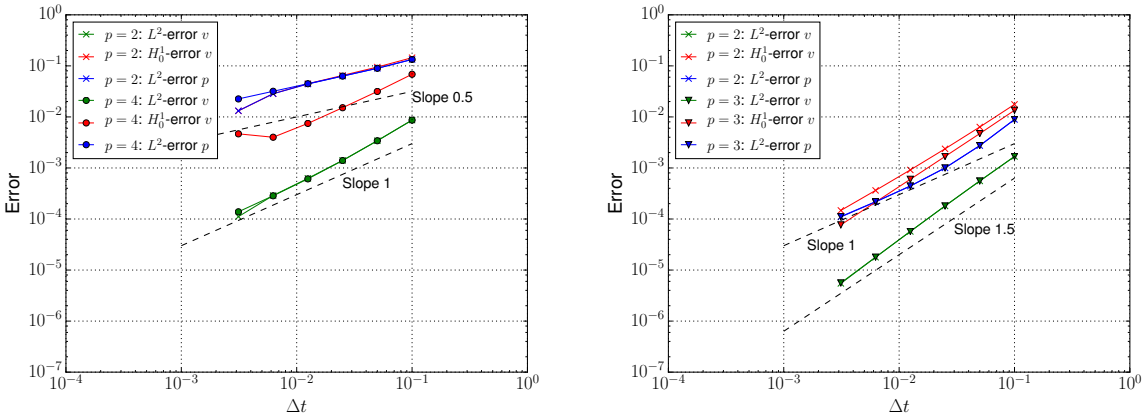


Figure 2.5: Errors and convergence rates at final time T=1 for the mixed boundary condition problem and spatial discretizations $\mathcal{Q}_2/\mathcal{Q}_1$, $\mathcal{Q}_3/\mathcal{Q}_2$, $\mathcal{Q}_4/\mathcal{Q}_3$. Left part shows the IPCS. Right part shows the RIPCS.

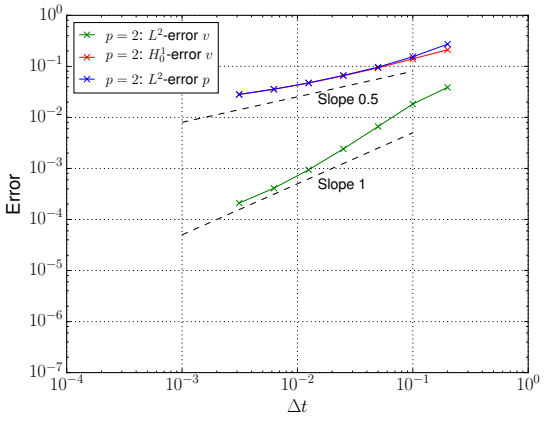


Figure 2.6: Errors and convergence rates at final time T=1 for the mixed boundary condition problem using the SSS and spatial discretization $\mathcal{Q}_2/\mathcal{Q}_1$.

at the time of writing. Numerical evidences show that the first order formulation is truly first order accurate which we can confirm. However the second order CSS becomes unstable if the time step goes below a certain threshold $\Delta t \sim h^2$. This behavior has also been reported by [GMS06].

2.3.4 Periodic boundary conditions

The following problems have only been calculated with the pressure-correction schemes.

We continue with the configuration and test problem presented in 2.3.1. Figure 2.7 shows the error and convergence rates as a function of Δt for the IPCS and RIPCS. The green curves show the L^2-error for the velocity, the red curves the H_0^1-error for the velocity and the blue curves the L^2-error for the pressure obtained by the polynomial

degrees $p = 2, 3$. The results for $p = 4$ are almost identical to $p = 3$, therefore it has been omitted for the sake of clarity. For the IPCS the curves grouped by the same color are almost identical meaning that the splitting error is dominant in the measured range of Δt. Note however that for $p = 2$ in the RIPCS the spatial error is already not negligible in this range and becomes all-dominant for additionally smaller time steps taken. It can be seen on the right that the H_0^1-error on the velocity and L^2-error on the pressure flattens out whereas the errors from spatial order three continue decreasing with the same rate. That puts in favor higher polynomial degrees since the error on the same spatial mesh for moderate time step sizes is minimized.

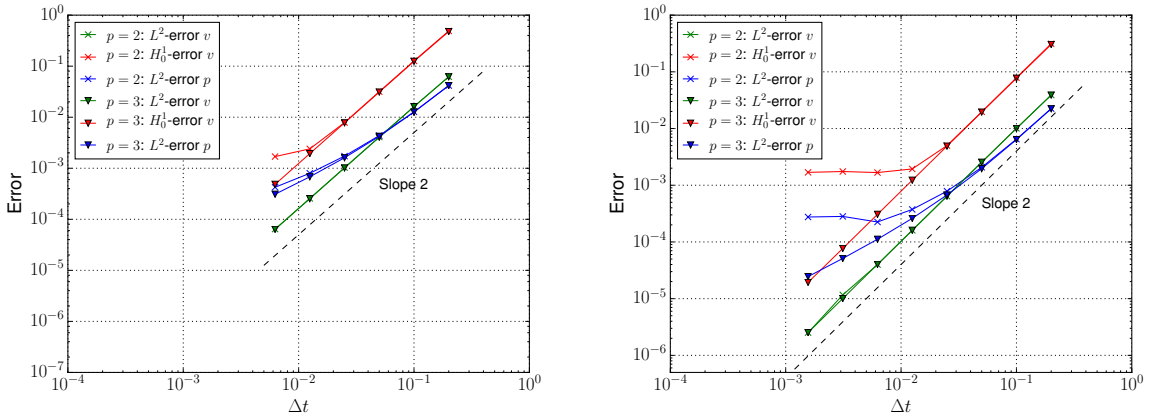


Figure 2.7: Errors and convergence rates at final time T=2 for the periodic boundary condition problem and spatial discretizations $\mathcal{Q}_2/\mathcal{Q}_1$, $\mathcal{Q}_3/\mathcal{Q}_2$, $\mathcal{Q}_4/\mathcal{Q}_3$. Left part shows the IPCS. Right part shows the RIPCS.

There is no rigorous error analysis of the projection methods for purely periodic boundary conditions. But since in the periodic case no artificial boundary conditions are imposed on the pressure, both the standard and rotational formulation are expected to be fully second order accurate. This is validated for the pressure-correction schemes in figure 2.7. The error of the RIPCS is slightly lower than the error of the IPCS, but both schemes have the same convergence rate. It is close to $\mathcal{O}(\Delta t^2)$ in the L^2-norm on the pressure while the rates of the velocity in the L^2-norm and H_0^1-norm are perfectly of second order.

The absence of artificial boundary conditions also implies that the error on local mass conservation is distributed over the interior on the domain. This was shown before in figure 2.2.

2.3.5 Beltrami flow

The Beltrami flow is one of the rare test problems where an exact fully three-dimensional solution of the Navier-Stokes equations is derived. It has its origin from [ES94] and has been later studied by [LMW12]. The domain is $\Omega = (-1, 1)^3$ and global Dirichlet

boundary conditions are imposed by the exact solution

$$\begin{aligned} v_1(x,y,z,t) &= -a\, e^{-d^2\, t}\ (e^{a\, x}\ \sin(d\, z + a\, y) + \cos(d\, y + a\, x)\ e^{a\, z}) \\ v_2(x,y,z,t) &= -a\, e^{-d^2\, t}\ (e^{a\, x}\ \cos(d\, z + a\, y) + e^{a\, y}\ \sin(a\, z + d\, x)) \\ v_3(x,y,z,t) &= -a\, e^{-d^2\, t}\ (e^{a\, y}\ \cos(a\, z + d\, x) + \sin(d\, y + a\, x)\ e^{a\, z}) \\ p(x,y,z,t) &= -0.5\, a^2\, \rho\, e^{-d^2\, t}\, (2\ \cos(d\, y + a\, x)\ e^{a\,(z+x)}\ \sin(d\, z + a\, y) \\ &\quad + 2\, e^{a\,(y+x)}\ \sin(a\, z + d\, x)\ \cos(d\, z + a\, y) \\ &\quad + 2\ \sin(d\, y + a\, x)\ e^{a\,(z+y)}\ \cos(a\, z + d\, x) + e^{2\, a\, z} + e^{2\, a\, y} + e^{2\, a\, x})\ . \end{aligned} \tag{2.77}$$

The Beltrami flow has the property that the velocity and vorticity vectors are aligned, namely $d\, v - \nabla \times v = 0$ The source term is given by $f = 0$, the density, viscosity are set to $\rho = \mu = 1$. The constants a and d may be chosen arbitrarily and have been set to $a = \pi/4$, $d = \pi/2$ as in [ES94]. Computations were done on a $50 \times 50 \times 50$ cubic mesh.

Figure 2.8 shows the error and convergence rates as a function of Δt for the RIPCS. The green curves show the L^2-error for the velocity and the red curves the H_0^1-error for the velocity obtained by the polynomial degrees $p = 2, 3$. The curves grouped by the same color are almost identical meaning that the splitting error is dominant in the measured range of Δt. It can be concluded from the figure that the error is fully second order convergent in both norms.

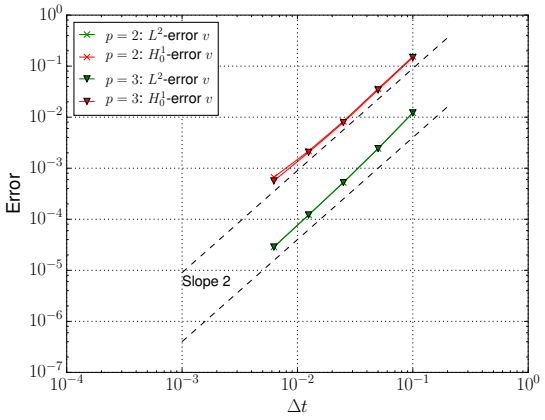


Figure 2.8: Errors and convergence rates at final time T=0.5 for the Beltrami test problem using the RIPCS with the spatial discretizations $\mathcal{Q}_2/\mathcal{Q}_1$, $\mathcal{Q}_3/\mathcal{Q}_2$.

2.4 Conclusion - Which method to choose

After the extensive numerical experiments on operator splitting methods, we close this chapter on a discussion which method to choose.

The choice of time stepping in the viscous substep in all variants of the projection methods presented here can be among onestep, multistep and IMEX Runge-Kutta methods. Though linear multistep methods are most easily applied for constant time step

sizes. Especially a BDF-approximation is required in the pressure Poisson equation for the consistent splitting schemes, c.f. (2.65) for the right-hand side. Further it has been observed that multistep methods such as BDF2 are less stable than Alexander's second order method or the Fractional-Step θ-method for high Reynolds numbers. In the IMEX Runge-Kutta method the diffusion is treated implicit and the nonlinear convection term is treated explicit. This follows from the drastic restriction of the time step size from the diffusive term based on the Courant number since it scales with h^2 instead of h for convective term. Also the explicit treatment of the convective term leads to linear equations of Helmholtz type. The application of IMEX Runge-Kutta methods did not provide a stable method for in-time evolution. According to the formula of the convective flux, $F(v) = v \otimes v$, the matrix $P(v, n)$ is not real diagonalizable for $v \cdot n = 0$. Hence the incompressible Navier-Stokes equations are a degenerate hyperbolic system. The characteristics of degenerate hyperbolic systems is scarcely discussed in the literature. We think that the case $v \cdot n = 0$ triggers instabilities if convection is treated explicitly. In the end we take diagonal implicit Runge-Kutta methods, preferably the Fractional-Step θ-method or Alexander's second order method. Adaptive time stepping based on a CFL-criterion is straightforward.

The second order CSS which is fully accurate for global Dirichlet boundary conditions is unstable in presence of outflow boundary conditions if the time step is small. A behavior that has been also reported in [GMS06]. Hence we choose second-order pressure-correction schemes, in particular the RIPCS which has superior accuracy over the IPCS.

Another reason that puts the pressure-correction schemes in favor is the $H(\mathrm{div})$ postprocessing in the projection step. Both variants return a velocity field that satisfies the discrete continuity equation exactly, and is in consequence locally mass conservative. As the postprocessed velocity fulfills the discrete continuity equation, those procedures both define a projection operator. Raviart-Thomas reconstruction of the discrete pressure gradient can also be done in lower order $H(\mathrm{div})$ approximations where local mass conservation still can be achieved. Raviart-Thomas reconstruction of the full Helmholtz flux provides in addition a pointwise divergence-free velocity and gives a smoother visualization of the flow field. Properties that are not given by other discrete Helmholtz decompositions and also by the consistent splitting schemes.

3 Solvers, High-Performance and implementation issues

Based on the development of a simulation method for incompressible fluid flow, this chapter is devoted to the evolution of the Navier-Stokes Boussinesq system, reducing complexity in the solution and an efficient implementation to realize high-performance computations. In the first Section 3.1 the solvers used across the whole range are discussed. The Navier-Stokes Boussinesq system is decoupled by means of operator splitting. The linear and nonlinear solvers for the individual problems that include elliptic/parabolic partial differential equations are presented next. In Section 3.2 the Spectral Discontinuous Galerkin Method (SDGM) and sum-factorization technique are described. The latter extracts common factors in the tensor product structure present in SDGM. Development of a code that combines sum-factorization with DG in the linear/nonlinear and elliptic/parabolic partial differential equations is a major part of this work. Sum-factorization practice for tensor product bases leads to a significant reduction of computational complexity in variational form evaluation respective on-the-fly preconditioned operator evaluation and thereby in the solution of those equations. In Section 3.3 the performance of the code is investigated. GFLOPs/s and MDOFs/s rates are measured on main and repeatedly occurring components of the solver which are variational form evaluation and matrix-free linearized operator application. Test cases start with a simple affine mesh covering a cuboid where geometry transformations are cheap and go to more complicated multilinear geometries. Scalability studies of our implementation for a realistic example problem are carried out. Weak and strong scaling on the BWFORDEV development cluster in Heidelberg for the 3D Taylor-Green vortex problem is investigated.

3.1 Solution of the Navier-Stokes Boussinesq system

3.1.1 Linearization in transport and buoyancy

The system (1.1)- (1.4) or (1.15)- (1.18), respectively, has more nonlinearities compared to the individual problems due to coupling as the advective field is also an unknown for example. By means of operator splitting it is simplified to a sequence of solutions to the Navier-Stokes equations and convection-diffusion equations. It is summarized in the following Algorithm 3.1:

Algorithm 3.1 Solution approach for the Navier-Stokes Boussinesq system:

Given are (v_h^k, p_h^k) and θ_h^k, c_h^k at time t^k.

1. Keep θ_h^k fixed in the buoyancy forcing term. Compute (v_h^{k+1}, p_h^{k+1}) at time t^{k+1} by doing one time step $\Delta t^{k+1} = t^{k+1} - t^k$ on the fully discrete system of

$$\partial_t v - \frac{1}{\mathrm{Re}}\Delta v + (v \cdot \nabla)v + \nabla p = f + \theta_h^k e_d - \frac{1}{\mathrm{Ro}} e_d \times v$$
$$\nabla \cdot v = 0 \, .$$

2. Use velocity v_h^{k+1} as advection field and compute $\theta_h^{k+1}, c_h^{k+1}$ at time t^{k+1} by doing one time step $\Delta t^{k+1} = t^{k+1} - t^k$ on each discrete formulation of the convection-diffusion equation

$$\partial_t \theta - \frac{1}{\mathrm{Re\ Pr}}\Delta \theta + \nabla \cdot (v_h^{k+1}\theta) = 0$$
$$\partial_t c - \frac{1}{\mathrm{Re\ Sc}}\Delta c + \nabla \cdot (v_h^{k+1} c) = 0$$

   individually.

3. Set $k = k + 1$ and go to 1.

For the Navier-Stokes problem use a fractional stepping technique established in Chapter 2. The scalars are simulated with the spatial discretization presented in [MPB17]. It is based on the conservative form of the advection term, so the velocity field v_h^{k+1} need not be divergence-free but has to be an element of $H(\mathrm{div}; \Omega)$. Unless an advection field is used with continuous normal component across the interfaces in a mesh, additional numerical artifacts are excited in the temperature and concentration field. However the Raviart-Thomas postprocessing techniques in the projection step meet this requirement and return a v_h^{k+1} that is in $H(\mathrm{div}; \Omega)$. Time stepping is fully implicit as in the viscous substep of Navier-Stokes problem.

Parallel implementation

Before heading to the solution of the subproblems in the operator splitting approach, we address here in brief the parallelization with domain decomposition. In the first instance a non-overlapping DG method is used. Decomposition is done on the level of linear systems. Preconditioners are therefore either inexact additive Schwarz methods (subdomain solves to be specified) or a hybrid AMG-DG. The latter has been initiated in [BBS12] and recently further developed in [BMMP18]. We also summarize this preconditioner again below.

3.1.2 Navier-Stokes viscous substep

All of the discussion held here can be straightforwardly transferred to the method of lines (MOL) discretization for the scalars in the Navier-Stokes Boussinesq system. The difference is that the spatial part is linear there.

Recall the section on temporal discretization of the viscous substep. We have explained that equations of the form (2.69) have to be solved for in-time advancement. Such nonlinear systems are solved here with Newton's method. Formulated in the space X_h^p it can be written:

Algorithm 3.2 Solution procedure for the root of $R: X_h^p \to X_h^p$ with Newton's method.

1. Current iterate v_h, compute $R(v_h)$. If norm of $R(v_h)$ sufficiently small, stop successfully.
2. Solve linear system: Find $\delta v_h \in X_h^p$ s.t. $R'(v_h)\delta v_h = R(v_h)$. R' is the Fréchet derivative of R.
3. Update $v_h = v_h - \lambda \delta v_h$ with line search parameter λ, $\lambda = 1$ undamped Newton, and go to 1.

As X_h^p is finite dimensional an equivalent view passes in the vector space $\mathbb{R}^{\dim(X_h^p)}$ of coefficients. Still we write $R: \mathbb{R}^{\dim(X_h^p)} \to \mathbb{R}^{\dim(X_h^p)}$. The Fréchet derivative becomes the Jacobian matrix $\in \mathbb{R}^{\dim(X_h^p)\times\dim(X_h^p)}$ in step 2.

$$\begin{aligned}\nabla_z R(z) &= \nabla_z \left(\frac{\alpha_{ii}}{\Delta t^{k+1}} M_h z + \beta_{ii} \mathcal{L}_h(z, t^k + \delta_i \Delta t^{k+1}) \right) \\ &= \frac{\alpha_{ii}}{\Delta t^{k+1}} M_h + \beta_{ii} A_h(z)\end{aligned}$$

where $A_h(z)$ is the linearized operator of $\mathcal{L}_h(z, t^k + \delta_i \Delta t^{k+1})$ at position z.

Several types of Newton's method 3.2 are well-known that can be used in combination. The first one being already implicitly formulated in Algorithm 3.2, is to use the Fréchet derivative/Jacobian, respectively, instead of an approximation to it. Then it is known that for sufficient regularity in R the undamped Newton method converges locally quadratic.
Another type are inexact Newton methods where the linear system in step 2 is only solved with finite and preferably low tolerance. In order to maintain quadratic convergence the accuracy in each Newton iteration to solve the linear system has to be chosen carefully. We realize this by setting the accuracy to the quadratic rate of convergence of the last two residual norms $\|R(z)\|_2$. Such an inexact Newton method is initiated by a fixed linear reduction in the first linear solve.

Finally we arrive at Newton-Krylov and Jacobian-Free Newton-Krylov (JFNK) methods. In Newton-Krylov, Krylov subspace methods are used to compute a correction in step 2. This becomes feasible together with the loose tolerance coming from inexact

Newton methods. The term JFNK is commonly used in the community [KK04]. We use however the synonym matrix-free for the abbreviation JF. As Krylov solvers, variants of the GMRes method are used that get restarted after every 20th iteration. Unpreconditioned GMRes [SS86] needs an acceptable amount of iterations (in the order of $4 \sim 5$ restarts) on structured quadrilateral/hexahedral grids without anisotropies. An alternative is to use the flexible GMRes method [Saa93, Saa03] which is formulated in Algorithm 3.3.

Algorithm 3.3 The flexible GMRes(m) method. If the preconditioner T_i is constant during the iterations then this method is equal to right-preconditioned GMRes.

initial guess $x^{(0)}$, dimension m of the Krylov subspace
compute $r^{(0)} = b - Ax^{(0)}, \beta = \left\|r^{(0)}\right\|_2$ and $v^{(1)} = r^{(0)}/\beta$
for $i = 1, 2, \ldots, m$ **do**
  solve $T_i z^{(i)} = v^{(i)}$
  compute $w = Az^{(i)}$
  for $k = 1, \ldots, i$ **do**
    $h_{k,i} = w^T v^{(k)}$
    $w = w - h_{k,i} v^{(k)}$
  end for
  $h_{i+1,i} = \|w\|_2$
  $v^{(i+1)} = \frac{w}{h_{i+1,i}}$
  define upper Hessenberg $\bar{H}_i = \{h_{k,j}\}_{1 \le k \le j+1,\, 1 \le j \le i}$, and $Z_i = [z^{(1)}, \ldots, z^{(i)}]$
  compute $\left\|r^{(i)}\right\|_2 = \min_y \left\|\beta e_1 - \bar{H}_i y\right\|_2$
  if $\left\|r^{(i)}\right\|_2$ small enough **then**
    compute $y_i = \operatorname{argmin}_y \left\|\beta e_1 - \bar{H}_i y\right\|_2$ and $x^{(i)} = x^{(0)} + Z_i y_i$
    stop successfully
  end if
end for
compute $y_m = \operatorname{argmin}_y \left\|\beta e_1 - \bar{H}_m y\right\|_2$ and $x^{(m)} = x^{(0)} + Z_m y_m$
set $x^{(0)} \leftarrow x^{(m)}$ and go to 1.

If the preconditioner is constant during the iterations then it is equal to right-preconditioned GMRes. Effects of preconditioning strategies are studied by Taylor expansion in R after a Newton update:

$$R(z_h - \lambda \delta z_h) = R(z_h) - \lambda \nabla_z R(z) \mid_{z=z_h} \delta z_h + \mathcal{O}(\|\delta z_h\|_2^2) \; .$$

A negligible quadratic error in δz_h yields a domain indicator where the linearization is valid. Moreover $R(z_h)$ is the right-hand side in the linear system. With a zero initial iterate that is used throughout in the Krylov solver it is also the initial defect, $R(z_h) - \nabla_z R(z) \mid_{z=z_h} 0 = R(z_h)$. In right-preconditioning the unpreconditioned and preconditioned defect are the same for every Krylov iteration. So at the final iterate when the convergence criterion from inexact Newton is met, the defect is $R(z_h) - \nabla_z R(z) \mid_{z=z_h}$

δz_h. Consequently, in right-preconditioning the final linear defect at previous Newton iteration and the nonlinear residual/initial linear defect at current Newton iteration are equal up to linear order. We observe that this minimizes the amount of Newton iterations compared to left-preconditioning for instance where the defects depend on the preconditioner.

In step 2, the traditional approach is to assemble the linearized operator, do operator applications with matrix-vector products and construct a preconditioner from the assembled operator. Now in the matrix-free method the linearization point is stored and operator applications are done on-the-fly. The essential is to have a matrix-free preconditioner to obtain also a fast convergent iterative method. Those preconditioners are inexact additive Schwarz methods with subdomain solves of the type Jacobi, Gauss-Seidel (GS) or symmetric Gauss-Seidel (SGS). The on-the-fly subdomain solves are summarized in Section 3.2.6. For a detailed description refer to [BMMP18].

3.1.3 Pressure Poisson equation

The solution of the Poisson equation for the pressure correction, step i) in Algorithms 2.1 - 2.4, is accomplished with the Conjugate Gradient method and a hybrid AMG-DG preconditioner. The idea of the particular hybrid multigrid preconditioner originated from [BBS12]. The combination of matrix-free block smoothers and low-order subspace corrections is described in detail in the subsequent publication [BMMP18] We consider low-order subspaces spanned by the piecewise constant $\mathcal{Q}_0$ and conforming piecewise linear elements $\mathcal{Q}_1$. The subspace matrix is re-discretized so that the DG operator is not required for this purpose and the subspace problem is solved using the aggregation based algebraic multigrid (AMG) solver described in [Bla10, BIB12]. The computationally most expensive components in this multigrid algorithm are the operator application and block-smoothing on the DG level - given by the decomposition in the inexact additive Schwarz - which are implemented in a matrix-free way.

The Conjugate Gradient method is equipped with the energy error estimator from [ST05] as a stopping criterion. The quality of the estimation depends on the *error-bandwidth* or delay with respect to the previous iterates, see Algorithm 3.4 for instance. Also observe that the quantities needed for the estimator are already available through the Lanczos procedure and thus this does not produce a significant overhead. The relative norm of the residual is not guaranteed to decrease in the Conjugate Gradient iterations. As numerical experiments in [BMMP18] indicate: The energy error estimation is more robust which reflects in a monotonic and smooth decrease of the curve during the iterations. We use an error-bandwidth of 4 as a greater value does not have a discernible impact on the results.

In case of purely Dirichlet or periodic boundary conditions the matrix in the Poisson equation for the pressure correction - when discretized in the space Q_h^{p-1} - is rank one deficient. We have for $1 \in Q_h^{p-1}$: $\alpha(1, q_h) = 0 \; \forall q_h \in Q_h^{p-1}$. Let $q_{h\,i}, i \in \{1, ..., \dim(Q_h^{p-1})\}$, be a basis of the space Q_h^{p-1}. Since the basis functions form a partition of unity, the belonging coefficients vector to the function $\equiv 1$ in Q_h^{p-1} is $1 \in \mathbb{R}^{\dim(Q_h^{p-1})}$. The matrix

$(\alpha(q_{h\,j}, q_{h\,i}))_{i,j=1}^{\dim(Q_h^{p-1})}$ in the Poisson equation for the pressure correction is symmetric. Moreover its kernel is $\mathrm{Ker}(\alpha(q_{h\,j}, q_{h\,i})) = \mathrm{span}\{1\}$ and it is positive definite on the space $\mathrm{span}\{1\}^\perp$. The well-known relations from linear algebra between the orthogonal complement of image and kernel, respectively,

$$\begin{aligned} \mathrm{Im}(\alpha(q_{h\,j}, q_{h\,i}))^\perp &= \mathrm{Ker}(\alpha(q_{h\,j}, q_{h\,i})), \\ \mathrm{Im}(\alpha(q_{h\,j}, q_{h\,i})) &= \mathrm{Ker}(\alpha(q_{h\,j}, q_{h\,i}))^\perp \end{aligned} \tag{3.1}$$

state that the right-hand side in the Poisson equation needs to fulfill a consistency requirement in terms of solvability. If the right-hand side has a contribution in direction to the kernel then there is no solution to the system. In exact arithmetic a breakdown of unpreconditioned Lanczos recurrence is to happen at some iteration. However, it is easy to check that the right-hand side $(b(w_h, q_{h\,i}) - r(q_{h\,i}; t))_i^{\dim(Q_h^{p-1})}$ is in $\mathrm{span}\{1\}^\perp$:

$$\begin{aligned} \sum_{i=1}^{\dim(Q_h^{p-1})} (b(w_h, q_{h\,i}) - r(q_{h\,i}; t)) \cdot 1 &= b(w_h, 1) - r(1; t) \\ &= -r(1; t) && \text{using Remark 1 on bil. form } b \\ &= 0 && \text{either compatibility cond. on } g \end{aligned}$$

for $\Gamma_h^D = \partial\Omega$ or periodic boundary conditions. If the consistency requirement is met the unpreconditioned Conjugate Gradient method passes in the subspace $\mathrm{Ker}(\alpha(q_{h\,j}, q_{h\,i}))^\perp$. Though care needs to be taken in constructing a preconditioner that returns a correction in the right subspace and does not destroy desirable properties such as symmetry or positive definiteness. We assure this by explicitly orthogonalizing to the kernel after preconditioning. The example of constant null space elimination is included in Algorithm 3.4.

Algorithm 3.4 The preconditioned Conjugate Gradient method with constant null space correction and error estimation.

initial guess $x^{(0)}$, calculate $r^{(0)} = b - Ax^{(0)}$, delay d for error estimation
solve $Tz^{(0)} = r^{(0)}$
orthogonalize to null space $z^{(0)} = z^{(0)} - \frac{(1, z^{(0)})}{(1,1)}\, 1$
$\rho_0 = (r^{(0)}, z^{(0)})$
$p^{(0)} = z^{(0)}$
for $i = 0, 1, \ldots$ **do**
  $q^{(i)} = Ap^{(i)}$
  $\alpha_i = (p^{(i)}, q^{(i)})$
  $\lambda_i = \rho_i / \alpha_i$
  $x^{(i+1)} = x^{(i)} + \lambda_i p^{(i)}$
  $r^{(i+1)} = r^{(i)} - \lambda_i q^{(i)}$
  solve $Tz^{(i+1)} = r^{(i+1)}$
  orthogonalize to null space $z^{(i+1)} = z^{(i+1)} - \frac{(1, z^{(i+1)})}{(1,1)}\, 1$
  $\rho_{i+1} = (r^{(i+1)}, z^{(i+1)})$

$\beta_{i+1} = \rho_{i+1}/\rho_i$
$p^{(i+1)} = z^{(i+1)} + \beta_{i+1} p^{(i)}$
save $\lambda_i \rho_i$ for error estimation
if $i + 1 \geq d$ **then**
Gauss quadrature approximation $\sqrt{\sum_{k=i}^{i+1-d} \lambda_k \rho_k} \leq \left\| x - x^{(i+1-d)} \right\|_A$
end if
end for

3.2 Sum-factorization

3.2.1 Tensor product Finite Element functions

Upon choosing a basis for $Q_h^p = \text{span}(\phi_1, \ldots, \phi_{N_h})$, on a given element $E \in \mathcal{E}_h$ a finite element function $u_h \in Q_h^p$ is in the span of $\{\hat{\phi}_j(\mu_E^{-1}(x))\}$ for $j \in \mathcal{J}$. Here the $\hat{\phi}_j$ are the $|\mathcal{J}|$ shape functions on the reference element spanning the polynomials $\mathbb{Q}_{p,d}$. The two main assumptions for the algorithms developed are:

1. Typically for discontinuous Galerkin we choose global basis functions that have support in only one element. This implies a disjoint partitioning of the index set $\{1, \cdots, N_h\}$. Together with an ordering imposed on the elements in $\mathcal{E}_h$, the vectors $\in \mathbb{R}^{N_h}$ and matrices $\in \mathbb{R}^{N_h \times N_h}$ associated to Q_h^p can be written in block notation. Finally this leads to the so called *minimal stencil* property, the off-block-diagonal of a matrix in this notation is only non-zero if two elements share a face. This property will become crucial later in the construction of matrix-free preconditioners.

2. The shape functions have tensor product structure,

$$\hat{\phi}_j(\hat{x}) = \hat{\phi}_{(j_1,\ldots,j_d)}(\hat{x}_1, \ldots, \hat{x}_d) = \prod_{k=1}^{d} \hat{\theta}_{j_k}^{(k)}(\hat{x}_k) \tag{3.2}$$

where $\mathcal{J} = J^{(1)} \times \ldots \times J^{(d)}$, $J^{(k)} = \{1, \ldots, n_k\}$, consists of d-tuples enumerating the shape functions and $\hat{\theta}_{j_k}^{(k)}$ is the one-dimensional basis function number j_k in direction k on the reference element.

In principle, the one-dimensional basis can be different for every direction and a different basis can be chosen for each element (anisotropic *hp*-refinement).

For the numerical evaluation of integrals quadrature of appropriate order is used:

$$\begin{aligned} \int_{\hat{E}} f(\hat{x}) \, \mathrm{d}\hat{x} &= \int_0^1 \cdots \int_0^1 f(\hat{x}_1, \cdots, \hat{x}_d) \, \mathrm{d}\hat{x}_1 \cdots \mathrm{d}\hat{x}_d \\ &= \sum_{i_1 \in I^{(1)}} \cdots \sum_{i_d \in I^{(d)}} f\left(\xi_{i_1}^{(1)}, \ldots, \xi_{i_d}^{(d)}\right) w_{(i_1,\ldots,i_d)} + \text{error}. \end{aligned} \tag{3.3}$$

Each integral over $[0,1]$ is approximated by a 1D quadrature rule with points $\xi_{i_k}^{(k)}$ and weights $w_{i_k}^{(k)}$, $k = 1, \cdots, d$, that may be different per direction and per element. In multi-index notation we have then a quadrature formula with points $\xi_{(i_1,\cdots,i_d)} = \left(\xi_{i_1}^{(1)}, \ldots, \xi_{i_d}^{(d)}\right)$ and weights $w_{(i_1,\ldots,i_d)} = \prod_{k=1}^{d} w_{i_k}^{(k)}$ that have tensor product form. $i_k \in I^{(k)} = \{1, \ldots, m_k\}$ and $\mathcal{I} = I^{(1)} \times \ldots \times I^{(d)}$ is the index set of all quadrature points. As above quadrature type and quadrature order can be chosen different per element and direction.

However in our implementation the one-dimensional polynomials and quadratures are chosen to be the same in every direction for all elements, i.e. $n_1 = \cdots = n_d = n = p+1$ and $m_1 = \cdots = m_d = m$ but not necessarily $m = n$. As 1D polynomials we take the Lagrange polynomials at $p+1$ Gauss-Lobatto-Legendre quadrature points. Those are abbreviated by GLL polynomials. The particular non-equidistant Lagrange nodes which include the end-points of the interval as well ensure that the maximum magnitude of the polynomials never grows beyond unity even for high degrees. As a demonstration the GLL basis of degree 3 and 10 with corresponding nodes are displayed in figures 3.1 and 3.2. Quadrature points $\xi_{i_k}^{(k)}$ and weights $w_{i_k}^{(k)}$ are given by Gauss-Legendre numerical integration of order q to be specified below. The number of nodes is then $m = \lfloor q/2 \rfloor + 1$. Both the GLL polynomials and quadrature formula are transformed to the 1D reference element $[0,1]$.

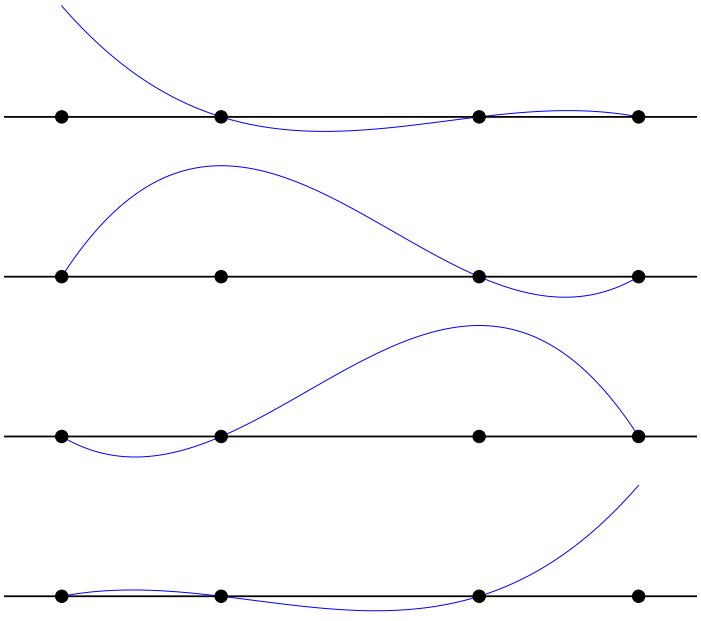


Figure 3.1: Shape of all GLL basis polynomials for degree $p = 3$ transformed to the unit interval $[0,1]$. Depicted are also the distribution of Lagrange nodes on the x-axis.

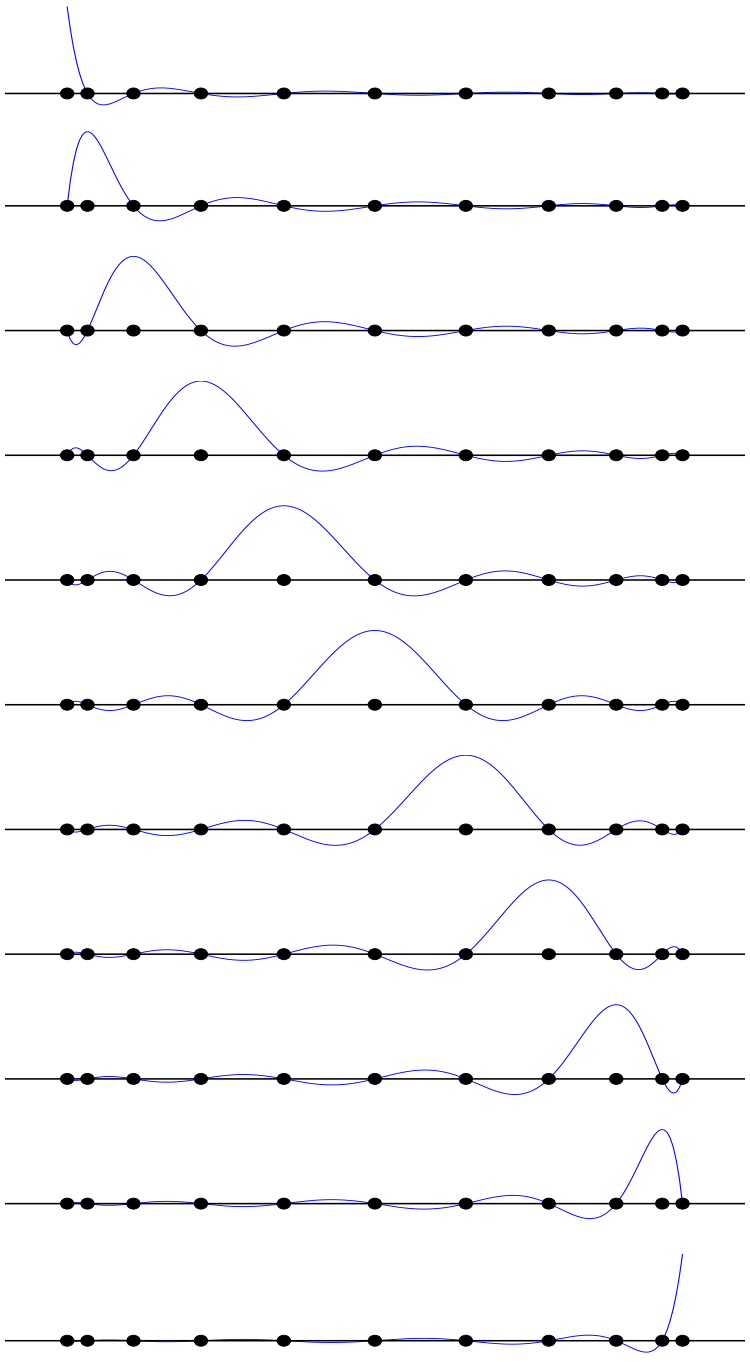


Figure 3.2: Shape of all GLL basis polynomials for degree $p = 10$ transformed to the unit interval $[0, 1]$. Depicted are also the distribution of Lagrange nodes on the x-axis.

3.2.2 Geometry maps as Finite Element functions

In this subsection we want to review formulas for transformation of volume and surface integrals that arise in the DG-discretization for the Navier-Stokes Boussinesq system. Together with exploiting tensor-product structure on geometry maps we head for an efficient implementation of the quantities encoding geometrical information. It has already been implicitly stated by the shape of the mesh elements at the beginning of Section 2.1 that $\mu_E \in \mathbb{Q}_{1,d} \times \cdots \times \mathbb{Q}_{1,d}\ \forall E \in \mathcal{E}_h$, i.e. μ_E is a multilinear transformation from the reference cube $\hat{E}$ to E. Special cases that require a separate consideration in terms of efficient implementation are:
(1) μ_E being affine linear: $\hat{\nabla}\mu_E(\hat{x})$ is constant for all $\hat{x} \in \hat{E}$. E has the shape of a parallelepiped in $\mathbb{R}^d$.
(2) μ_E being further an axi-parallel transformation: $\hat{\nabla}\mu_E(\hat{x})$ is a constant diagonal matrix for all $\hat{x} \in \hat{E}$. E has the shape of a rectangle or a rectangular cuboid. A square/cube is a special case where $\hat{\nabla}\mu_E(\hat{x})$ is even a multiple of the identity matrix.

Some more notation is needed to describe the evaluation of face integrals. The embedding of face e into $E^{\text{int}}, E^{\text{ext}}$ is described by maps $\eta_e^{\text{int,ext}} : \hat{e} \to \hat{E}$ of the corresponding reference elements such that $\mu_{E^{\text{int,ext}}(e)} \circ \eta_e^{\text{int,ext}} = \mu_e$ holds. The maps $\eta_e^{\text{int,ext}} : \hat{e} \to \hat{E}$ map coordinate number $q \in \{1, \dots, d-1\}$ in $\hat{e}$ to coordinate number $\pi_e^{\text{int,ext}}(q) \in \{1, \dots, d\}$ in $\hat{E}$. We may extend the map $\pi_e^{\text{int,ext}}$ to $\{1, \dots, d\}$ by requiring $\pi_e^{\text{int,ext}}(d)$ to be the unique coordinate number that face $\hat{e}$ is perpendicular to in the embedded $\hat{E}$. Thereby $\pi_e^{\text{int,ext}}$ is a permutation of $\{1, \dots, d\}$ and the numbers $\pi_e^{\text{int,ext}}(q), q \in \{1, \dots, d-1\}$ are referred to as the *tangential directions* in $\hat{E}$. This implies an extension of the map $\eta_e^{\text{int,ext}}$ to $\hat{e} \times \{0\}$ and the normal component can take the values $\eta^{\text{int,ext}}_{e,\pi_e^{\text{int,ext}}(d)}(0) \in \{0, 1\}$. Finally, since faces in the volume reference element are axis-parallel, the component

$$\left(\eta_e^{\text{int,ext}}(\hat{s})\right)_{\pi_e^{\text{int,ext}}(q)} = \eta^{\text{int,ext}}_{e,\pi_e^{\text{int,ext}}(q)}(\hat{s}) = \eta^{\text{int,ext}}_{e,\pi_e^{\text{int,ext}}(q)}(\hat{s}_q)$$

is a function of one variable and in absence of hanging nodes it is even an isometry.

The following formulas with respect to change of coordinates can be found in [Gue15, AE04] and also hold for diffeomorphic μ_E.

- Transformation of volume measures: The volume measure in E at x and in $\hat{E}$ at $\hat{x}$ are related by integration with substitution:

$$\mathrm{d}x = \left|\det \hat{\nabla}\mu_E(\hat{x})\right| \mathrm{d}\hat{x}\ . \tag{3.4}$$

- Transformation of surface measures: The surface measure in e at $s = \mu_e(\hat{s})$ and in $\hat{e}$ at $\hat{s}$ are given by

$$\mathrm{d}s = \sqrt{\det(\hat{\nabla}\mu_e(\hat{s})^T \hat{\nabla}\mu_e(\hat{s}))}\mathrm{d}\hat{s}\ . \tag{3.5}$$

  Note that the Jacobian of the map μ_e is a rectangular matrix, $\hat{\nabla}\mu_e(\hat{s}) \in \mathbb{R}^{d\times(d-1)}$. Therefore the transformation is accomplished by the quantity $\hat{\nabla}\mu_e(\hat{s})^T\hat{\nabla}\mu_e(\hat{s}) \in \mathbb{R}^{(d-1)\times(d-1)}$ that is referred to as the *surface metric tensor*.

We want to give a sketch of the proof for (3.5) in $\mathbb{R}^3$. It is known from analysis and differential geometry that $\hat{\nabla}\mu_e(\hat{s})e_1$ and $\hat{\nabla}\mu_e(\hat{s})e_2$, here $e_1, e_2 \in \mathbb{R}^2$, form tangent vectors at $\mu_e(\hat{s})$ to the hypersurface parametrized by μ_e. Now let us write for brevity columnwise $\hat{\nabla}\mu_e(\hat{s}) = [a, b] \in \mathbb{R}^{3\times 2}$. An object that measures the infinitesimal area at each point on the hypersurface can be calculated in terms of the cross-product $\hat{\nabla}\mu_e(\hat{s})e_1 \times \hat{\nabla}\mu_e(\hat{s})e_2 = a \times b$. The square of the integration element can be recast to Euclidean scalar products and norms:

$$\begin{aligned}
\|a \times b\|_2^2 &= \|a\|_2^2 \|b\|_2^2 \sin^2\theta \\
&= \|a\|_2^2 \|b\|_2^2 (1 - \cos^2\theta) \\
&= \|a\|_2^2 \|b\|_2^2 \left(1 - \frac{|a \cdot b|^2}{\|a\|_2^2 \|b\|_2^2}\right) \\
&= \|a\|_2^2 \|b\|_2^2 - |a \cdot b|^2 \ .
\end{aligned}$$

On the other hand consider the surface metric tensor:

$$\begin{aligned}
\hat{\nabla}\mu_e(\hat{s})^T \hat{\nabla}\mu_e(\hat{s}) &= \begin{bmatrix} a^T \\ b^T \end{bmatrix} [a,\ b] = \begin{bmatrix} a^T a & a^T b \\ b^T a & b^T b \end{bmatrix} \\
\Rightarrow \det(\hat{\nabla}\mu_e(\hat{s})^T \hat{\nabla}\mu_e(\hat{s})) &= (a^T a)(b^T b) - (a^T b)^2
\end{aligned}$$

which finishes the sketch of proof. □

- Transformation of tangent vectors: Let $\hat{x} \in \partial\hat{E}$ and $\hat{t}(\hat{x})$ be a unit tangent vector on $\partial\hat{E}$ at $\hat{x}$. Then
$$t(x) = \hat{\nabla}\mu_E(\hat{x})\hat{t}(\hat{x}) \tag{3.6}$$
is tangent on ∂E at $x = \mu_E(\hat{x})$.

- Transformation of normal vector: Let $\hat{x} \in \partial\hat{E}$ and $\hat{n}(\hat{x})$ be the unit outward normal vector on $\partial\hat{E}$ at $\hat{x}$. Then
$$n(x) = \hat{\nabla}\mu_E(\hat{x})^{-T}\hat{n}(\hat{x}) \tag{3.7}$$
is an outward normal vector on ∂E at $x = \mu_E(\hat{x})$.

It can be immediately seen that $n(x)$ and $t(x)$ are orthogonal to each other. Outward normal vector on e with respect to E^{int} and tangent vectors can be computed by the geometrical information on the inside cell with the outward normal $\hat{n}$ and $\hat{t}$ on the corresponding face on $\hat{E}$ that are easily known. This has advantages over using the map μ_e. The problem is not the computation of tangent vectors on e that are the columns of $\hat{\nabla}\mu_e(\hat{s})$ but the construction of the outer normal. Especially in three dimensions the order of column vectors in $\hat{\nabla}\mu_e(\hat{s})$ need to be chosen carefully for the cross product. It depends on the orientation of tangent vectors on e with respect to E^{int} and requires also geometry and embedding information with respect to the inside cell.

- Transformation of surface measures on ∂E: The surface measures on ∂E at x and on $\partial \hat{E}$ at $\hat{x}$ are related as follows:

$$\mathrm{d}s = \left|\det \hat{\nabla}\mu_E(\hat{x})\right| \left\|\hat{\nabla}\mu_E(\hat{x})^{-T}\hat{n}(\hat{x})\right\|_2 \mathrm{d}\hat{s} \; . \tag{3.8}$$

- Relation of surface measures on ∂E to the surface metric tensor: If the embedding map η_e^{int} is an isometry (exterior case can be treated similar), then for $\hat{x} = \eta_e^{\mathrm{int}}(\hat{s})$

$$\sqrt{\det(\hat{\nabla}\mu_e(\hat{s})^T \hat{\nabla}\mu_e(\hat{s}))} = \left|\det \hat{\nabla}\mu_E(\hat{x})\right| \left\|\hat{\nabla}\mu_E(\hat{x})^{-T}\hat{n}(\hat{x})\right\|_2 \tag{3.9}$$

  and the transformation weight in (3.5) can be replaced.

  Identity (3.9) together with (3.6), (3.7) are very important since the transformation of surface integrals can be expressed by using only quantities attached to adjacent cells. For instance the computation of an outer normal vector by (3.7) can be reused on the right-hand side of (3.9). Therein the Jacobian inverse transposed is calculated via inversion of the Jacobian for which closed formulas exist in 2D and 3D. Such type of inversion requires the determinant of the Jacobian that can also be reused in the transformation weight (3.9).

Now let us go back to multilinear transformations and investigate how those can be implemented efficiently with amongst others sum-factorization. We also point out to savings in the affine linear case and axi-parallel case. The idea is motivated that corners of the quadrilateral/hexahedral element E can inexpensively be obtained in the DUNE grid implementations considered. Hence let a corner of an element E be denoted by

$$V(E, \hat{k}) := \mu_E(\hat{k}) \text{ for } \hat{k} \in \{0,1\}^d \; . \tag{3.10}$$

Compare with the construction of reference elements in 1D, 2D and 3D in figure A2.1. Further let a corner of the face e be denoted by

$$V(e, \hat{k}) := \mu_e(\hat{k}) \text{ for } \hat{k} \in \{0,1\}^{d-1} \; . \tag{3.11}$$

Choosing equidistant Lagrange polynomials with nodes at the end points 0 and 1 as one-dimensional basis functions, a Lagrange basis of $\mathbb{Q}_{1,d}$ can be built whose nodes are the 2^d corners of $\hat{E}$. We name those functions $\hat{\phi}_j = \hat{\phi}_{(j_1,\ldots,j_d)} \in \mathbb{Q}_{1,d},\ j_1,\ldots,j_d \in \{0,1\}$. The numbering of this basis is given by the lexicographic numbering of the corners in the 1D/2D/3D reference element, respectively, c.f. figure A2.1. Then the coefficient of basis function j is given by the image of the j-th corner in $\hat{E}$ under μ_E. Eventually the map μ_E can be expressed as

$$x = \sum_{j_1,\ldots,j_d=0}^{1} V(E, (j_1,\ldots,j_d))\ \hat{\phi}_{(j_1,\ldots,j_d)}(\hat{x}) \; . \tag{3.12}$$

The global coordinate $x \in E$ is interpreted as a vector-valued finite element function, similar to the velocity ansatz space. Obviously the Jacobian has the expansion

$$\hat{\nabla}\mu_E(\hat{x}) = \sum_{j_1,\dots,j_d=0}^{1} V(E,(j_1,\dots,j_d)) \otimes \hat{\nabla}\hat{\phi}_{(j_1,\dots,j_d)}(\hat{x}) \; . \tag{3.13}$$

Owing to both identities (3.12), (3.13) global coordinates and Jacobian of a multilinear geometry map can be evaluated efficiently using sum-factorization, as we will see in the next subsection.

In the affine linear case it is $x = B_E\hat{x} + b_E$ with $B_E \in \mathbb{R}^{d\times d}$, $b_E \in \mathbb{R}^d$ both not depending on $\hat{x}$. As E is a parallelepiped it is sufficient to request for the corners

$$\begin{aligned} V(E,(0,0,0)) &= b_E \\ V(E,(1,0,0)) &= B_E e_1 + b_E \\ V(E,(0,1,0)) &= B_E e_2 + b_E \\ V(E,(0,0,1)) &= B_E e_3 + b_E \end{aligned}$$

in three dimensions. Then writing $\hat{x} = \sum_{i=1}^{d} \hat{x}_i e_i$, we have for $B_E\hat{x}$

$$\begin{aligned} B_E\hat{x} &= \hat{x}_1\Big(V(E,(1,0,0)) - V(E,(0,0,0))\Big) + \hat{x}_2\Big(V(E,(0,1,0)) - V(E,(0,0,0))\Big) \\ &\quad + \hat{x}_3\Big(V(E,(0,0,1)) - V(E,(0,0,0))\Big) \\ &= x - b_E \; . \end{aligned} \tag{3.14}$$

The factors grouped by $\hat{x}_i$ are called the extents of the parallelepiped and make up the i-th column of the Jacobian $\hat{\nabla}\mu_E(\hat{x})$. The global coordinate can consequently be expressed as

$$\begin{aligned} x &= V(E,(0,0,0)) + \hat{x}_1\Big(V(E,(1,0,0)) - V(E,(0,0,0))\Big) \\ &\quad + \hat{x}_2\Big(V(E,(0,1,0)) - V(E,(0,0,0))\Big) + \hat{x}_3\Big(V(E,(0,0,1)) - V(E,(0,0,0))\Big) \; . \end{aligned} \tag{3.15}$$

In contrast to the general multilinear case the global coordinate $x \in E$ can be calculated instead by fused-multiply-add operations with the extents. The Jacobian is sufficed to compute once per cell. The two-dimensional case can be derived analogously.

If further μ_E describes an axi-parallel transformation, only the corners $V(E,(0,0,0))$ and $V(E,(1,1,1))$ have to be requested as E is an axis-aligned rectangular cuboid in three dimensions. The extents for the parallelepiped can be retrieved as

$$\begin{aligned} V(E,(1,0,0)) - V(E,(0,0,0)) &= \Big(e_1 \otimes e_1\Big)\Big(V(E,(1,1,1)) - V(E,(0,0,0))\Big) \\ V(E,(0,1,0)) - V(E,(0,0,0)) &= \Big(e_2 \otimes e_2\Big)\Big(V(E,(1,1,1)) - V(E,(0,0,0))\Big) \end{aligned}$$

$$V(E,(0,0,1)) - V(E,(0,0,0)) = \Big(e_3 \otimes e_3\Big)\Big(V(E,(1,1,1)) - V(E,(0,0,0))\Big)$$

such that (3.15) simplifies to

$$\begin{aligned} x = V(E,(0,0,0)) &+ e_1 \cdot \Big(V(E,(1,1,1)) - V(E,(0,0,0))\Big)\, \hat{x}_1 e_1 + e_2 \cdot \Big(V(E,(1,1,1)) \\ &- V(E,(0,0,0))\Big)\, \hat{x}_2 e_2 + e_3 \cdot \Big(V(E,(1,1,1)) - V(E,(0,0,0))\Big)\, \hat{x}_3 e_3 \end{aligned} \tag{3.16}$$

and (3.14) to

$$\hat{\nabla}\mu_E(\hat{x}) = \operatorname{diag}\Big(V(E,(1,1,1)) - V(E,(0,0,0))\Big)\,. \tag{3.17}$$

3.2.3 Start-up examples

In the previous two subsections we have briefly introduced tensor product finite elements, expressed geometry maps in terms of tensor product finite elements and pointed out there the use of sum-factorization. Here we want to demonstrate sum-factorization in the finite element solution. The evaluation of volume and face integrals in the finite element discretization is carried out in three consecutive steps. This procedure is explained in simple examples.

Volume integrals Consider the volume integral

$$(u_h, \phi)_{0,\Omega} = \sum_{E \in \mathcal{E}_h} (u_h, \phi)_{0,E}\,.$$

As usual in the finite element approach the per-element summand $(u_h, \phi)_{0,E}$ is transformed to the reference element $\hat{E}$ for all ϕ with support on E:

$$\begin{aligned} &\int_{\hat{E}} \hat{u}_h(\hat{x})\hat{\phi}_j(\hat{x})\big|\det \hat{\nabla}\mu_E(\hat{x})\big|\mathrm{d}\hat{x}\,, \quad j \in \mathcal{J} \\ &\overset{(3.3)}{\approx} \sum_{i_1 \in I^{(1)}} \cdots \sum_{i_d \in I^{(d)}} \hat{u}_h\left(\xi_{i_1}^{(1)}, \ldots, \xi_{i_d}^{(d)}\right) \hat{\phi}_{(j_1,\ldots,j_d)}\left(\xi_{i_1}^{(1)}, \ldots, \xi_{i_d}^{(d)}\right) w_{(i_1,\ldots,i_d)} \gamma_{(i_1,\ldots,i_d)} \end{aligned} \tag{3.18}$$

where

$$\gamma_{(i_1,\ldots,i_d)} := \left|\det \hat{\nabla}\mu_E\left(\xi_{i_1}^{(1)}, \ldots, \xi_{i_d}^{(d)}\right)\right|$$

is the Jacobian determinant evaluated at quadrature points. The computation of the quantity (3.18) is done in three stages.

Stage 1 $\hat{u}_h$ is evaluated at all quadrature points exploiting tensor product structure. The local coefficients vector is interpreted as a rank d tensor $x_{(j_1,\ldots,j_d)}$.

$$y_{(i_1,\ldots,i_d)} := \hat{u}_h\left(\xi_{i_1}^{(1)}, \ldots, \xi_{i_d}^{(d)}\right) = \sum_{j_d \in J^{(d)}} \cdots \sum_{j_1 \in J^{(1)}} A_{i_d,j_d}^{(d)} \cdot \ldots \cdot A_{i_1,j_1}^{(1)} x_{(j_1,\ldots,j_d)}$$

The matrices $A^{(k)}_{i_k,j_k} = \hat{\theta}^{(k)}_{j_k}\left(\xi^{(k)}_{i_k}\right)$ contain the 1D basis functions evaluated at the 1D quadrature points. This step can be done efficiently using sum-factorization which reduces the point evaluations to a sequence of d matrix-vector products with tensor rotations in between. The tensor rotations are required to have the ordering $(i_1, \ldots, i_d)$ in the multiindex consistent with the orientation in the volume reference element. Note that $\gamma_{(i_1,\ldots,i_d)}$ can be evaluated with the same exploitation if the multilinear formula (3.13) is chosen. Otherwise the Jacobian and its determinant are computed once per cell, Equation (3.14) and (3.17).

Stage 2 In the quadrature loop multiply with weights and geometry transformations - compute componentwise $r_i = y_i w_i \gamma_i \; \forall i \in \mathcal{I}$.

Stage 3 Finish quadrature, multiply with all local basis functions $\hat{\phi}_j, \; j \in \mathcal{J}$.

$$\sum_{i_d \in I^{(d)}} \cdots \sum_{i_1 \in I^{(1)}} A^{(d)}_{i_d,j_d} \cdot \ldots \cdot A^{(1)}_{i_1,j_1} r_{(i_1,\ldots,i_d)} = \quad (3.18)$$

This has the same abstract structure as *Stage 1*. In the sum-factorization we use the transpose of the matrices $A^{(k)}_{i_k,j_k}$.

Face integrals Now consider the surface integral $([u_h],[\phi])_{0,e}$ on a face e. Let us choose basis functions ϕ that have only support on $E^{\text{int}}(e)$. In the finite element approach the integral is transformed to the reference face element with the hypersurface parameterization μ_e:

$$\begin{aligned}([u_h],[\phi])_{0,e} &= ([u_h],\phi^{\text{int}})_{0,e} \\ &= \int_{\hat{e}} \left(\hat{u}^{\text{int}}_h(\eta^{\text{int}}_e(\hat{s})) - \hat{u}^{\text{ext}}_h(\eta^{\text{ext}}_e(\hat{s}))\right) \hat{\phi}_j(\eta^{\text{int}}_e(\hat{s})) \sqrt{\det(\hat{\nabla}\mu_e(\hat{s})^T \hat{\nabla}\mu_e(\hat{s}))}\mathrm{d}\hat{s} \; ,\end{aligned}$$

for $j \in \mathcal{J}$. If η^{int}_e is an isometry, the hypersurface map can be moreover avoided by means of Equation (3.9)

$$\begin{aligned}= \int_{\hat{e}} &\left(\hat{u}^{\text{int}}_h(\eta^{\text{int}}_e(\hat{s})) - \hat{u}^{\text{ext}}_h(\eta^{\text{ext}}_e(\hat{s}))\right) \hat{\phi}_j(\eta^{\text{int}}_e(\hat{s}))\left|\det \hat{\nabla}\mu_{E^{\text{int}}}(\eta^{\text{int}}_e(\hat{s}))\right| \\ &\left\|\hat{\nabla}\mu_{E^{\text{int}}}(\eta^{\text{int}}_e(\hat{s}))^{-T}\hat{n}(\eta^{\text{int}}_e(\hat{s}))\right\|_2 \mathrm{d}\hat{s} \; .\end{aligned}$$

For the sake of clarity we will concentrate on the interior part of the jump operator and apply quadrature in the style of (3.3):

$$\sum_{i_1 \in I^{(1)}} \cdots \sum_{i_{d-1} \in I^{(d-1)}} \hat{u}^{\text{int}}_h\left(\eta^{\text{int}}_e(\xi^{(1)}_{i_1},\ldots,\xi^{(d-1)}_{i_{d-1}})\right) \hat{\phi}_{(j_1,\ldots,j_d)}\left(\eta^{\text{int}}_e(\xi^{(1)}_{i_1},\ldots,\xi^{(d-1)}_{i_{d-1}})\right)$$
$$w_{(i_1,\ldots,i_{d-1})}\gamma_{(i_1,\ldots,i_{d-1})} \; . \quad (3.19)$$

$\gamma_{(i_1,\ldots,i_{d-1})}$ is the rank $(d-1)$ tensor containing the transformation weights at all face quadrature points. As in a volume integral the computation is split in three consecutive

stages.

Stage 1 Evaluate $\hat{u}_h^{\text{int}}$ at all face quadrature points exploiting tensor product structure. The local coefficients vector is still interpreted as a rank d tensor.

$$\begin{aligned} y_{(i_1,\dots,i_{d-1})} &:= \hat{u}_h^{\text{int}}\left(\eta_e^{\text{int}}(\xi_{i_1}^{(1)},\dots,\xi_{i_{d-1}}^{(d-1)})\right) \\ &= \sum_{j_1 \in J^{(\pi_e^{\text{int}}(1))}} \dots \sum_{j_d \in J^{(\pi_e^{\text{int}}(d))}} B_{i_1,j_1}^{(1)} \dots B_{i_{d-1},j_{d-1}}^{(d-1)} B_{i_d,j_d}^{(d)} x_{(j_{\pi_e^{\text{int}-1}(1)},\dots,j_{\pi_e^{\text{int}-1}(d)})} \end{aligned}$$

Here we introduced the matrices that contain the 1D basis functions evaluated at the 1D quadrature points in the $(d-1)$-dimensional tensor product quadrature on the reference face - and as a last factor the endpoints of the interval $(0,1)$ since the local basis functions are defined on the reference element.

$$B_{i_k,j_k}^{(k)} = \begin{cases} \hat{\theta}_{j_k}^{(\pi_e^{\text{int}}(k))}\left(\eta_{e,\pi_e^{\text{int}}(k)}^{\text{int}}\left(\xi_{i_k}^{(k)}\right)\right) & 1 \le k < d, j_k \in J^{(\pi_e^{\text{int}}(k))}, i_k \in I^{(k)} \\ \hat{\theta}_{j_k}^{(\pi_e^{\text{int}}(d))}\left(\eta_{e,\pi_e^{\text{int}}(d)}^{\text{int}}(0)\right) & k = d, \qquad j_k \in J^{(\pi_e^{\text{int}}(d))}, i_k = 1 \end{cases}$$

Again this step can be done efficiently using sum-factorization by a sequence of d matrix-vector products with tensor rotations in between. To achieve optimal complexity the normal direction is done first. The output $y_{(i_1,\dots,i_{d-1})}$ of stage 1 is consistent with the orientation in the face reference element. Furthermore $\hat{u}_h^{\text{ext}}$ is processed the same way and also the tensor $\gamma_{(i_1,\dots,i_{d-1})}$ when given by the multilinear formula (3.13).

Stage 2 In the quadrature loop multiply with weights and geometry transformations - compute componentwise $r_i = y_i w_i \gamma_i \ \forall i \in I^{(1)} \times \dots \times I^{(d-1)}$. The exterior contribution from *Stage 1* is calculated in the same go.

Stage 3 Finish quadrature, multiply with all local basis functions $\hat{\phi}_j,\ j \in \mathcal{J}$.

$$\sum_{i_{d-1} \in I^{(d-1)}} \dots \sum_{i_1 \in I^{(1)}} B_{i_d,j_{\pi_e^{\text{int}}(d)}}^{(d)} B_{i_{d-1},j_{\pi_e^{\text{int}}(d-1)}}^{(d-1)} \dots B_{i_1,j_{\pi_e^{\text{int}}(1)}}^{(1)} r_{(i_1,\dots,i_{d-1})} = (3.19)$$

Again this has the same abstract structure as *Stage 1*. In the sum-factorization we use the transpose of the matrices $B_{i_k,j_k}^{(k)}$. To achieve optimal complexity the normal direction is done last.

3.2.4 Variational form evaluation

For the rest of this section, focus is on the fully discretized viscous substep. Novelties compared to the application of sum-factorization in the linear convection-diffusion equation are: Nonlinearities and a system of equations. The latter application has been discussed in [MPB17] before in detail and covers here the evolution of passive/active scalars, respectively, and the pressure Poisson equation. Parameter functions as the advective velocity field or the continuity equation on the right-hand side of pressure Poisson

- that are given by finite element functions as well - can also be directly evaluated using sum-factorization.

Integration over volume elements and faces is done with Gauss-Legendre quadrature which is optimal/minimal in the number of nodes in order to fulfill the given accuracy. Quadrature order in the viscous substep is chosen to be at least $q = 3p$ so that the number of points per coordinate direction is $m = \lfloor 3p/2 \rfloor + 1$. In the remaining problems of the operator splitting approach quadrature order is chosen to be at least $q = 2p$ so that the number of points per coordinate direction here is $m = \lfloor 2p/2 \rfloor + 1 = p + 1$. Both quadrature orders are exact for the corresponding variational forms on affine cells and avoid aliasing effects due to inexact quadrature. On non-affine meshes order is increased by a constant $q_{add} > 0$.

The basis set for the velocity space is straightforwardly constructed from the basis of the space Q_h^p. We set $\{\varphi_1, \ldots, \varphi_{dN_h}\} = \{\phi_j e_s\}_{j,s=1}^{N_h,d}$ where e_s is the canonical basis vector in the s-th direction. The basis functions are sorted in interleaved ordering. Given an element $E \in \mathcal{E}_h$ each DOF in the system associated to this element is mapped contiguously into one block. Now the matrices and vectors in this ordering look formally the same as in the partitioning for Q_h^p such that the number of blocks is equal but their size is larger.

We have seen in the Sections 2.2.5, 3.1.2 that the nonlinear residual evaluation requires computations of the form $\left(\Delta t^{k+1}\right)^{-1} \alpha_{ii} M_h z_h + \beta_{ii} \mathcal{L}_h(z_h, t^k + \delta_i \Delta t^{k+1})$ (note that we have silently dropped the index indicating the stage from now on). Testing in this expression with functions $\varphi_{\tilde{i}} = \phi_i e_r$ and using the ansatz $\varphi_{\tilde{j}} = \phi_j e_s$ leads to a set of d equations of convection-diffusion type for the velocity components (recall that e.g. $(\varphi_{\tilde{j}}, \varphi_{\tilde{i}})_{0,\Omega} = (\phi_j, \phi_i)_{0,\Omega} \delta_{rs}$ and $a(\varphi_{\tilde{j}}, \varphi_{\tilde{i}}) = a(\phi_j, \phi_i)\delta_{rs}$). The belonging variational forms in the space Q_h^p are evaluated efficiently using sum-factorization in three steps per mesh entity, c.f. start-up examples in Section 3.2.3. A detailed description of this procedure can furthermore be found in [MPB17]. Algorithm 3.5 summarizes the computations done for the viscous substep.

Algorithm 3.5 Variational form evaluation $\rho(v_h, \varphi)_{0,\Omega}$ and $a(v_h, \varphi) + \rho c(v_h, \varphi) + b(\varphi, p_h^\bigstar)$ $\forall \varphi$ in basis set of X_h^p.

---

for E in ordered $\mathcal{E}_h$ **do**
  (1) Compute $\hat{v}_h, \hat{\nabla} \hat{v}_h$ and $\hat{p}_h^\bigstar$ for all quadrature points on corresp. $\hat{E}$
  (2) Compute coefficients at all quadrature points
  (3) Compute $\rho(v_h, \varphi)_{0,E}$ and $a_E(v_h, \varphi) + \rho c_E(v_h, \varphi) + b_E(\varphi, p_h^\bigstar)$
    for all basis functions φ with support on E
  for $e \in \Gamma_h^{\text{int}} \cap \partial E$ and e not treated yet **do**
    (4) Compute $\hat{v}_h^{\text{int}}, \hat{v}_h^{\text{ext}}, \hat{\nabla} \hat{v}_h^{\text{int}}, \hat{\nabla} \hat{v}_h^{\text{ext}}$ and $\hat{p}_h^{\bigstar,\,\text{int}}, \hat{p}_h^{\bigstar,\,\text{ext}}$
      for all quadrature points on corresponding $\hat{e}$
    (5) Compute coefficients at all quadrature points
    (6) Compute $a_{\text{int},e}(v_h, \varphi^{\text{int,ext}}) + \rho c_{\text{int},e}(v_h, \varphi^{\text{int,ext}}) + b_{\text{int},e}(\varphi^{\text{int,ext}}, p_h^\bigstar)$
      for all basis functions φ^{int}, φ^{ext} with support on $E^{\text{int}}(e)$, $E^{\text{ext}}(e)$
    (7) Mark e as treated
  end for

for $e \in (\Gamma_h^D \cup \Gamma_h^N) \cap \partial E$ **do**
(8) Compute $\hat{v}_h^{\text{int}}, \hat{\nabla}\hat{v}_h^{\text{int}}$ and $\hat{p}_h^{\bigstar,\,\text{int}}$
for all quadrature points on corresponding $\hat{e}$
(9) Compute coefficients at all quadrature points
(10) Compute $a_{\Gamma_h^D,e}(v_h,\varphi) + \rho c_{\Gamma_h^D,e}(v_h,\varphi) + \rho c_{\Gamma_h^N,e}(v_h,\varphi) + b_{\Gamma_h^D,e}(\varphi, p_h^{\bigstar})$
for all basis functions φ with support on E
end for
end for

As said steps (1), (4), (8) in algorithm 3.5 are calculated per velocity component: $\hat{v}_{i,h}, \hat{\nabla}\hat{v}_{i,h}$ for $i \in \{1,\dots,d\}$. Since $\{\varphi \mid \text{supp}(\varphi) \subset E\} = \{\phi e_r \text{ for } r \in \{1,...,d\} \mid \text{supp}(\phi) \subset E\}$, steps (3), (6), (10) in algorithm 3.5 are calculated per test functions ϕ. The explicit pressure contribution and buoyancy forcing term if required in step (1), are also evaluated using sum-factorization. The optimizations with respect to high-performance computations presented in [MPB17] are incorporated into the code. In particular for the viscous substep: In stage 1, $\hat{v}_{i,h}, \hat{\nabla}\hat{v}_{i,h}$ are collected into one SIMD vector instruction for $i \in \{1,\dots,d\}$. In 3D this gives fully loaded AVX2 registers. In stage 2, the quadrature loop is calculated in parallel using the same vector instructions. Finally, stage 3 is repeated d times with different input tensors coming from stage 2. For each (scalar) basis function $\phi, \nabla\phi$ is gathered into one SIMD vector instruction. The sequence of matrix-matrix computations is completed by a horizontal-add in order to accumulate a test function and its gradient into the same residual entry.

3.2.5 Matrix-free linearized operator application

The solution of the linearized systems in Newton's method requires operator applications of the form $\left(\Delta t^{k+1}\right)^{-1}\alpha_{ii}M_h\delta z_h + \beta_{ii}A_h(z_h)\delta z_h$ with linearization point z_h and input δz_h, c.f. Section 3.1.2. At the finite element level the linearized operator can be expressed as a Fréchet derivative of the variational form. Hence let $a'(v_h,\varphi)$ be the Fréchet derivative of $a(v_h,\varphi)$ with respect to v_h for $\varphi \in X_h^p$. If the variational form is linear in v_h, then the application of the Fréchet derivative to an element of the space can be written in terms of the original operator. For example a': $a'(v_h,\varphi)\delta v_h = a(\delta v_h,\varphi)$. Similarly for the mass term. In that case we are back in the setting of [MPB17], Algorithm 1.

Algorithm 3.6 presents the linearized operator application of the viscous substep. As for the variational form evaluation in steps (1), (4), (8) the linearization point is calculated per component - and besides the input for the operator application: $\delta\hat{v}_{i,h}, \hat{\nabla}\delta\hat{v}_{i,h}$ for $i \in \{1,...,d\}$. Steps (3), (6), (10) are calculated per test functions ϕ since $\{\varphi \mid \text{supp}(\varphi) \subset E\} = \{\phi e_r \text{ for } r \in \{1,...,d\} \mid \text{supp}(\phi) \subset E\}$. Recall the high-performance optimizations for the form evaluation with respect to stages 1, 2 and 3 in sum-factorization that are also found in here.

Algorithm 3.6 Matrix-free linearized operator application $\rho(\delta v_h,\varphi)_{0,\Omega}$ and $a'(v_h,\varphi)\delta v_h + \rho c'(v_h,\varphi)\delta v_h\ \forall\varphi$ in basis set of X_h^p.

for E in ordered $\mathcal{E}_h$ **do**

(1) Compute linearization point $\hat{v}_h, \hat{\nabla}\hat{v}_h$ and input $\delta\hat{v}_h, \hat{\nabla}\delta\hat{v}_h$ for all quadrature points on corresponding $\hat{E}$
(2) Compute coefficients at all quadrature points
(3) Compute $\rho(\delta v_h, \varphi)_{0,E}$ and $a'_E(v_h, \varphi)\delta v_h + \rho c'_E(v_h, \varphi)\delta v_h$, action of Fréchet derivatives for all φ with support on E
for $e \in \Gamma_h^{\text{int}} \cap \partial E$ and e not treated yet **do**
(4) Compute linearization point $\hat{v}_h^{\text{int}}, \hat{v}_h^{\text{ext}}, \hat{\nabla}\hat{v}_h^{\text{int}}, \hat{\nabla}\hat{v}_h^{\text{ext}}$ and input $\delta\hat{v}_h^{\text{int}}, \delta\hat{v}_h^{\text{ext}}, \hat{\nabla}\delta\hat{v}_h^{\text{int}}, \hat{\nabla}\delta\hat{v}_h^{\text{ext}}$ for all quadrature points on corresp. $\hat{e}$
(5) Compute coefficients at all quadrature points
(6) Compute $a'_{\text{int},e}(v_h, \varphi^{\text{int,ext}})\delta v_h + \rho c'_{\text{int},e}(v_h, \varphi^{\text{int,ext}})\delta v_h$, action of Fréchet derivatives for all basis functions φ^{int}, φ^{ext} with support on $E^{\text{int}}(e)$, $E^{\text{ext}}(e)$
(7) Mark e as treated
end for
for $e \in (\Gamma_h^D \cup \Gamma_h^N) \cap \partial E$ **do**
(8) Compute linearization point $\hat{v}_h^{\text{int}}, \hat{\nabla}\hat{v}_h^{\text{int}}$ and input $\delta\hat{v}_h^{\text{int}}, \hat{\nabla}\delta\hat{v}_h^{\text{int}}$ for all quadrature points on corresponding $\hat{e}$
(9) Compute coefficients at all quadrature points
(10) Comp. $a'_{\Gamma_h^D,e}(v_h, \varphi)\delta v_h + \rho c'_{\Gamma_h^D,e}(v_h, \varphi)\delta v_h + \rho c'_{\Gamma_h^N,e}(v_h, \varphi)\delta v_h$, action of Fréchet derivatives for all φ with support on E
end for
end for

3.2.6 Matrix-free preconditioning

Let $T_h(z_h)$ denote a preconditioner to the linearized operator $\left(\Delta t^{k+1}\right)^{-1}\alpha_{ii}M_h + \beta_{ii}A_h(z_h)$. In preconditioned Krylov subspace methods it is required to compute $T_h(z_h)^{-1}\delta z_h$ for inputs δz_h. Now, in the matrix-free approach, the preconditioner inverse application is to happen on-the-fly without assembling the linearized operator. We consider $T_h(z_h)$ to be either a block Jacobi, block GS or block SGS approximation. Utilization as a subdomain solve in an inexact additive Schwarz preconditioner is straightforward by restriction. Following the block partitioning of the velocity function space, let $A_h(z_h) = L_h(z_h) + D_h(z_h) + U_h(z_h)$ be decomposed into the strictly lower block triangular part $L_h(z_h)$, the block diagonal $D_h(z_h)$ and the strictly upper block triangular part $U_h(z_h)$. $M_h(z_h)$ is already block diagonal with the choice of support in the basis functions. The action of $T_h(z_h)^{-1}$ involves solutions with the block diagonal $\left(\Delta t^{k+1}\right)^{-1}\alpha_{ii}M_h + \beta_{ii}D_h(z_h)$ that can be carried out per block independently due to the disjoint partitioning of the index set. We recapitulate the three implementations used to solve with the block diagonal:

1. **Matrix-based (MX)** The traditional approach is to assemble the full linearized operator, do operator applications with matrix-vector products and realize the preconditioner from the assembled operator by amongst others storing the LU-factorization of the diagonal blocks.

2. **Partially matrix-free (PMF)** In this approach the operator application and the application of the off-block-diagonal elements in the GS and SGS process are done matrix-free. Only the block diagonal of the linearized operator is assembled to store the LU-factorization per element.

3. **Matrix-free (MF)** As in the partially matrix-free variant the operator and off-block-diagonal applications are done matrix-free. The diagonal blocks are inverted iteratively in a local matrix-free solver. This requires on-the-fly applications of the diagonal blocks.

For all three items the optimizations discussed in Sections 3.2.4, 3.2.5 are utilized, though the focus is on the last two items.

Local solvers and the block Jacobi method

In the PMF variant the systems associated to the blocks of $\left(\Delta t^{k+1}\right)^{-1}\alpha_{ii}M_h + \beta_{ii}D_h(z_h)$ are solved with the previously set up LU-factorization. It requires in addition not only the action of the local Fréchet derivatives but their explicit setup.

In the MF variant we choose BiCGSTAB as a element-local solver. It is a Krylov subspace method with a short recurrence of operator applications that is suited for non-symmetric diagonal blocks as well. Algorithm 3.7 contains the computations needed to apply $M_h\delta z_h$ and $D_h(z_h)\delta z_h$ on-the-fly in order to realize such a BiCGSTAB solver. Note that nevertheless both traces of the linearization point on a face need to be computed. Also all interior faces are visited twice, thus the application of the block diagonal has the same proportion as a full operator application. Several such local applications are needed per cell in the iterative inversion and therefore it has the computationally most expensive operations part of the outer Krylov solver. A fast inversion is desirable and to speed up the convergence of the local matrix-free solver a point Jacobi preconditioner is used which requires therefore the explicit point diagonal assembly in the linearized operator. The results in [BMMP18] already demonstrate that a loose tolerance in the iterative inversion is sufficient. Here we stop the BiCGSTAB solver after a relative reduction of 10^{-3} in the defect norm is achieved. This does not deteriorate the number of outer iterations in flexible GMRes and puts the approach in a competitive spot.

For structured isotropic quadrilateral/hexahedral meshes local BiCGSTAB is observed to require $4 \sim 5$ iterations per average which corresponds - owing to our implementation - to twice as many block diagonal applications. It outperforms the PMF variant especially for high polynomial degrees of the ansatz space. However the number of average local BiCGSTAB iterations grows significantly in presence of strong anisotropies in the elements even for the mesh still being structured. On such grids where the aspect ratio of elements can become greater than 10, the PMF variant is eventually faster in terms of total time and time per iteration within the outer Krylov loop. The same conclusion has been obtained when solving the SPE10 test case in [BMMP18]. Recall that the LU-factorizations are reused for every preconditioned operator application. In particular for the viscous substep: The block or point diagonal for the Jacobi, GS or SGS

preconditioner setup are computed even once per time step and reused over multiple invocations of Newton's method therein.

Algorithm 3.7 Matrix-free application of the block diagonal contributions $M_h\delta z_h$ and $D_h(z_h)\delta z_h$ on the finite element level $\forall\varphi$ in basis set of X_h^p.

for E in ordered $\mathcal{E}_h$ **do**
  (1) Compute linearization point $\hat{v}_h, \hat{\nabla}\hat{v}_h$ and input $\delta\hat{v}_h, \hat{\nabla}\delta\hat{v}_h$ for all quadrature points on corresponding $\hat{E}$
  (2) Compute coefficients at all quadrature points
  (3) Compute $\rho(\delta v_h, \varphi)_{0,E}$ and $a'_E(v_h,\varphi)\delta v_h + \rho c'_E(v_h,\varphi)\delta v_h$, action of Fréchet derivatives for all φ with support on E
  for $e \in \Gamma_h^{\text{int}} \cap \partial E$ **do**
    (4) Compute linearization point $\hat{v}_h^{\text{int}}, \hat{v}_h^{\text{ext}}, \hat{\nabla}\hat{v}_h^{\text{int}}, \hat{\nabla}\hat{v}_h^{\text{ext}}$ and input $\delta\hat{v}_h^{\text{int}}, \hat{\nabla}\delta\hat{v}_h^{\text{int}}$ for all quadrature points on corresp. $\hat{e}$
    (5) Compute coefficients at all quadrature points
    (6) Compute $a'_{\text{int},e}(v_h,\varphi^{\text{int}})\delta v_h^{\text{int}} + \rho c'_{\text{int},e}(v_h,\varphi^{\text{int}})\delta v_h^{\text{int}}$, action of Fréchet derivatives for all basis functions φ^{int} with support on $E^{\text{int}}(e)$
  end for
  for $e \in (\Gamma_h^D \cup \Gamma_h^N) \cap \partial E$ **do**
    (7) Compute linearization point $\hat{v}_h^{\text{int}}, \hat{\nabla}\hat{v}_h^{\text{int}}$ and input $\delta\hat{v}_h^{\text{int}}, \hat{\nabla}\delta\hat{v}_h^{\text{int}}$ for all quadrature points on corresponding $\hat{e}$
    (8) Compute coefficients at all quadrature points
    (9) Comp. $a'_{\Gamma_h^D,e}(v_h,\varphi)\delta v_h + \rho c'_{\Gamma_h^D,e}(v_h,\varphi)\delta v_h + \rho c'_{\Gamma_h^N,e}(v_h,\varphi)\delta v_h$, action of Fréchet derivatives for all φ with support on E
  end for
end for

Block Gauss-Seidel and symmetric block Gauss-Seidel

As recap for the GS iteration and in the light of developing MF and PMF variants, consider for simplicity the matrix $M_h + \Delta t A_h(z_h)$. With some initial guess $\delta z_h^{(0)}$ and right-hand side d_h compute perpetually

$$\delta z_h^{(k+1)} = \delta z_h^{(k)} + \Big(\omega^{-1}(M_h + \Delta t D_h(z_h)) + \Delta t L_h(z_h)\Big)^{-1}\Big(d_h - (M_h + \Delta t A_h(z_h))\delta z_h^{(k)}\Big)\,.$$

The GS preconditioning process is retrieved as one iteration with zero initial vector. One such iteration can be transformed to the solution with the diagonal blocks post to application of off-block-diagonals $L_h(z_h)\delta z_h^{(k+1)}$ and $U_h(z_h)\delta z_h^{(k)}$. Due to the disjoint partitioning of the index set, $\delta z_h^{(k)}$ can be manipulated in place by a forward substitution to obtain $\delta z_h^{(k+1)}$. Now with respect to the on-the-fly application the two crucial assumptions from Section 3.2.1 come into play. The minimal stencil property allows us to compute these contributions per block row as the integrals over all faces surrounding

the associated E. Compare with Steps (1)-(3) in Algorithm 3.8. The lower triangular solve in the GS process is accomplished by the forward iteration through the ordered mesh elements.

Algorithm 3.8 Matrix-free off-block-diagonals application $L_h(z_h)\delta z_h^{(k+1)}$, $U_h(z_h)\delta z_h^{(k)}$ in a forward substitution followed by a local solve. This implements a partially matrix-free or matrix-free GS process, respectively.

for E obtained by forward iteration in ordered $\mathcal{E}_h$ **do**
  for $e \in \Gamma_h^{\text{int}} \cap \partial E$ **do**
    (1) Compute linearization point $\hat{v}_h^{\text{int}}, \hat{v}_h^{\text{ext}}, \hat{\nabla}\hat{v}_h^{\text{int}}, \hat{\nabla}\hat{v}_h^{\text{ext}}$ and input $\delta\hat{v}_h^{\text{ext}}, \hat{\nabla}\delta\hat{v}_h^{\text{ext}}$ for all quadrature points on corresp. $\hat{e}$
    (2) Compute coefficients at all quadrature points
    (3) Compute $a'_{\text{int},e}(v_h, \varphi^{\text{int}})\delta v_h^{\text{ext}} + \rho c'_{\text{int},e}(v_h, \varphi^{\text{int}})\delta v_h^{\text{ext}}$, action of Fréchet derivatives for all basis functions φ^{int} with support on $E^{\text{int}}(e)$
    (4) Subtract these contributions from local defect $d_{h,E}$
  end for
  (5) Solve local system with the help of e.g. Steps (1)-(9) in Algorithm 3.7 on E
  (6) Update $\delta z_{h,E}$
end for

Owing to our discussion GS is not significantly more expensive than a Jacobi iteration since the time-dominating part is the solution with the diagonal blocks. The corresponding backward GS is obtained by reverse iteration in ordered $\mathcal{E}_h$. It can be shown that one matrix-free SGS process is equivalent to a forward GS iteration followed by a backward sweep.

3.3 High-Performance implementation tests

We have realized the numerical scheme outlined in the Sections 2.1- 2.2 and 3.1- 3.2 within the PDELab [BHM10] finite element framework which is based on the Dune framework [BBD+08]. By design, Dune and PDELab are very general and allow for different mesh types and general discretizations. This flexibility sacrifices - to some extent - performance for generality and in order to obtain a high-performance implementation of our sum-factorized DG assembly, we had to redesign parts of the underlying framework. One part that was crucial for the performance of our code is the implementation of a separate assembler tailored to DG. Like other general-purpose finite element discretization frameworks, PDELab by default uses a temporary buffer to gather all degrees of freedom associated with a single element contiguously into memory before processing the element computations. Likewise, the computational results associated with one element are stored in a buffer and then scattered to corresponding locations in the global data structure. While this cannot be avoided for continuous Galerkin methods, the gather and scatter operations are not necessary in DG where the global data may already be

arranged contiguously in memory. In fact due to our assumptions on the support of basis functions and an ordering imposed on the mesh elements, the index set of the DOFs has a disjoint partitioning. We thus extended PDELab with a DG-specific code path that exploits this block structure of DG problems avoiding any superfluous copy operations.

Another part is to exploit local parallelism of modern CPU architectures by using SIMD vector instructions in the three stages of sum-factorization. For a detailed description thereof we refer to [MPB17]. Our implementation precomputes and caches the values of one-dimensional polynomials and their derivatives used to build the tensor product bases for $\mathbb{Q}_{p,d}$ and $\mathbb{Q}_{p-1,d}$. It also does so for the local coordinates and weights of the d-dimensional and $(d-1)$-dimensional tensor product quadrature rule. This allows to use a flat iteration space during the quadrature loops instead of a nested iteration to reconstruct the multiindex.

3.3.1 Floating-point operations and solution time throughput

Two measures are used here to evaluate the performance of our implementation: (1) Floating-point operations per second expressed as GFLOPs/s and given as a percentage of the machine's maximum floating-point performance. (2) Degrees of freedom per second processed during a full variational form evaluation and matrix-free linearized operator application. Numbers are given in MDOFs/s - note that we prefer this measure over its inverse (time per degree of freedom) because a higher value yields a faster throughput. Good results on the latter measure are always more important from the application point of view, as it gives an accurate level of how fast a real problem can be solved. However, the former is still an interesting measure that allows reasoning about how good a code is suited for a given hardware and how much of the theoretical peak performance is actually reached. For this benchmark methodology we are concerned with Intel Haswell processors offering the AVX2 instruction set which operates on four double-precision floating-point numbers in parallel. We therefore use the wrapper type from [Fog17] that overloads the common arithmetic operators from single- and double-precision for the AVX2 instruction set. Our compute node is equipped with two 16-core Intel Xeon E5-2698 v3 sockets. The peak floating-point performance of this processor is decomposed into its frequency, AVX2 parallelism and 2 fused multiply-add (FMA) instructions per cycle. This gives a theoretical rate of 30.4 GFLOPs/s per core and thus 972.8 GFLOPs/s on a fully loaded node.

In order to give accurate numbers on floating-point operations per second, the number of performed floating-point operations needs to be measured exactly. We exploit the fact that we are within the PDELab discretization framework which uses C++-templates to the extent that we can replace the underlying floating-point type throughout all our simulation. Instead of using `double` and the wrapper type `Vec4d` from [Fog17], respectively, we use a custom templated type which has overloads for all arithmetic operations that increase a global counter and forward the operation to the underlying type. This counting of course introduces a non-negligible performance overhead. We therefore compile different executables from the same source for (1) the benchmark run and time measurement only on the outermost PDELab call, (2) fine-grained time

measurement and (3) operation counting. Apart from counting operations, accurate time measurements are needed. We instrument our code with C macros to start and stop high resolution timers using the `std::chrono` library.

It is important as well to achieve performance of the code with problems of different computational intensity. We therefore have implemented separate code paths for axi-parallel, affine and multilinear geometry transformations μ_E. Those code paths are based on the formulas (3.16)- (3.17) for axi-parallel transformations, (3.14)- (3.15) for affine linear transformations and (3.12)- (3.13) for multilinear transformations. Recall that on an element E for the last code path sum-factorization is employed. Furthermore it is possible to enable/disable evaluations of the right-hand side f, the buoyancy and Coriolis forcing term at compile time depending on the problem.

We are going to focus on GFLOPs/s and MDOFs/s rates attained in the viscous substep. In particular the full variational form evaluation,

$$\frac{1}{\Delta t} M_h z_h + \mathcal{L}_h(z_h, \Delta t) \ , \tag{3.20}$$

c.f. Algorithm 3.5, and matrix-free linearized operator application,

$$\frac{1}{\Delta t} M_h \delta z_h + A_h(z_h) \delta z_h \ , \tag{3.21}$$

c.f. Algorithm 3.6, that are both repeatedly required in implicit time-stepping. Note that M_h and $\mathcal{L}_h(z_h, t)$ are defined below (2.68). Therein the parameter functions f and g are constructed from rational functions of the type $1/(1+\|x\|_2^2)$ where each component is scaled differently in the enumerator, and shifted in space such that the maximum is not always at $x = 0$. The same construction is made for the velocity field $v(x)$ and pressure $p(x)$ which are then L^2-interpolated to obtain $v_h \in X_h^p$ and $p_h^\bigstar \in Q_h^{p-1}$. Thereby the linearization point z_h is given by $v_h = \sum_{j=1}^{dN_h} z_{h,\,j} \varphi_j$ and the input for operator application by $\delta z_h = z_h$. Each core of the node has the computational domain $\Omega = (0,1)^3$ with 20 cells per coordinate direction for this first benchmark. Quadrature order is chosen to be $q = 3p$. A realistic setting where such calculations occur would be the 3D Taylor-Green vortex, Section 4.5, or the 3D turbulent channel flow, Section 4.6.

Figure 3.3 and 3.4 show throughput and floating-point performance of our implementation for the DG-discretization pairs $\mathcal{Q}_p/\mathcal{Q}_{p-1}$, $2 \leq p \leq 7$. The curves grouped by the blue color show the results on affine, axi-parallel cells. The curves grouped by the green color show the results on cells expressed as multilinear geometries. Results are presented for a fully loaded node, 2x Xeon E5-2698 v3, where each core has the same workload of 20 cells per direction. We observe that given a sufficiently high polynomial degree, our code hits the flopbound regime. Matrix-free operator application for affine, axi-parallel transformations performs best in terms of GFLOPs/s rates. 50% of machine's peak capability is reached. For the multilinear code path it is expected that more work per DOF needs to be done. Also the floating-point performance falls to in between $40 - 50\%$. In the case of axi-parallel cells, only d FLOPs are needed to set up $\hat{\nabla}\mu_E(\hat{x})$ once per cell and further d FMA operations to calculate the global coordinate $x = \mu_E(\hat{x})$ using vector math units which is cheaper than the cost of sum-factorized

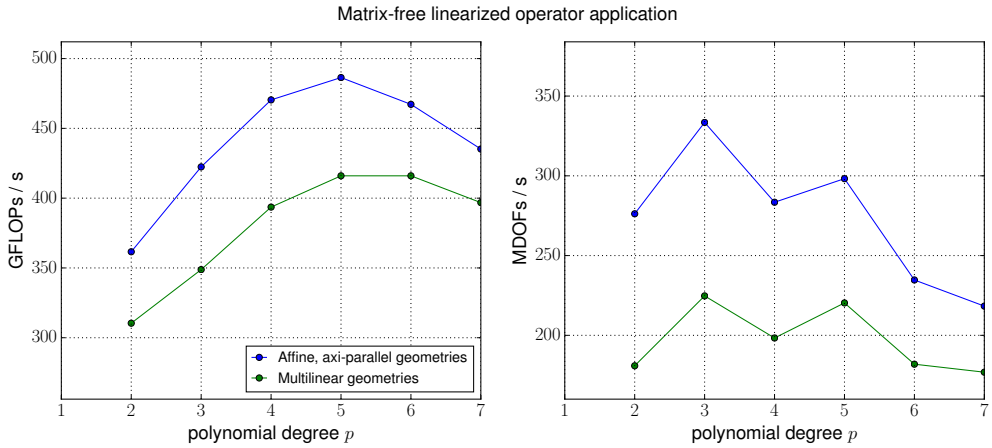


Figure 3.3: Floating-point performance in GFLOPs/s and throughput in MDOFs/s for full linearized operator application in the viscous substep. Measurements for multilinear (green curve) and affine, axi-parallel (blue curve) geometry transformations refer to a fully loaded node, 2x Intel Xeon E5-2698 v3.

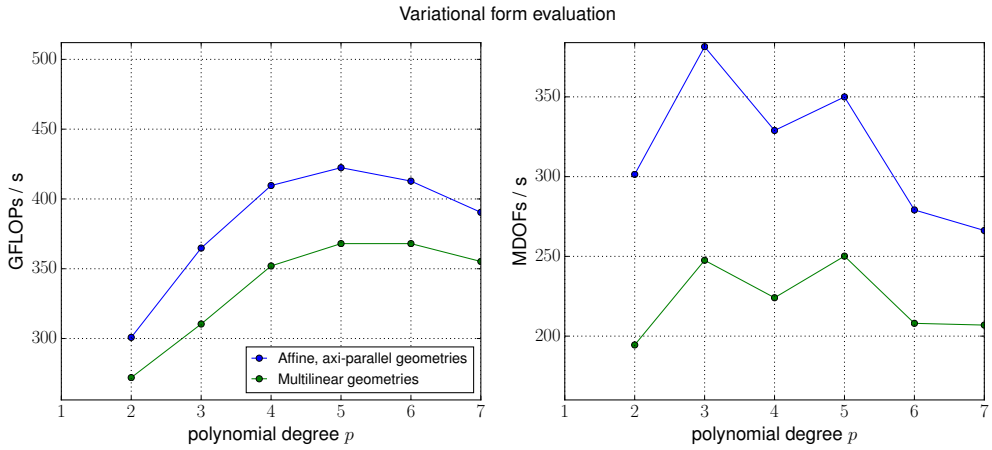


Figure 3.4: Floating-point performance in GFLOPs/s and throughput in MDOFs/s for variational form evaluation in the viscous substep. Measurements for multilinear (green curve) and affine, axi-parallel (blue curve) geometry transformations refer to a fully loaded node, 2x Intel Xeon E5-2698 v3.

evaluation of these quantities. For the variational form evaluation we measure performance for the same two cases. The curves compared to the upper figure 3.3 only differ by an offset in the vertical direction, for both the left and right part. The GFLOPs/s rates measured on operator application are higher whereas the MDOFs/s throughput measured on variational form evaluation is then higher. The latter achieves a rate of 384 MDOFs/s for $p = 3$ in the axi-parallel case. This effect can be explained by the difference with respect to algorithmic complexity. While both procedures compute the linearization point $v_h \in X_h^p$, on-the-fly operator application requires the evaluation of input $\delta v_h \in X_h^p$ as a vector-valued finite element function. In contrast variational form

evaluation computes the explicit pressure $p_h^\bigstar \in Q_h^{p-1}$ which is part of $\mathcal{L}_h$ in Equation (3.20). $p_h^\bigstar$ is a scalar-valued finite element function by one polynomial degree lower than δv_h which grasps the difference in algorithmic complexity.

The second benchmark uses a fully multilinear grid on a non-cuboid computational domain. The setting arises in the 3D flow over a 2D periodic hill. This CFD benchmark is extensively described in [QNE13] and further referenced by [Wan09]. It has been part of the problem collection for the international workshop on High-Order CFD Methods (HiOCFD) until the third edition, [Per15]. The domain Ω is obtained by deformation of a host $\hat{\Omega} = (0, 9) \times (0, 3) \times (0, 4.5)$ such that the crest is located at the outflow plane in streamwise direction and the hill is periodically continued from the opposite inflow plane. We use here however a simpler parameterization of the hill than proposed in [QNE13] which is given by

$$x = \hat{x} + C(3 - \hat{x}_2)\Big[1 + \tanh\Big(B(|\hat{x}_1 - A| - B)\Big)\Big]\, e_2 \qquad x \in \Omega,\ \hat{x} \in \hat{\Omega} \tag{3.22}$$

with the constants $A = 4.5$, $B = 3.5$, $C = 1/6$ as in [Nek17]. A $\mathcal{Q}_1$-approximation to the target domain is depicted in figure 3.5. The host grid for $\hat{\Omega}$ has 20 cuboid cells per direction and the deformed vertices are precomputed by the map (3.22). Each multilinear geometry transformation is then built from the stored vertices.

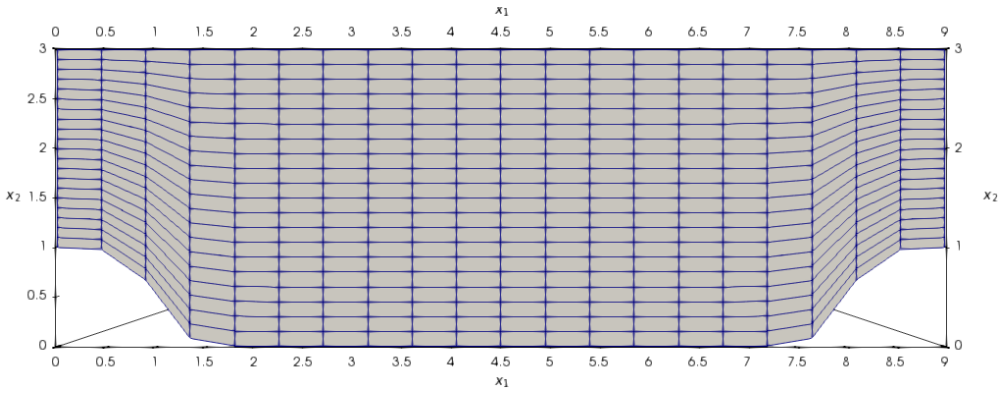


Figure 3.5: Benchmark multilinear grid for the GFLOPs/s and MDOFs/s measurements. Number of cells are 20^3. The deformation represents a coarse approximation to the two-dimensional periodic hill flow.

Figure 3.6 and 3.7 show throughput and floating-point performance of the multilinear code path for the DG-discretization pairs $\mathcal{Q}_p/\mathcal{Q}_{p-1}$, $2 \leq p \leq 7$. The curves are colored in green in order to link to the corresponding measurements presented in figure 3.3 and 3.4. Note here the suitably rescaled y-axes in both plots though. We observe that for the lowest polynomial degrees an overhead of the underlying grid implementation is present as performance results drop by ~ 1 GFLOPs/s/core and ~ 1 MDOFs/s/core, respectively. For increasing polynomial degrees this overhead becomes negligible. Our code hits the flopbound regime, and from $p = 7$ onwards the results converge to the rates depicted above. Differences between on-the-fly operator application and variational form

evaluation with respect to GFLOPs/s and MDOFs/s are furthermore consistent with the above cases and have been already discussed. Notably for matrix-free operator application (examine left part of figure 3.6), 40% of the theoretical peak is still achieved on this multilinear grid manager.

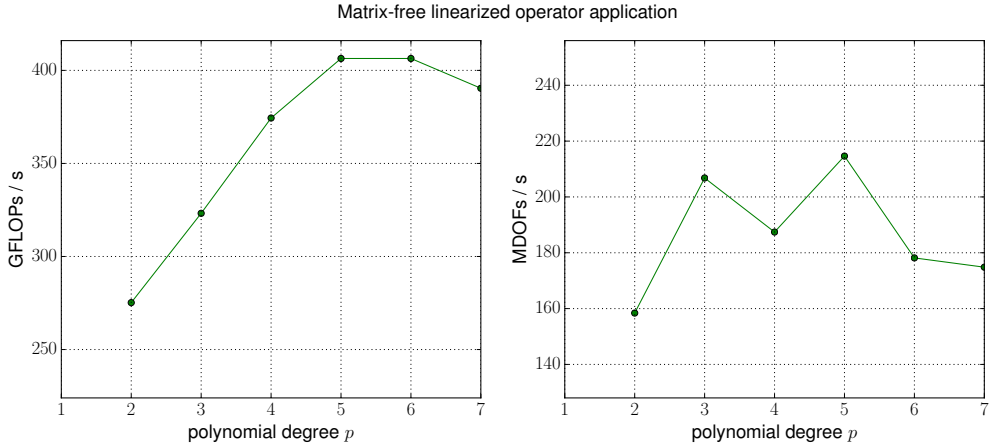


Figure 3.6: Floating-point performance in GFLOPs/s and throughput in MDOFs/s on a multilinear grid for linearized operator application in the viscous substep. Measurements refer to a fully loaded node, 2x Intel Xeon E5-2698 v3.

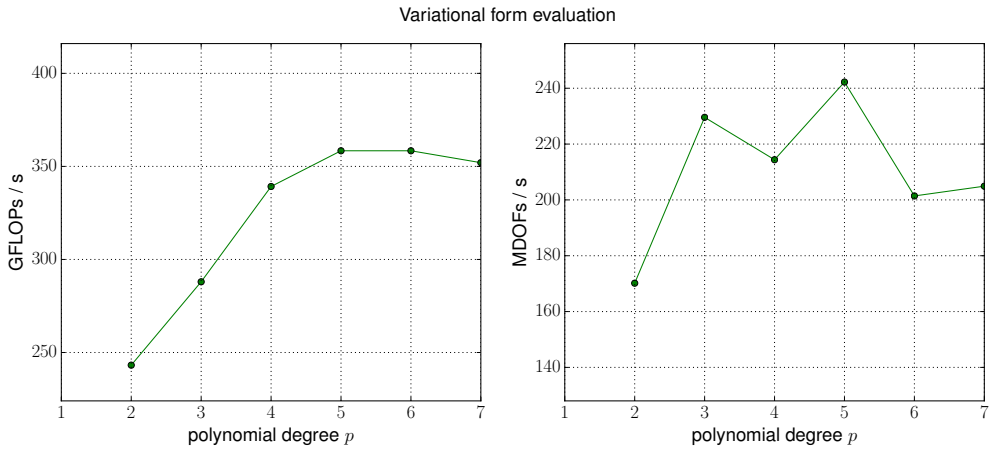


Figure 3.7: Floating-point performance in GFLOPs/s and throughput in MDOFs/s on a multilinear grid for variational form evaluation in the viscous substep. Measurements refer to a fully loaded node, 2x Intel Xeon E5-2698 v3.

The last benchmark uses another fully multilinear grid that is provided by the grid manager from the finite element code `UG`. This grid manager is able to handle unstructured meshes with triangular and quadrilateral elements in 2D, and with tetrahedral and hexahedral elements in 3D. Functionality in DUNE exists to read files in the Gmsh [GR09] mesh generator format, which are processed for storage within the `UG` grid manager. Hence this test case shows the performance of our code for a broader class of

applications with complex geometries involved. The measurement setup here is motivated by the three-dimensional flow around a cylinder obstacle, a well-studied CFD problem that we summarize in Section 4.3. The domain is discretized with 8448 hexahedral elements and is displayed in figure 3.8. For a fair comparison the workload size per core is approximately equal to the size of the previous two benchmarks (8000 cells).

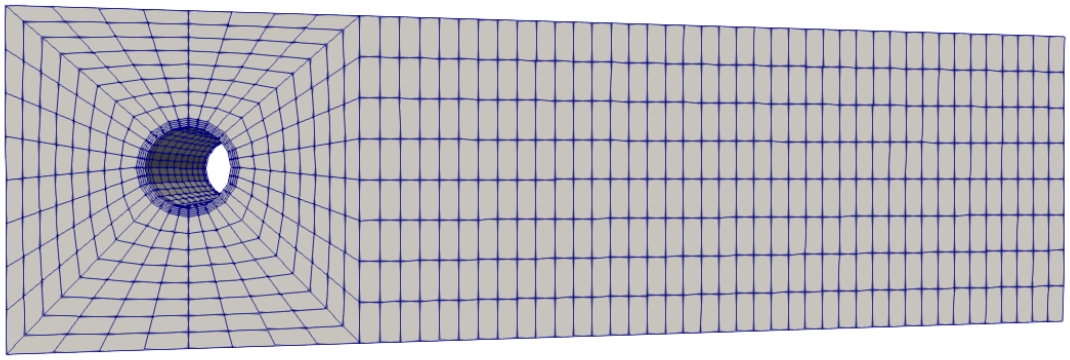


Figure 3.8: Multilinear grid for benchmarking GFLOPs/s and MDOFs/s rates using the grid manager `UG`. The number of cells (8448) is chosen to reach the count for the previous two benchmarks (8000).

Figure 3.9 and 3.10 show throughput and floating-point performance of the multilinear code path achieved with the `UG` grid manager. As above results for the DG-discretization pairs $\mathcal{Q}_p/\mathcal{Q}_{p-1}$, $2 \leq p \leq 7$, are shown. We use the same range for the y-axes as in the previous test case to provide an easier comparison. There is no significant difference to the performance curves in figure 3.6 and 3.7. We observe that for the lowest polynomial degrees an overhead of the underlying grid implementation is again present. Compared to the previous grid manager, performance measures drop further by a minor fraction of ~ 0.5 GFLOPs/s/core and ~ 0.5 MDOFs/s/core. This overhead becomes negligible as the polynomial degree increases, and eventually differences in the rates to the above test cases tend to vanish (from $p = 7$ onwards). Still we are able to reach on this grid manager, 40% of machine's theoretical peak for matrix-free linearized operator application (c.f. left part of figure 3.9).

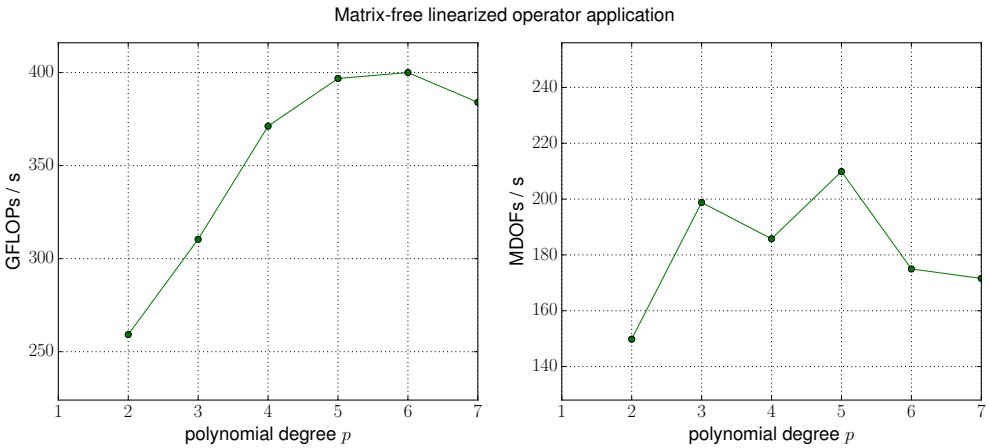


Figure 3.9: Floating-point performance in GFLOPs/s and throughput in MDOFs/s on a multilinear grid managed by `UG`. The curves show the result of linearized operator application in the viscous substep for a fully loaded node, 2x Intel Xeon E5-2698 v3.

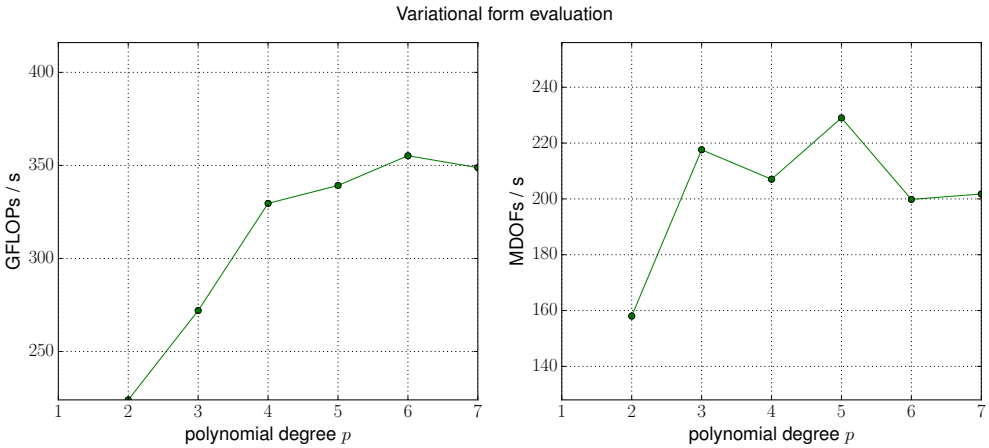


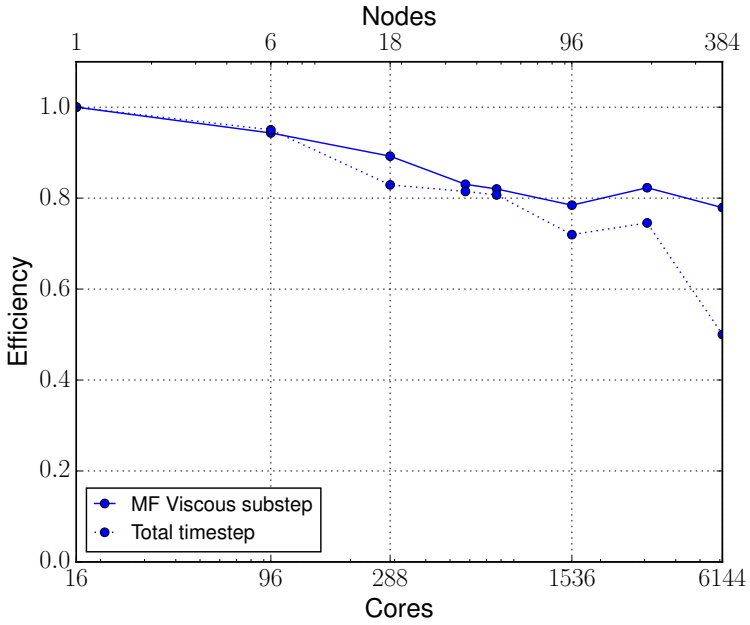
Figure 3.10: Floating-point performance in GFLOPs/s and throughput in MDOFs/s on a multilinear grid managed by `UG`. The curves show the result of variational form evaluation in the viscous substep for a fully loaded node, 2x Intel Xeon E5-2698 v3.

3.3.2 Scalability tests

In this subsection we investigate the scalability of our implementation for a realistic example problem on the BWFORDEV development cluster in Heidelberg. The cluster is equipped with 416 nodes of Intel Haswell E5-2630 v3 16 core processors, 64 GiB/node, connected by QDR Infiniband. Weak and strong scaling is tested on the 3D Taylor-Green vortex problem. The description to the problem is postponed to Section 4.5. We nevertheless want to give some details of the benchmark setup: The fractional step technique is the RIPCS 2.2.3, where the computationally dominant parts are the viscous substep (step 1 in this section) and the pressure Poisson equation (first part of step 2). The remainder are the projection step (second part of step 2) which is here unlike Section 4.5 given by Helmholtz-flux $H(\mathrm{div})$ reconstruction in RT_h^{p-1} and the divergence correction in the rotational formulation (step 3). Both leave only a (block-) diagonal system to solve.

Run times are measured per viscous substep and entire fractional step. Time-stepping is done with the two-stage diagonal implicit Runge-Kutta method from Alexander, Appendix A.5.2. Each nonlinear equation is solved JFNK and flexible GMRes as a linear solver. Here, both the MF and PMF variant of a Gauss-Seidel iteration are tested as subdomain solvers for weak scaling. The solver for the pressure Poisson equation is made from the Conjugate Gradient method with energy error estimation and null space correction, Algorithm 3.4. It is preconditioned by a hybrid AMG-DG. On the DG level, block smoothing per subdomain is carried out with one symmetric Gauss-Seidel sweep of the PMF variant. The subspace to this is spanned by the conforming piecewise linear elements $\mathcal{Q}_1$, the corresponding matrix is rediscretized and a further hierarchy is built from the aggregation based AMG. Number of pre-/post-smoothing steps on all levels are set to one.

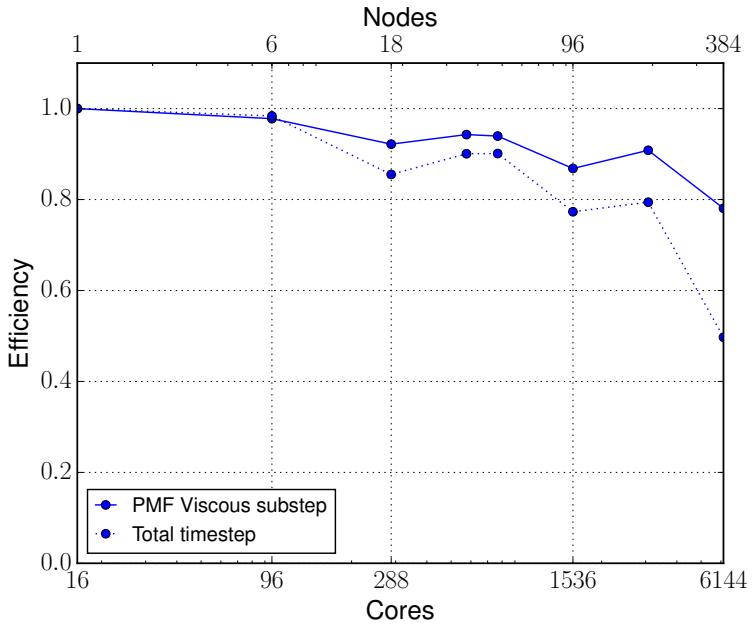
Figure 3.11 and 3.12 show the efficiency of weakly scaling the Taylor-Green vortex problem for $p = 3$ with 16^3 cells/core from 1 to 384 nodes of the cluster. The DOFs per core is ~ 800000, number of unknowns in Raviart-Thomas postprocessing are not counted. The total number of DOFs on 6144 cores in weak scaling is therefore $\sim 5.51 \cdot 10^9$. As our implicit Navier-Stokes solver is essentially made up of local computations and the time step is calibrated to have the Courant number $\mathrm{Cr} \sim 0.2$, we expect the mostly flat efficiency curve observed in our measurements. However at larger problem sizes communication jitter increases as the program starts to span more of the cluster's communication infrastructure which can be seen in the standard deviations from the mean times for both the viscous substep and entire operator splitting. Furthermore we observe a performance degradation for the maximum number of nodes in the total RIPCS which is absent in the viscous substep. The number of unknowns in the internal AMG implementation are repartitioned to one core once the DOFs per processor are too low for further coarsening. Now post to this accumulation, for the largest number of cores (6144) a multigrid hierarchy is still constructed on one rank. This causes the sequential part of the parallel multigrid algorithm to become more pronounced and is therefore visible in the efficiency curve. Successive accumulation to fewer processors was regrettably dysfunctional at the time of benchmarking.



Cores	Viscous min [s]	Viscous mean±std [s]	Total min [s]	Total mean±std [s]
16	33.336	36.151±1.356	41.798	44.672±1.357
96	35.342	35.721±0.826	43.988	44.401±0.914
288	37.362	37.384±0.010	50.413	50.662±0.125
576	40.145	40.165±0.014	51.301	51.442±0.107
768	40.651	40.666±0.006	51.796	52.001±0.133
1536	42.495	43.013±1.784	58.078	59.835±3.345
3072	40.510	40.986±1.178	56.068	57.520±2.218
6144	42.786	44.481±2.941	83.594	87.324±5.222

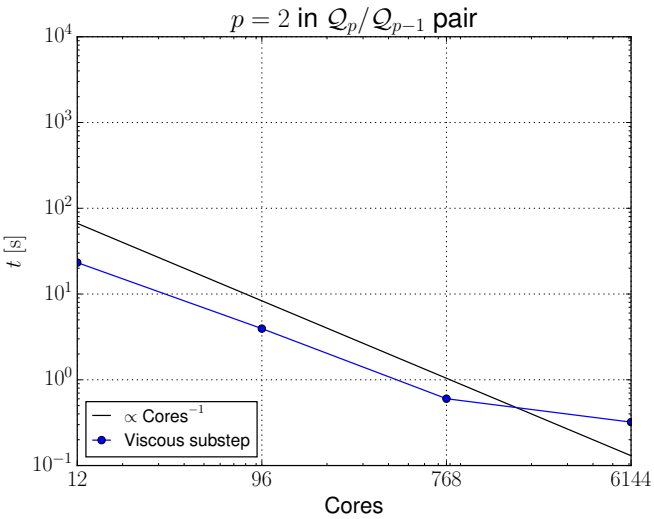
Figure 3.11: Efficiency and run times for the matrix-free viscous substep solver and the entire operator splitting on IWR compute cluster (416 nodes with 2 x E5-2630 v3 each, 64 GiB/node, QDR infiniband). The plot is based on the fastest times, mean values show a large amount of jitter as seen in the table.

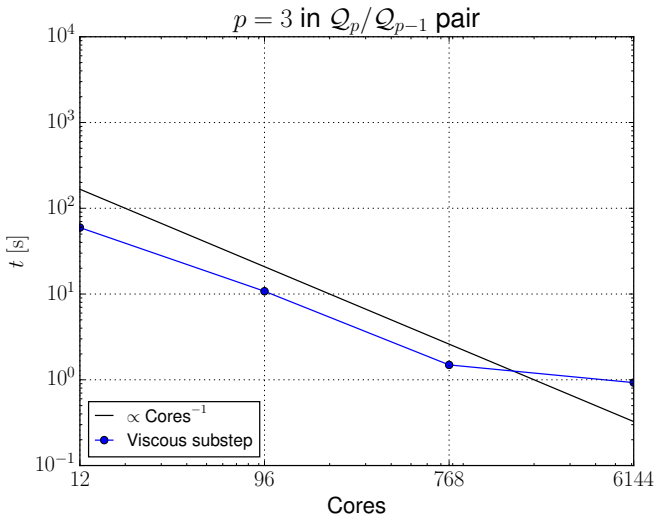
The results of our strong scalability benchmarks are shown in figure 3.13 and 3.14. For the discretization $\mathcal{Q}_p/\mathcal{Q}_{p-1}$ with $p = 2, 3$, we are able to measure scalability from 12 to 6144 cores on up to 384 nodes. Across this range, the number of cells per core shrinks from $24 \times 24 \times 16$ to $3 \times 3 \times 2$, which corresponds to 1602 or 3942 DOFs per core at 6144 cores, respectively. For the pair $\mathcal{Q}_p/\mathcal{Q}_{p-1}$ with $p = 5$ however, we measure scalability from 32 to 6144 cores. In this case the number of cells per core shrinks from $12 \times 12 \times 24$ to $3 \times 3 \times 2$, which corresponds to 13914 DOFs per core at 6144 cores. For these small working sets, scalability suffers mostly because our implementation is currently not able to overlap computation and halo communication, which can be seen in more details in the tables below.



Cores	Viscous min [s]	Viscous mean±std [s]	Total min [s]	Total mean±std [s]
16	33.335	33.463±0.628	42.133	42.265±0.626
96	34.090	34.102±0.007	42.824	42.926±0.069
288	36.163	36.284±0.530	49.268	49.664±0.543
576	35.355	35.527±0.489	46.776	47.054±0.464
768	35.479	36.064±2.392	46.754	47.903±3.762
1536	38.393	39.570±3.833	54.495	56.486±4.889
3072	36.700	36.975±0.772	53.050	53.817±1.093
6144	42.698	44.197±1.980	84.748	87.986±4.400

Figure 3.12: Efficiency and run times for the partially matrix-free viscous substep solver and the entire operator splitting on IWR compute cluster (416 nodes with 2 x E5-2630 v3 each, 64 GiB/node, QDR infiniband). The plot is based on the fastest times, mean values show a large amount of jitter as seen in the table.

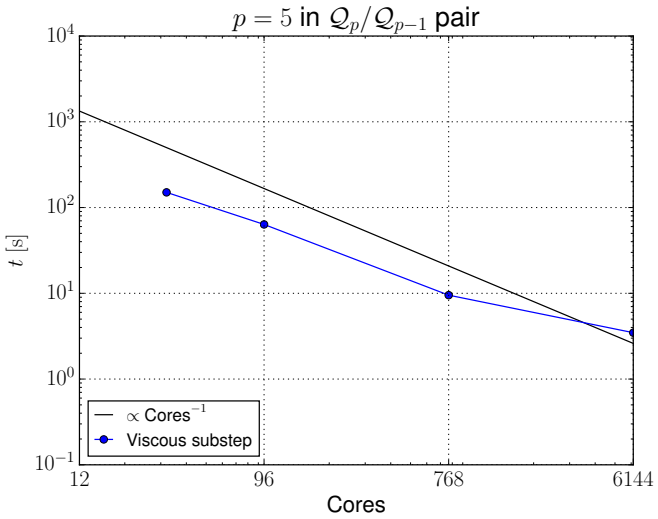




Cores	Viscous min [s]	Viscous η
12	23.259	1.000
96	3.952	0.736
768	0.602	0.604
6144	0.321	0.142

Viscous min [s]	Viscous η
59.564	1.000
10.776	0.691
1.491	0.624
0.927	0.126

Figure 3.13: Run times and efficiencies of strong scaling for the matrix-free viscous substep solver on IWR compute cluster (416 nodes with 2 x E5-2630 v3 each, 64 GiB/node, QDR infiniband). Shown are the polynomial degrees $p = 2, 3$ in the discretization pair $\mathcal{Q}_p/\mathcal{Q}_{p-1}$. The solid black line shows the slope of ideal strong scaling.



Cores	Viscous min [s]	Viscous η
32	150.337	1.000
96	63.522	0.789
768	9.513	0.658
6144	3.475	0.225

Figure 3.14: Run times and efficiency of strong scaling for the matrix-free viscous substep solver on IWR compute cluster (416 nodes with 2 x E5-2630 v3 each, 64 GiB/node, QDR infiniband). Shown is the polynomial degree $p = 5$ in the discretization pair $\mathcal{Q}_p/\mathcal{Q}_{p-1}$. The solid black line shows the slope of ideal strong scaling.

4 Simulations of incompressible fluid flow

The last two chapters were focused on a numerical method for incompressible fluid flow, and its implementation using the Spectral Discontinuous Galerkin Method (SDGM) respective solution of the arising systems. In this chapter we give a description of numerical simulations performed on types of incompressible flow that have been introduced to some extent in Chapter 1. In Section 4.1 we present a setup of two-dimensional mixing layer turbulence. In the next section together with number 4.4, we move on to turbulence in buoyancy driven flows with vertical heating in two and three space dimensions, respectively. In Section 4.3 we are concerned with the flow around a cylinder obstacle, in particular the formation of a turbulent wake behind it. In Section 4.5 we are going to consider the 3D Taylor-Green vortex as an example of decaying Homogeneous Isotropic Turbulence (HIT). DNS results are compared to reference quantities which are available to the public, likewise the behavior of our method in underresolved turbulence. Further and related to that topic, we investigate the spectral distribution of kinetic energy. Mainly we are interested in the appearance of a power law in the inertial subrange followed by a rapid decay of the spectrum. A comparison is made between the numerically determined power exponent and the outcome of K41 theory. Another example situation of turbulence is described in Section 4.6 - the case of boundary layer turbulence between two parallel flat plates. An extension to this configuration is given in Section 4.7 where we propose a model for numerical study of surface renewal (SR) effect in the Atmospheric Boundary Layer (ABL). The proposed model is then analyzed with the methodology summarized in Chapter 1. Movement of coherent structures in surface renewal are detected by structure functions and concentration cross-sections.

4.1 2D Mixing layer turbulence

Mixing layer turbulence arises in the setting of either two fluids at different height with opposing velocities and possibly different densities or a single fluid with such a velocity difference. In the interface across the velocity difference a shear zone is created with high vorticity which is zero away from the interface. The flow is stably stratified with the lighter layer uppermost. However above a critical Reynolds number, small vertical perturbations in the shear zone cause the configuration to mix up and transition to turbulence. From the thin vorticity band several vortex roll-ups are observed which are responsible for the well-known shape of the *Kelvin-Helmholtz instability*. As an incompressible Navier-Stokes problem, mixing layer turbulence is modeled with constant

density and an initial velocity field that exhibits a shear zone. As a two-dimensional flow it has been investigated recently in [SS12, SJL+18] in great detail.

The simulation parameters for the Kelvin-Helmholtz instability are based on the publications mentioned above. As a domain we take $\Omega = (0,1)^2$ with periodic boundary conditions in all directions and no external forcing, $f = 0$. The initial velocity field is another double shear zone with a smooth transition:

$$\begin{aligned} v_0(x) \cdot e_1 &= \begin{cases} \tanh(30(0.25 - x_2)) \,, & x_2 \leq 0.5 \\ \tanh(30(x_2 - 0.75)) \,, & \text{else} \end{cases} \\ v_0(x) \cdot e_2 &= \epsilon \sin(4\pi x_1) \,. \end{aligned} \tag{4.1}$$

The bottommost layer up to $x_2 \approx 0.25$ characterizes a flow in positive x_1-direction with unit magnitude, the intermediate layer $0.25 \lessapprox x_2 \lessapprox 0.75$ a flow in negative x_1-direction with unit magnitude and the uppermost layer is identical to the bottommost one. A small sinusoidal perturbation is added to the vertical component with magnitude $\epsilon = 10^{-2}$. A wavenumber of two with respect to the horizontal extent triggers the most unstable modes within linear perturbation theory, recall Section 1.2.1. We consider further the evolution of two different tracers in this stratification that get advected either in positive x_1-direction or negative x_1-direction. Therefore a passive scalar is chosen with a double-valued initial condition and smooth transition in the shear zones:

$$c_0(x) = \begin{cases} \frac{1}{2}\tanh(30(x_2 - 0.25)) + \frac{3}{2} \,, & x_2 \leq 0.5 \\ \frac{1}{2}\tanh(30(0.75 - x_2)) + \frac{3}{2} \,, & \text{else} \end{cases} \tag{4.2}$$

so that $1 \leq c_0(x) \leq 2$. The fluid parameters are set to $\rho = 1$, $\mu = 10^{-4}$, $\mathrm{Sc} = 1$. The Reynolds number is then $\mathrm{Re} = \frac{1}{\mu}$ and the Peclet number also has the value 10^4. Computations have been done on a 512×512 equidistant rectangular mesh. Polynomial degree for velocity and passive scalar ansatz spaces is $p = 3$ and $p - 1 = 2$ for the pressure. The variant of the discrete solenoidal projection operator is given by pressure Poisson $H(\mathrm{div})$ reconstruction in RT_h^1. The Navier-Stokes equations are advanced in time with the RIPCS. Temporal discretization of the viscous substep is done with Alexander's second order method. The systems arising therein are solved with JFNK where the flexible GMRes method is used. The subdomain solves in the additive Schwarz preconditioner are carried out with the partially matrix-free variant of one symmetric Gauss-Seidel iteration. The initial time step is $\Delta t^1 = 5 \cdot 10^{-4}$, otherwise the time step is calibrated by a fixed $\mathrm{Cr}_{max} = 0.5$ where the Courant number is estimated by means of Equation (A2.14) and the corresponding explanation in Appendix A.2. The mixing layer problem is simulated up to time $T = 20$, sizes in the range $\Delta t \sim 7 \cdot 10^{-4}$ and therefore number of time steps ~ 29000.

Two time snapshots in the Kelvin-Helmholtz instability are displayed in figure 4.1. The left column shows the distribution of the passive scalar during vortex during onset and nicely demonstrates the formation of Kelvin-Helmholtz billows. The bottom row shows the vortex roll-up at later times where sharp fronts of high and low concentration are already nested within each other that are well resolved by the spatial discretization.

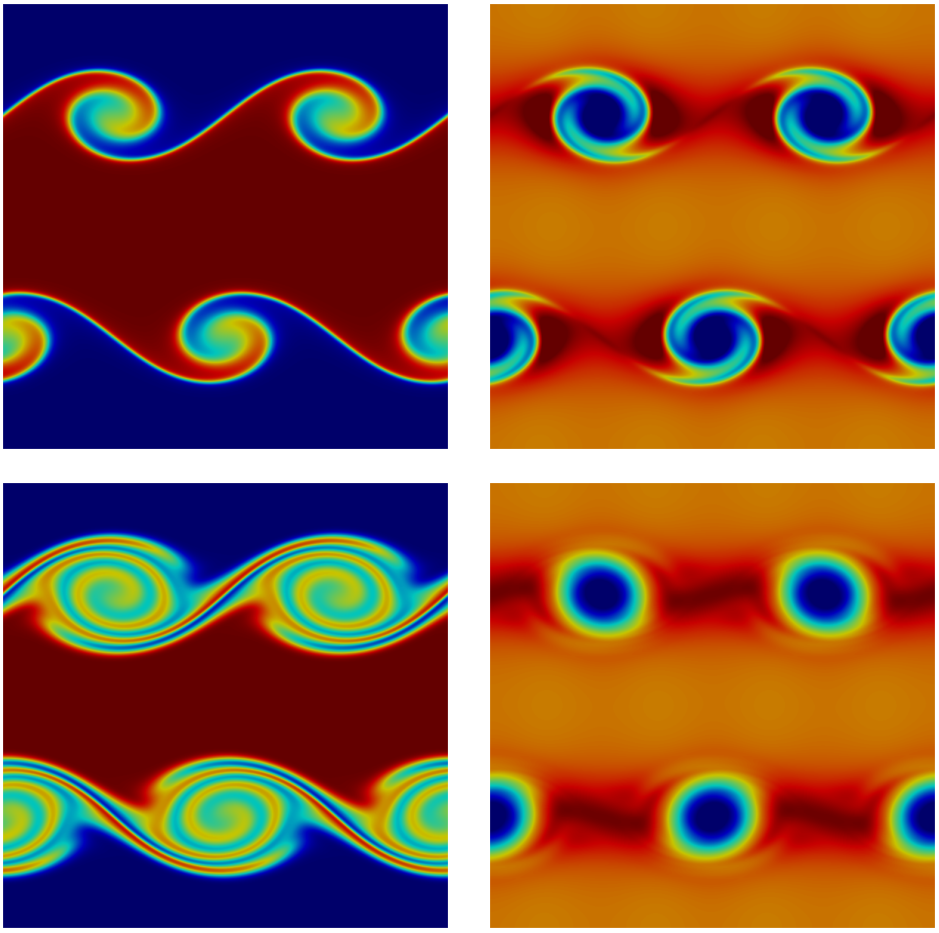


Figure 4.1: Passive scalar distribution and λ_2-criterion of the velocity field in Kelvin-Helmholtz instability. Two different time snapshots grouped by row. Left column shows passive scalar concentration. Right column shows the λ_2-criterion where regions of low values are displayed by bluish colors.

The right column shows the distribution of the λ_2-criterion. It is seen that it is able to capture the rolls in mixing layer turbulence indicated by regions of low values and the ribs which are the tails of the minima. At later times which is not shown here, the vortices then pair to larger and larger vortices until finally one vortex remains. Such an inverse energy cascade from small to large scales is characteristic for two-dimensional high Reynolds number flows. Three-dimensional flows cannot reorganize themselves into large structures.

4.2 2D Rayleigh-Bénard convection

Rayleigh-Bénard convection is modeled within a vertically heated thin cavity. The bottom boundary is hot and the top boundary is cold so the buoyancy effect of rising warm

fluid is balanced by gravity acting downward. Such problems in two dimensions have been studied before for example in [DHOT09, CGS02, ESW14]. If the temperature gradient is in vertical direction, below a critical Rayleigh number a quiescent steady state will be taken. In this case the temperature decreases linearly in height and is solely balanced by the pressure gradient. In fact the horizontally stratified configuration, c.f. Equation (4.3) for the specific choice of boundary conditions below, is a solution for all Rayleigh numbers. However above a critical Rayleigh number this equilibrium solution is not stable and the system will pass over to time-periodic solutions. In this regime the fluid circulates in attached regions between the lower and upper plate that are referred to as steady *Rayleigh-Bénard convection rolls*.

The choice of domain is $\Omega = (0, 8) \times (0, 1)$ with periodic boundary conditions in x_1-direction and Dirichlet boundary conditions at the lower and upper plate. No-slip conditions are imposed for the velocity on the Dirichlet boundary. The temperature is held constant to 0.0 on the top/cold boundary and to 1.0 on the bottom/hot boundary. The flow is driven by the buoyancy forcing term with no additional external forcing, $f = 0$. The analytical solution describing the quiescent steady state is ($d = 2, 3$)

$$v(x) = 0\,, \qquad p(x) = -\frac{1}{2}x_d^2 + x_d - \frac{1}{3}\,, \qquad \theta(x) = 1 - x_d\,. \tag{4.3}$$

There is no convection, a parabolic profile in the pressure that balances the linear temperature decrease. Isosurfaces are parallel to the lower and upper plate.

The simulation is initiated by $v_0(x) = 0$ and a temperature field which is zero apart from a thin layer at the hot boundary. This layer is perturbed in the middle in order to accelerate the onset of instabilities. We set $\mathrm{Ra} = 10^6$, $\mathrm{Pr} = 1$ which gives $\mathrm{Re} = 1000$. Computations have been done on a 320×40 equidistant rectangular mesh. Polynomial degree for velocity and active scalar ansatz spaces is $p = 3$ and $p-1 = 2$ for the pressure. Consequently the pressure dependency in Equation (4.3) is in the ansatz space for the discrete pressure. The temperature profile is contained in the piecewise linear polynomial space Q_h^1, though a polynomial degree equal to the velocity space balances the spatial approximation order in the momentum equation. The variant of the discrete solenoidal projection operator is given by pressure Poisson $H(\mathrm{div})$ reconstruction in RT_h^1. The Navier-Stokes equations are advanced in time with the RIPCS. Temporal discretization of the viscous substep is done with Alexander's second order method. The systems arising therein are solved with JFNK where unpreconditioned GMRes is used. The time step is kept constant at $\Delta t = 10^{-2}$. The Rayleigh-Bénard convection problem is simulated up to time $T = 80$ with 8000 time steps taken.

Two time snapshots of the temperature field are displayed in figure 4.2. Surface Line Integral Convolution (LIC) has been added which is a streamlines visualization technique of the velocity field. Our choice of the Rayleigh number is even a couple of magnitudes larger than the critical value which causes the formation of unstable upward fingers in a unpredictable way as can be seen in the upper part of figure 4.2. The two-dimensional system transitions then to a chaotic motion but eventually reorganizes itself to quasi steady-state convection rolls. This is different to the three-dimensional counterpart as we will see in Section 4.4.

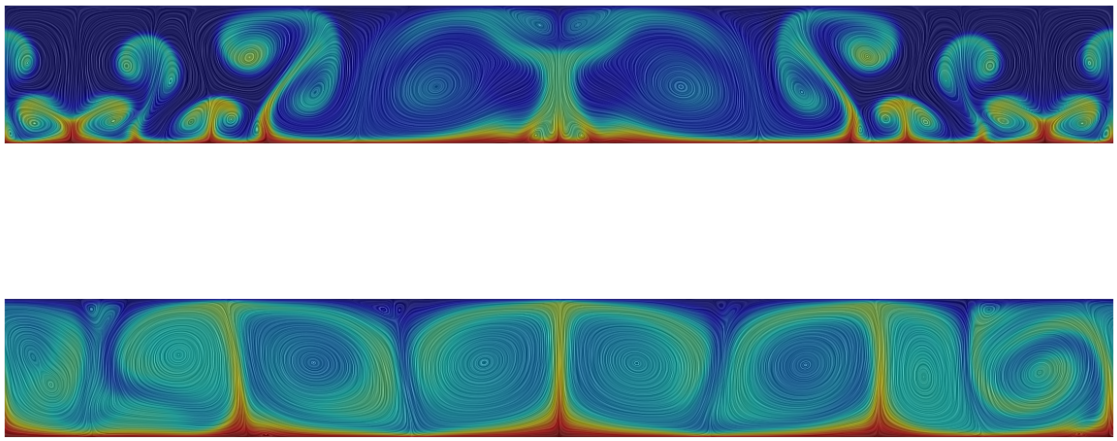


Figure 4.2: Visualization of two-dimensional Rayleigh-Bénard convection with $\mathrm{Ra} = 10^6$ and $\mathrm{Re} = 1000$. Two time snapshots of temperature distribution shaded by Surface LIC with the velocity field. Upper part shows the onset of instabilities by upward rising finger of warm fluid. Bottom part shows the reorganization of the two-dimensional flow to quasi steady convection rolls at later times.

4.3 3D Flow around cylinder

Flow around a cylinder originates from the two-dimensional setting where a disc is placed slightly asymmetric with respect to the middle of the channel. The three-dimensional problem is obtained by spanwise extrusion in x_3-direction and has been investigated in [Nab98, BMT11, Joh05] for example. Configuration with measures of the channel and the cylindrical obstacle are displayed in figure 4.3. No slip boundary conditions are imposed except from the inflow plane where a parabolic profile is prescribed, and except from the outflow plane where the DDN condition is imposed. The flow is only driven by the parabolic profile

$$g(x,t) = \begin{cases} u_{max} \sin\left(\frac{\pi}{8} \min(t,4)\right) \prod_{i=2}^{d} \frac{4x_i(H-x_i)}{H^2} \cdot e_1 \ , & \text{if } x \in \partial\Omega \text{ and } x_1 = 0 \\ 0 \ , & \text{else} \end{cases} \tag{4.4}$$

with no additional forcing, $f = 0$. As an initial velocity field let $v_0(x) = 0$. The Reynolds number $\mathrm{Re} = \frac{\overline{v}D}{\nu}$ is defined via the diameter D of the cylinder and the mean inflow velocity $\overline{v}$ at $x_1 = 0$. $\overline{v}$ can be calculated by the normalized surface integral

$$\begin{aligned} \overline{v} &= \frac{1}{H^2} \int_0^H \int_0^H v_1(0, x_2, x_3) \mathrm{d}x_2 \mathrm{d}x_3 \\ &= u_{max} \left(\frac{4}{H^3} \int_0^H yH - y^2 \mathrm{d}y \right)^2 \end{aligned}$$

$$= \frac{4}{9} u_{max} \ .$$

Together with the diameter $D = 0.1$ we then obtain for the Reynolds number

$$\mathrm{Re} = \frac{2}{45\nu} u_{max} \ . \tag{4.5}$$

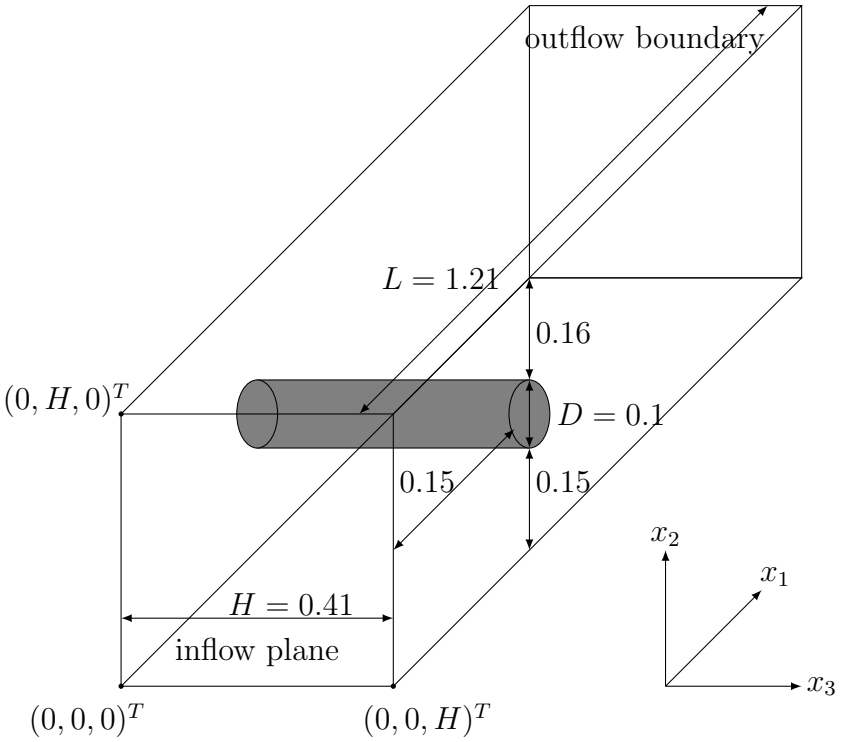


Figure 4.3: Computational setup for the three-dimensional flow around a cylindrical obstacle. Shown are the choice of coordinates and the measures of the channel and obstacle.

For laminar flows a periodic wake behind the obstacle is observed, a phenomenon which is called the *von Kármán vortex street*. In contrast to the two-dimensional setting, there is no vortex shedding behind the cylinder for Re = 100, [Joh05]. If $u_{max} = 2.25$ is chosen, the formula (4.5) further simplifies to $\mathrm{Re} = 1/10\nu$. With $\nu = 2 \cdot 10^{-4}$ we obtain a Reynolds number of 500 which exhibits already irregular patterns in the wake. The mesh of the coarsest level consists of 320 hexahedra and is displayed in figure 4.4. From boundary layer theory, the thickness of linear decrease from maximum velocity to zero is determined by $\sim D/\sqrt{\mathrm{Re}}$ where D is the diameter of the object to circulate around. With our choice of parameters this quantity is equal to 0.00447 which is in the same order of magnitude as 0.0025, the meshing width of the coarse grid on the boundary layer in radial direction. The size of the elements behind the obstacle on the coarsest

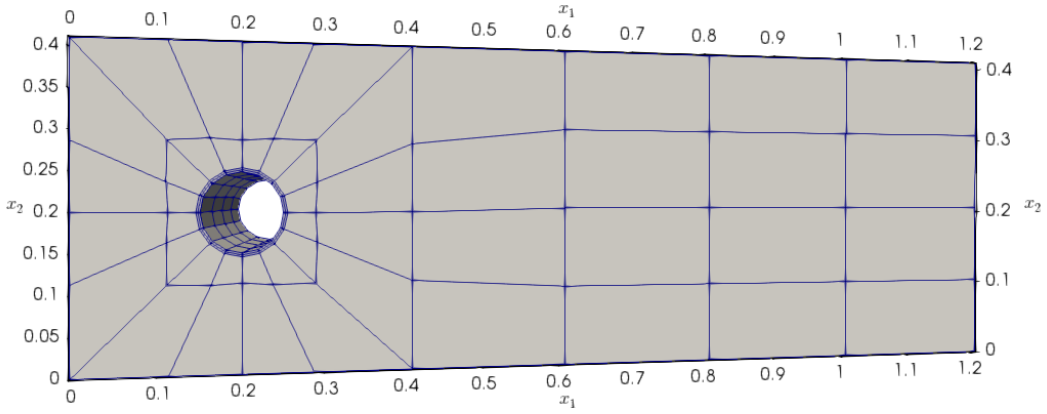


Figure 4.4: Coarse mesh for 3D flow around cylinder which consists of 320 hexahedra. Anisotropic cells have been placed to account for the boundary layer on the obstacle. The mesh is equidistantly extruded with four hexahedra in x_3-direction.

level however is too coarse to resolve the wake zone. For the computations the mesh has been refined three levels so that the number of hexahedra is 163840. This problem tests the handle of complex geometries (i.e. non-affine geometries) in our spectral DG code. Through the grid manager in DUNE the mesh is treated as an unstructured grid and each cell is treated as a multilinear map from the reference element. Global coordinates and Jacobian are calculated by Equations (3.12)- (3.13) and no shortcuts are taken as in an affine transformation. Also the ability of discretization and solvers to handle high aspect ratio cells - already indicated by the coarsest mesh in figure 4.4 - are tested. We take polynomial degree $p = 3$ for the velocity ansatz space and $p - 1 = 2$ for the pressure. The variant of the discrete solenoidal projection operator is given by Helmholtz-flux $H(\mathrm{div})$ reconstruction in RT_h^2. Recall that the Piola transform here is evaluated per quadrature point. Time advancement is realized with the RIPCS and Alexander's second order method within the viscous substep. The equations in the viscous substep are solved with JFNK, as a linear solver flexible GMRes is used. The subdomain solves in the additive Schwarz preconditioner are accomplished with the partially matrix-free (PMF) variant of one Gauss-Seidel iteration. The initial time step is $\Delta t^1 = 5 \cdot 10^{-4}$. The remaining time steps are calibrated by a fixed $\mathrm{Cr}_{max} = 1.0$ where the Courant number is estimated by means of Equation (A2.14) (and the corresponding explanation in Appendix A.2) such that during warm-up of the inflow they also never exceed the maximum $\Delta t_{max} = 5 \cdot 10^{-3}$. Eventually the problem has been simulated up to time $T = 7$.

The resulting flow field at two time snapshots is displayed in figure 4.5. Formation of first tongues in the wake with hairpin vortices at the end can be seen in the upper part. Such regions are characterized by low velocity magnitude as indicated by the cross-section on the right. This snapshot represents the time $t = 2.16$ and confines the

time window of transition from laminar flow to creation of vortices at Re = 500. At later times vortex shedding behind the obstacle is observed which is demonstrated in the bottom part for time $t = 4.76$. Note further that from the boundary of the channel, coherent structures evolve which are also identified as vortex cores by normalized helicity H_n being close to +/– 1 (in red/blue color). Such creation of vortex tubes with high respective low helicity are visible through disturbance of the low-velocity thin boundary layer on the two-dimensional cut, c.f. bottom right of figure. The channel length is chosen 1.21 so that the base drag from the obstacle is still noticeable at the outflow plane. Hence regions of backward-facing velocity vectors with respect to the mean streamwise direction get a correction from the DDN contribution.

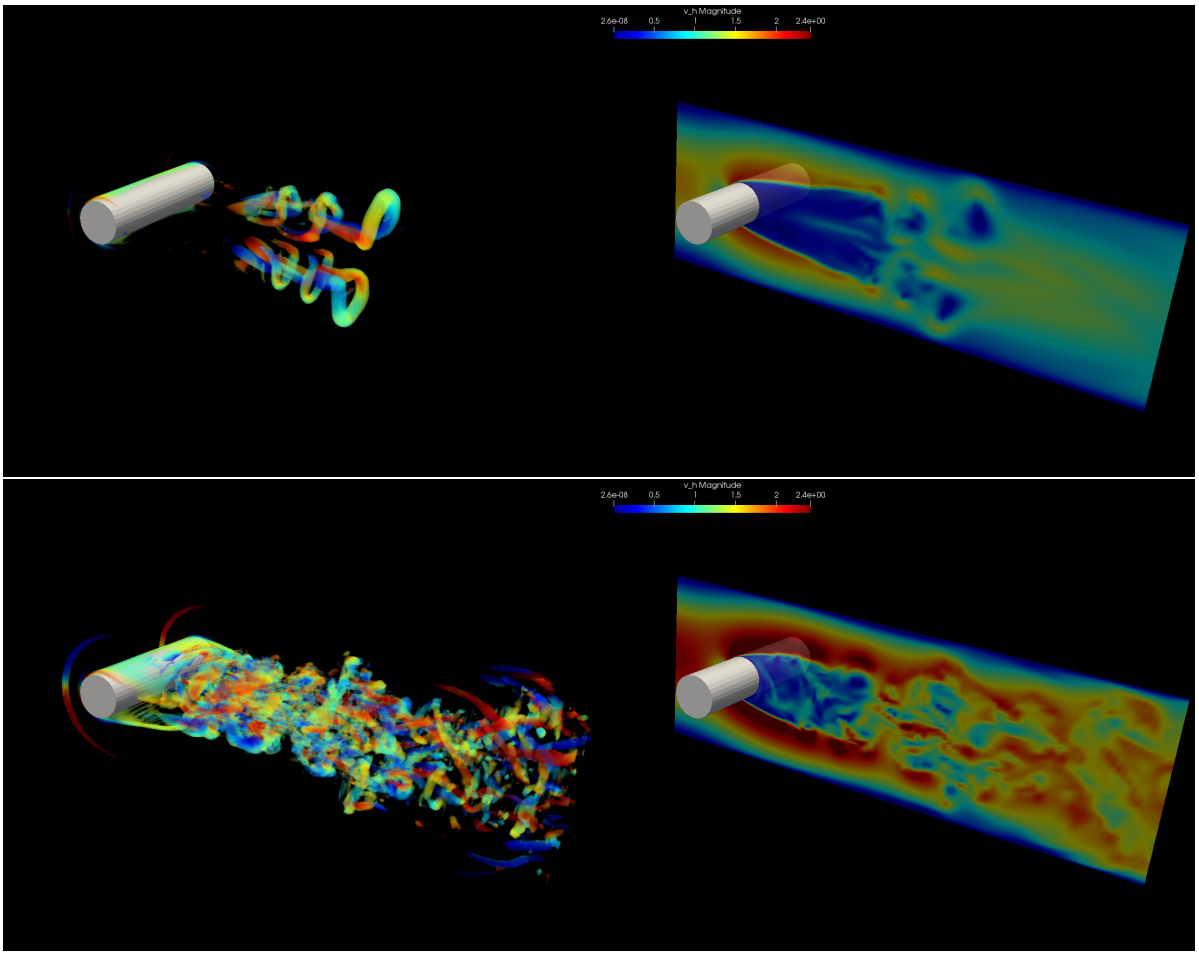


Figure 4.5: Flow field in vortex shedding behind a cylinder obstacle at Re = 500. Shown are the λ_2-criterion colored by the helicity H_n (left) and distribution of velocity magnitude on the cross-section $x_3 = 0.23$ (right). Upper snapshot taken at time $t = 2.16$, lower snapshot at time $t = 4.76$.

Streamlines starting from the inflow plane at time $t = 4.9$ are depicted in figure 4.6. The set of curves are colored by the velocity magnitude and do not only show swirling of the field behind the cylinder but also on the boundary of the channel. An effect that

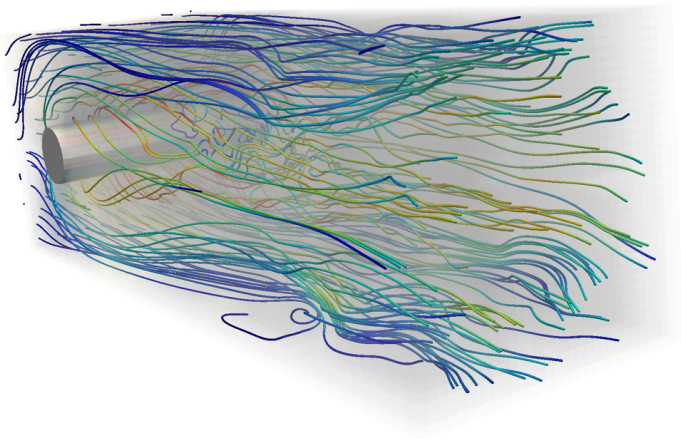


Figure 4.6: Visualization of three-dimensional flow around a cylinder with Re = 500. Shown are the streamlines colored by velocity magnitude at time $t = 4.9$.

we have already recognized above in terms of coherent vortex cores.

4.4 3D Rayleigh-Bénard convection

The three-dimensional Rayleigh-Bénard configuration is constructed from the two-dimensional counterpart by spanwise extension. The domain is $\Omega = (0, 8) \times (0, 8) \times (0, 1)$ with periodic boundary conditions in x_1- and x_2-direction, here x_3 being the vertical direction to the lower and upper plate. No-slip boundary conditions for the velocity are imposed on both plates and the temperature is held constant to 0.0 at the top, constant to 1.0 at the bottom. The initial condition in the Navier-Stokes Boussinesq system are the zero velocity field, the initial temperature perturbation is placed at the bottom center. Rayleigh, Prandtl and thus Reynolds number are adopted from the two-dimensional problem. Computations have been done on a $105 \times 105 \times 14$ equidistant cuboid mesh. Polynomial degree for velocity and active scalar ansatz spaces is $p = 3$ and $p - 1 = 2$ for the pressure. The variant of the discrete solenoidal projection operator is given by pressure Poisson $H(\mathrm{div})$ reconstruction in RT_h^1. The temporal evolution in the Navier-Stokes subproblem is realized with the RIPCS and Alexander's second order method within the viscous substep. The appearing systems are solved with JFNK and the linear solver from the two-dimensional configuration is utilized. Time steps are kept constant at $\Delta t = 5 \cdot 10^{-3}$ and the simulation has been carried out with 30000 time steps up to $T = 150$.

The horizontal cross section of the temperature field for three time snapshots at height $x_3 = 0.85$ are displayed in figure 4.7. As in the two-dimensional case the onset of insta-

bilities is caused by upward rising fingers of warm fluid. After transition to turbulence, when the system has reached a statistical equilibrium, large islands of warm fluid (in red color) moving to the top that are separated by sharp thin valleys of cold fluid (in blue color) are visible in all three time snapshots collected in figure 4.7. In contrast to two dimensions those observed patterns evolve chaotically in time and no quasi steady-state convection rolls are recaptured.

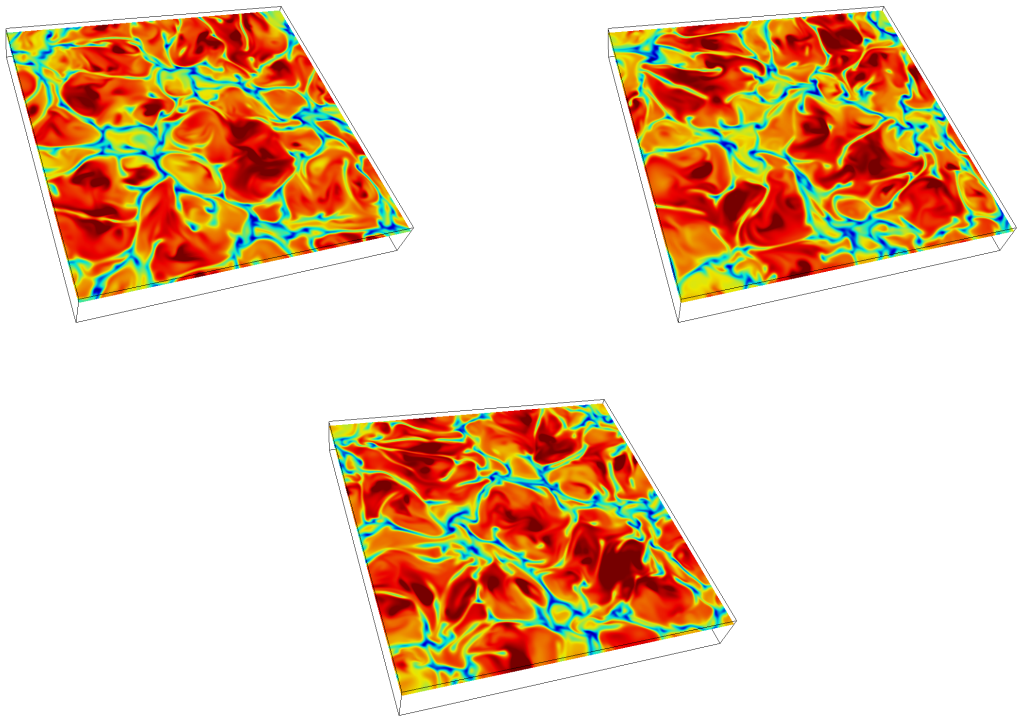


Figure 4.7: Visualization of three-dimensional Rayleigh-Bénard convection with Ra = 10^6 and Re = 1000. Shown is the temperature field on a horizontal cross section for three time snapshots at height $x_3 = 0.85$. All three time snapshots have been taken after the system has reached a statistical equilibrium in turbulence. Visible are patterns resembling to large islands of warm fluid (in red color) separated by sharp thin valleys of cold fluid (in blue color) that evolve chaotically in time.

4.5 3D Taylor-Green vortex

The three-dimensional Taylor-Green vortex is aimed at testing the accuracy and performance of high-order methods in a DNS. The flow starts from a simple large scale initial condition. In the early phase it undergoes vortex stretching until laminar breakdown before the maximum dissipation of the fluid is reached. The flow then transitions

to turbulence followed by a decay phase of eventually *Homogeneous Isotropic Turbulence* (HIT). The problem originates from [TG37] where classes of sinusoidal fields were considered as an initial condition that satisfy the continuity equation $\nabla \cdot v = 0$. It has been proposed as a reference benchmark since the first edition of the international workshop on High-Order CFD Methods (HiOCFD) and is quoted in [Tay13], C3.5, for instance. Recall the fundamentally different two-dimensional Taylor-Green vortex which possesses the exact solution (2.74) that describes a steady in space, exponential decay in time, of the initial vortex distribution.

The simulation domain is $\Omega = (-\pi L, \pi L)^3$ with periodic boundary conditions in all directions and no external forcing, $f = 0$. The initial flow field is given by

$$\begin{aligned} v_0(x) \cdot e_1 &= V_0 \sin\left(\frac{x_1}{L}\right) \cos\left(\frac{x_2}{L}\right) \cos\left(\frac{x_3}{L}\right) \\ v_0(x) \cdot e_2 &= -V_0 \cos\left(\frac{x_1}{L}\right) \sin\left(\frac{x_2}{L}\right) \cos\left(\frac{x_3}{L}\right) \qquad (4.6) \\ v_0(x) \cdot e_3 &= 0 \end{aligned}$$

$$p_0(x) = p_0 + \frac{\rho_0 V_0^2}{16} \left(\cos\left(\frac{2x_1}{L}\right) + \cos\left(\frac{2x_2}{L}\right)\right) \left(\cos\left(\frac{2x_3}{L}\right) + 2\right) . \qquad (4.7)$$

The Reynolds number of the flow here is defined as $\mathrm{Re} = \frac{\rho_0 V_0 L}{\mu}$. As in the references [Tay13, LMW12] we set $L = 1,\ V_0 = 1,\ \rho_0 = \rho = 1,\ p_0 = 0,\ \mathrm{Re} = 1600$. Computations have been done on a series of uniformly refined, equidistant cuboid meshes. We present here only results for the polynomial degree $p = 3$ of the velocity ansatz space and $p - 1 = 2$ of pressure. The variant of the discrete solenoidal projection operator is given by pressure Poisson $H(\mathrm{div})$ reconstruction in RT_h^1. The incompressible Taylor-Green vortex problem is advanced in time with the RIPCS and Alexander's second order method in the viscous substep. The arising systems therein are solved with JFNK and flexible GMRes as a linear solver. Subdomain solves on the equidistantly isotropic meshes here are the matrix-free variant of one Gauss-Seidel iteration.

The finest mesh for the computation with the $\mathcal{Q}_3/\mathcal{Q}_2$ discretization consists of 128^3 cuboid cells. Based on the convective time unit $t_c = \frac{L}{V_0}$, the simulation ranges over the time interval $t_0 = 0$ up to $T = 30t_c$ which is by $10t_c$ longer than the suggested reference duration. Time steps are kept constant at $\Delta t = 5 \cdot 10^{-3}\, t_c$ yielding an estimated Courant number of $\mathrm{Cr}_{max} \sim 0.25$ around $t \sim 8t_c$. In total 6000 time steps are taken with about $4.6 \cdot 10^8$ spatial DOFs partitioned to $16^3 = 4096$ processors. This DNS amongst other runs are computed on the BWFORDEV development cluster in Heidelberg. Figure 4.8 shows the evolution of the resulting velocity field. Arising turbulent structures are captured by the Q-criterion colored with the normalized helicity H_n. The upper left part shows the vortex distribution given by initial condition (4.6). The upper right picture shows the flow field at $t = 9t_c$, around the maximum dissipation rate of the fluid, where small scale structures start to develop. Such structures are identified as vortex cores with helicity H_n being close to $^+/_-$ 1 (depicted in red/blue color) and are further visible in bottom left snapshot at time $t = 12t_c$. Both snapshots demonstrate the phase of decaying homogeneous turbulence for $t \gtrsim 9t_c$ yet here non isotropic. Finally the bottom right part visualizes decaying HIT at time $t = 25t_c$.

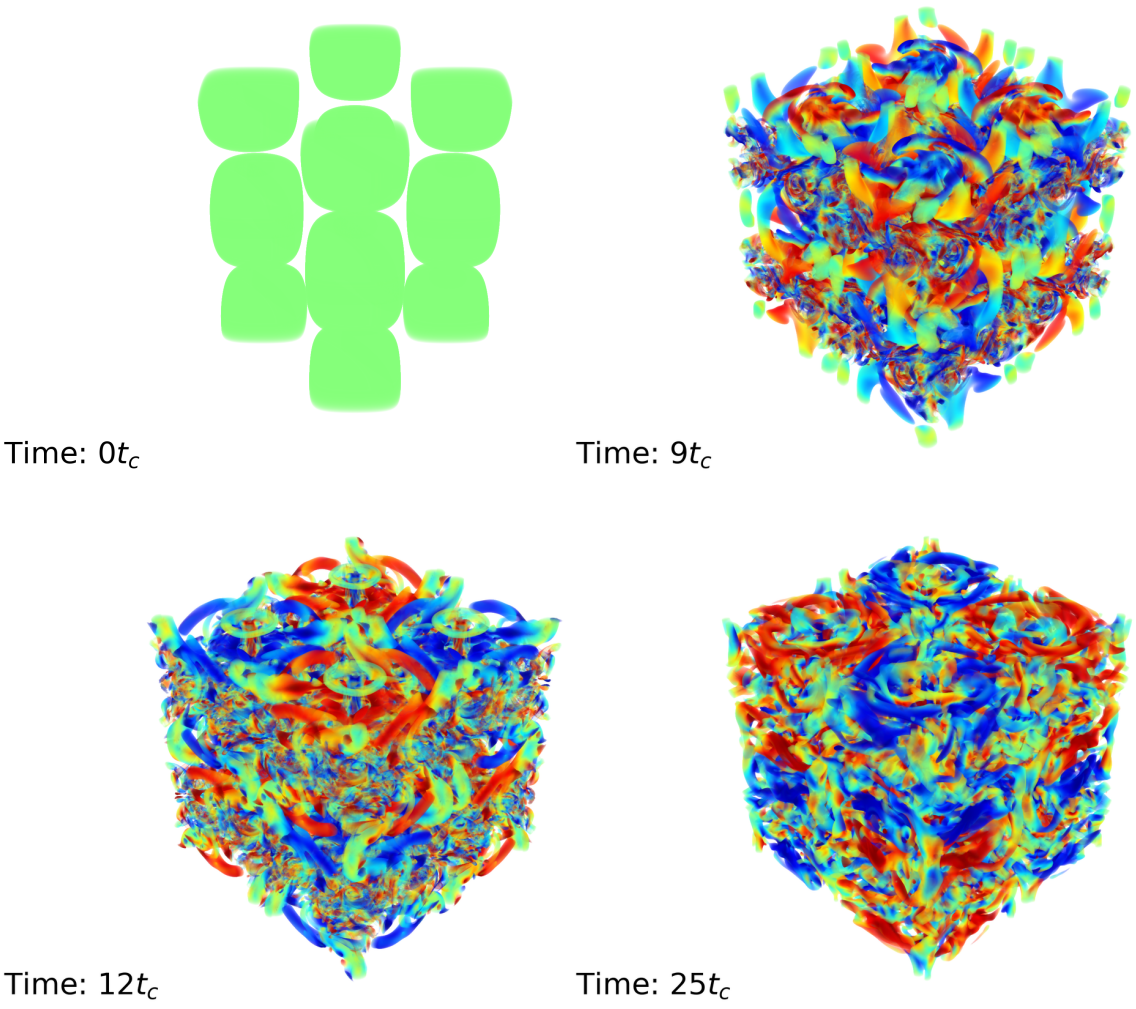


Figure 4.8: Visualization of the three-dimensional Taylor-Green vortex at $\mathrm{Re} = 1600$ on the finest mesh level. Utilized are the Q-criterion colored by the helicity H_n. Added are also the times of the corresponding snapshots.

We now compare the results to the reference values that contain the temporal evolution of the

- kinetic energy $E(t) = \frac{1}{\rho_0|\Omega|}\frac{1}{2}(\rho v, v)_{0,\Omega}$, equal to Equation (1.59) in for incompressible fluids,
- dissipation rate $\epsilon(t) = \frac{\nu}{|\Omega|}(\nabla v, \nabla v)_{0,\Omega}$,
- enstrophy $\mathcal{E}(t) = \frac{1}{\rho_0|\Omega|}\frac{1}{2}(\rho\zeta, \zeta)_{0,\Omega}$ which differs for incompressible fluids from Equation (1.21) by normalizing with the integration volume.

The reference solution was obtained with a dealiased pseudo-spectral code run on a 512^3 grid, time integration was performed with a low-storage three-step Runge-Kutta scheme

an a time step of $\Delta t = 10^{-3}\, t_c$, but only up to time $t = 20 t_c$. A comparison with those three quantities is presented in figure 4.9. The values computed on the finest mesh give a good agreement with the reference. The resulting kinetic energy and dissipation rate curves are colored in green and look almost indistinguishable with respect to the blue reference curves. Further those quantities are connected through the relation (1.31) which holds on the continuous level. Therefore deviations from the equalities indicate discrete dissipation and divergence errors in a numerical scheme. Differences in our results over time between $\epsilon(t)$ and $-\frac{\mathrm{d}E(t)}{\mathrm{d}t}$ computed with the central difference quotient in the interior, are similar to the reference dissipation rate. A zoomed-in plot around the maximum amount of dissipation can be found in the top right. Besides the L^2-error over time $t_0 = 0$ until $30 t_c$ of the computed $\epsilon(t) - 2\nu\mathcal{E}(t)$ on the finest level is in the order of $9.077 \cdot 10^{-6}$ where the integral is approximated by the trapezoidal rule.

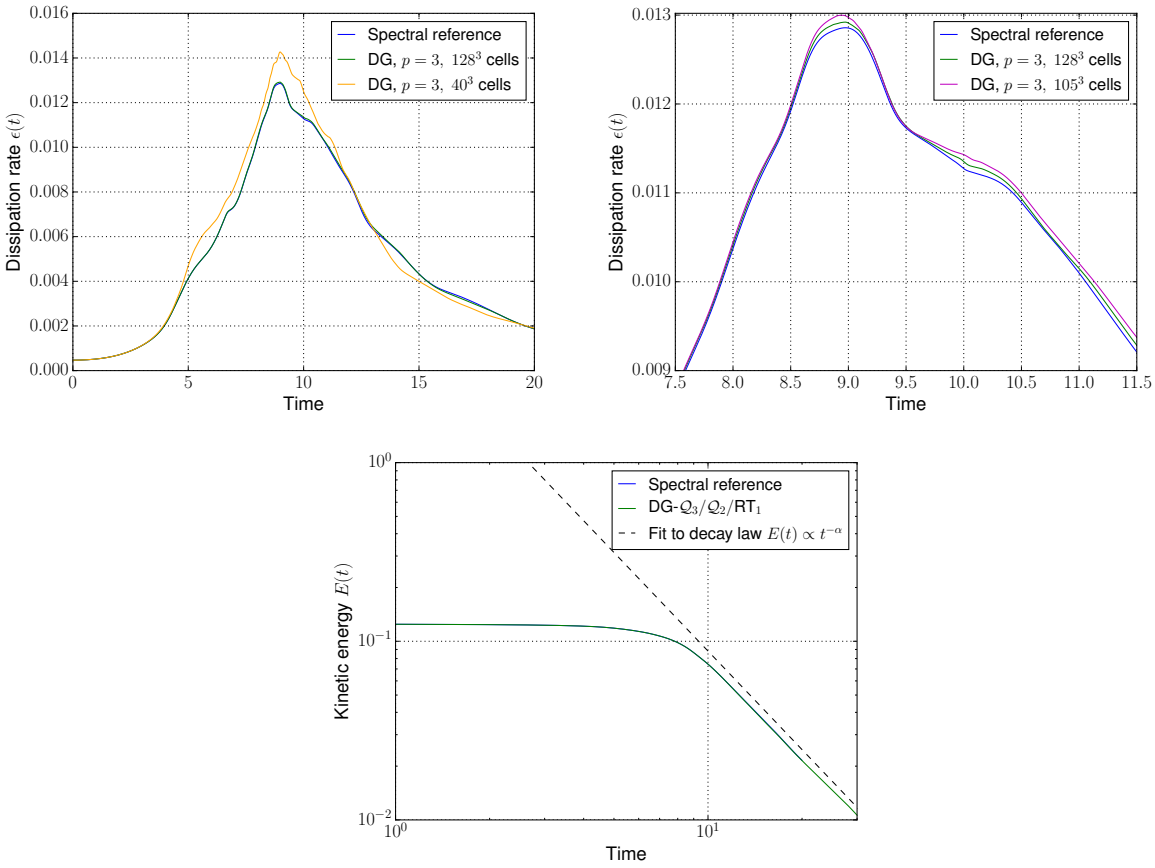


Figure 4.9: Evolution of kinetic energy and dissipation rate of the Taylor-Green vortex at $\mathrm{Re} = 1600$. Shown are results for the $\mathcal{Q}_3/\mathcal{Q}_2$ space on various mesh levels together with the reference spectral code.

A typical evolution of the kinetic energy in decaying turbulence represented in a log-log plot is displayed at the bottom of figure 4.9. The curve exhibits a plateau-like progression up to the point around maximum dissipation followed by a decay law of the

form (1.63).

The behavior of kinetic energy, dissipation and enstrophy under h-refinement constitutes the first part of investigation on our code with respect to accuracy in underresolved turbulence. Runs have been performed with $\{40, 64, 128\}$ cells per direction and time step sizes $\{2 \cdot 10^{-2}\, t_c, 1 \cdot 10^{-2}\, t_c, 5 \cdot 10^{-3}\, t_c\}$. A previous run with 105 cells per direction and a time step size $5 \cdot 10^{-3}\, t_c$ from [PMB18] is also taken into account. The dissipation curve for 40 cells per direction is found together with the 128 cells per direction and reference curve in figure 4.9. The progression for 64 cells per direction has been omitted in the plot for sake of clarity. It can be seen that the solution is over-dissipative on the coarsest level as expected, but the curve already captures well the shape of the fine resolution. On the top right a zoom into the vicinity of maximum dissipation is taken where the result from 105^3 cells is added in purple color. Grid convergence towards the reference is expected as the green curve is confined by the blue and purple one.

Now let us discuss computed energy spectra from the Taylor-Green vortex. An approximation quality of a CFD method is to at first reproduce the schematic plot of the spectral energy distribution in figure 1.5 caused by the three regimes in a turbulent flow. In explicit the curve should exhibit an integral range at smallest wavenumbers, a power law in wavenumbers representing the inertial range and a rapid decay at smallest wavenumbers used for the computation. Secondly a desirable outcome is the numerical verification of the 5/3 exponent in K41 theory. One of the first publications that measured the kinetic spectrum numerically was [BMO$^+$83] who performed DNS on the Taylor-Green vortex in a set of Reynolds numbers including 1600 up to 3000. And [Bra91] who further examined the Taylor-Green vortex at Reynolds number 5000. [GB12] gives a detailed investigation of high-order methods in underresolved turbulence applied to the Taylor-Green vortex at Reynolds numbers 800 and 1600. Computation of the kinetic energy spectrum has been included to the collection of reference values since the 4th edition of HiOCFD [Tay16].

We have characterized turbulent flows by a broad range of spatial and temporal scales. A resolution of all fine scales in a DNS may thus become infeasible as the total number of unknowns depends on the Reynolds number through the power law presented in Equation (1.72). Robust behavior of high-order methods in underresolved turbulence, or in treatment of fractions of finest scales not captured, is of interest and advantage - even when used in a DNS, also for possible cost reasons. [GB12] mentions high-order methods in underresolved turbulence to be unstable and presents necessary stabilizations that are applied to the Taylor-Green problem. One of them is exact integration of the nonlinearity which is referred therein as overintegration to achieve dealiasing. Recall that our choice of quadrature points delivers exact quadrature for the variational forms on affine cells. The distribution of spectral energy should ideally be a decaying power law in the numerically resolved inertial subrange, smaller scales below the mesh resolution should be suppressed as if they belong to the dissipative subrange.

Figure 4.10 contains a collection of computed kinetic energy spectra over various times and for a fixed time under h-refinement. Starting from a large scale initial condition where modes are settled at low wavenumbers, small scale structures develop. This is seen

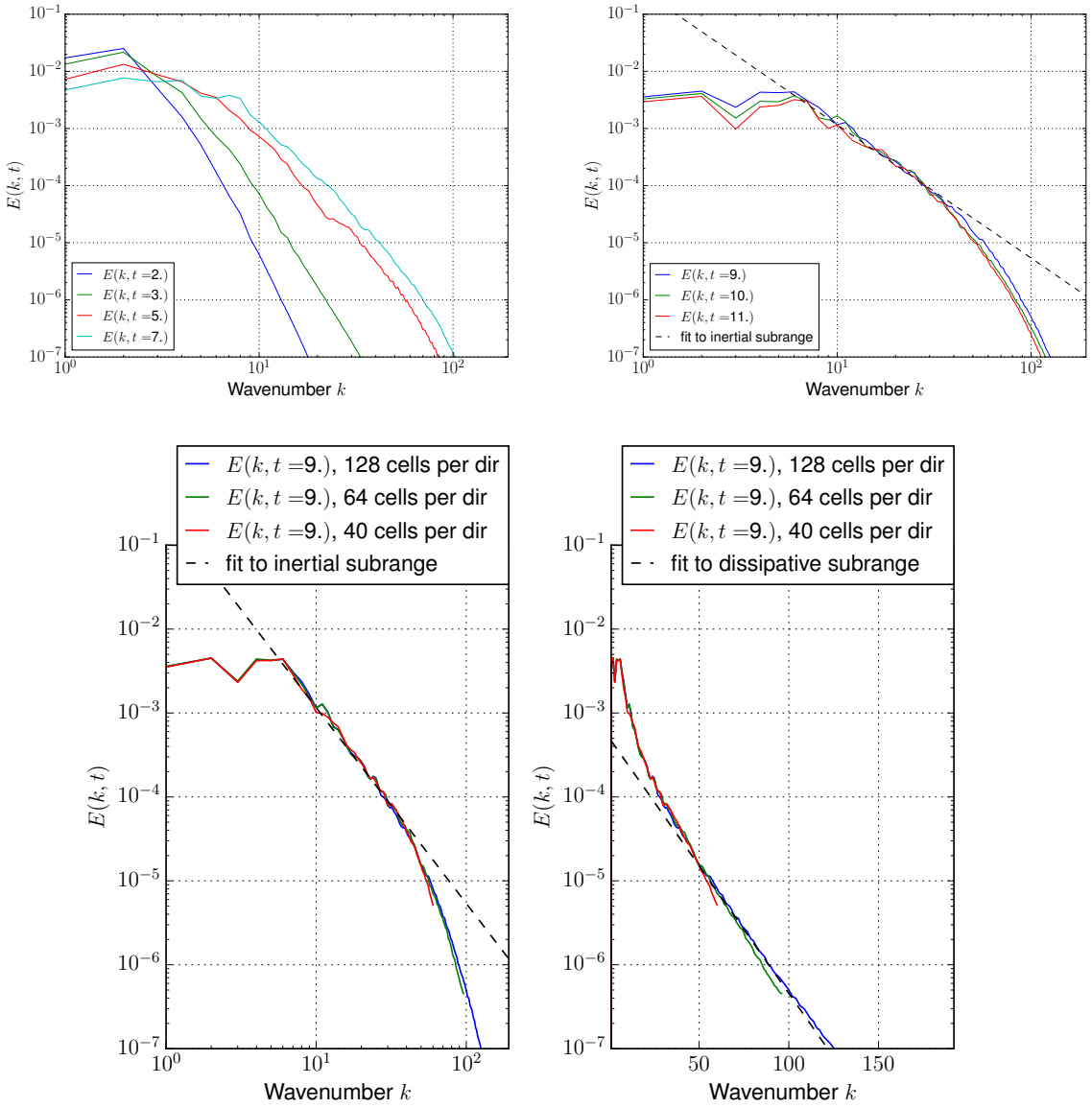


Figure 4.10: Plots of kinetic energy spectra: Top left shows early build-up. Top right shows decaying homogeneous turbulence yet here non isotropic. The second row shows grid convergence for the spectral energy distribution at time $t = 9t_c$ with fits to the inertial and dissipative subrange.

for the curves $E(k,t)$ at times $t \in \{2t_c, 3t_c, 5t_c, 7t_c\}$ in the top left. In the following the blue curve represents the spectrum at time $t = 9t_c$ for the finest mesh resolution of 128 cells per direction. A fit to the inertial subrange added in the top right part demonstrates the desired behavior of a power law which is clearly separated by progression within the integral and dissipative subrange. Note that the curves taken at times $t \in \{10t_c, 11t_c\}$ do not change significantly. They are slightly reduced towards the bottom through decrease

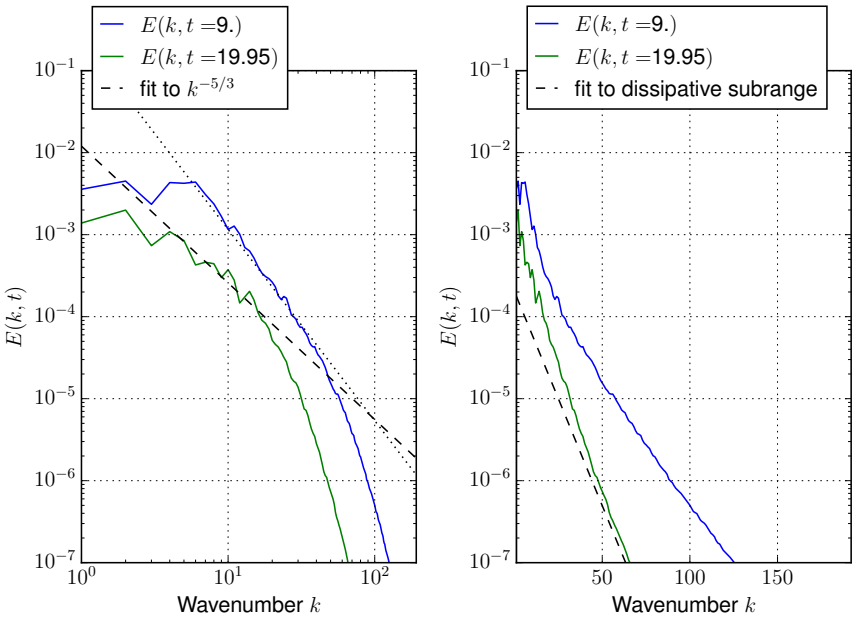


Figure 4.11: Plots of kinetic energy spectra continued: The green curve shows decaying HIT at later times.

of dissipation $\epsilon(t)$ which is in agreement with Equation (1.60). The reference spectra are typically measured around $t \sim 9t_c$ in the community. The exponent for the inertial subrange fit however is $-7/3$ and thus steeper than $-5/3$ from Kolmogorov. This is not a contradiction to theoretical considerations that allow exponents up to $-8/3$, [Fri95], and consistent with the spectrum taken from HiOCFD4. Figure 4.12 reproduces the plot of this spectrum found in [Tay16] and adds our result at time $t = 9t_c$ together with lines in proportion to the power law with exponents $-5/3$ and $-7/3$. Both curves do not differ significantly and show a decay with exponent $-7/3$ in the inertial subrange that is caused by decaying homogeneous turbulence yet here non isotropic - as already indicated by the time snapshots at $t = 9t_c$ and $t = 12t_c$ in figure 4.8 above.

The second row of 4.10 shows the spectral energy distribution at time $t = 9t_c$ under h-refinement. Largest scales that are resolved by all meshes considered, give the same progression at smallest wavenumbers. There are no oscillations around the shorter inertial subrange at coarser levels. Moreover the higher modes below each discretization resolution are suppressed by an exponential drop as can be seen in a log-lin plot of the same range on the right half. The lower the accuracy is, preferably the more corresponding wavenumbers get dissipated in our grid convergence results.

Finally, figure 4.11 shows decaying HIT at later times $\sim 20t_c$ where a power spectrum with exponent close to $-5/3$ is observed. This is demonstrated by the dashed line in the log-log plot. For comparison we have added the energy spectrum at time $t = 9t_c$ in blue color from the plots above with its belonging fit to the inertial subrange in dotted linestyle.

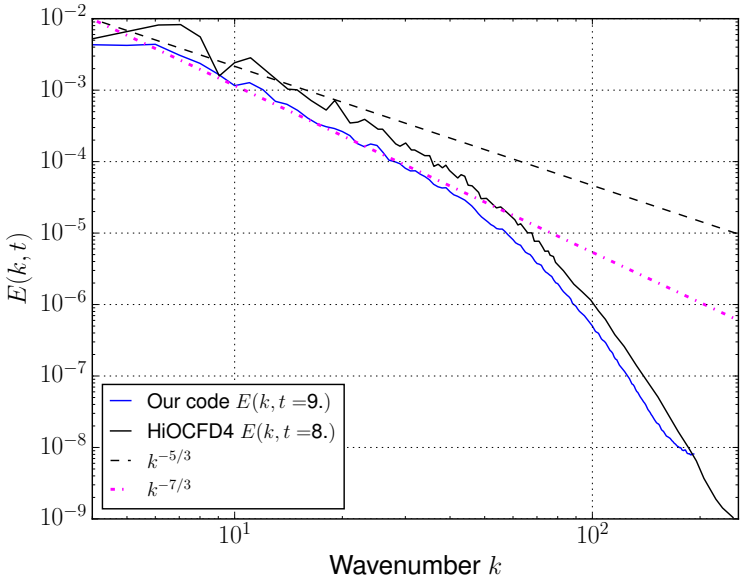


Figure 4.12: Kinetic energy spectrum at time $t = 8$ taken from the problem description to the 4th International Workshop on High-Order CFD Methods. In addition: A spectral energy distribution of our code calculated at time $t = 9$, dashed and dash-dotted lines showing $k^{-5/3}$ and $k^{-7/3}$-law, respectively.

4.6 3D Turbulent channel flow

4.6.1 Modeling

The 3D turbulent channel flow is strictly speaking a flow between two in streamwise and spanwise infinitely extended plates. The flow is driven by a pressure gradient in the streamwise direction. In order to understand the modeling of the problem we refer for the following to [TL72]. The idea is based on splitting up the velocity and pressure into a mean part $(\bar{v}, \bar{p})$ and a fluctuating part $(\tilde{v}, \tilde{p})$, c.f. Section 1.2.1. The mean part of the turbulent profile is considered to be a time-averaged steady quantity

$$\begin{aligned} \langle v \rangle_T = \bar{v}(x) &= \frac{1}{T} \int_{\bar{t}}^{\bar{t}+T} v(x,t) \mathrm{d}t \\ \langle p \rangle_T = \bar{p}(x) &= \frac{1}{T} \int_{\bar{t}}^{\bar{t}+T} p(x,t) \mathrm{d}t \ . \end{aligned} \tag{4.8}$$

Then the fluctuating part has a vanishing temporal average:

$$\frac{1}{T} \int_{\bar{t}}^{\bar{t}+T} \tilde{v}(x,t) \mathrm{d}t = \frac{1}{T} \int_{\bar{t}}^{\bar{t}+T} v(x,t) - \bar{v}(x) \mathrm{d}t = \bar{v}(x) - \bar{v}(x) = 0 \ , \tag{4.9}$$

similar for the pressure. Now insert this decomposition first into the mass conservation equation. This gives

$$\nabla \cdot \bar{v} = 0 = \nabla \cdot \tilde{v} \ . \tag{4.10}$$

In contrast to (1.37) we do not neglect the quadratic term in the momentum equation. Primarily we have

$$-\mu\Delta\bar{v} - \mu\Delta\tilde{v} + \rho\left[(\bar{v}\cdot\nabla)\bar{v} + (\bar{v}\cdot\nabla)\tilde{v} + (\tilde{v}\cdot\nabla)\bar{v} + (\tilde{v}\cdot\nabla)\tilde{v}\right] + \nabla\bar{p} + \nabla\tilde{p} = 0\,.$$

Since $\nabla\cdot\tilde{v} = 0$ we can write the last term in the expansion of the convective term as $(\tilde{v}\cdot\nabla)\tilde{v} = \nabla\cdot(\tilde{v}\otimes\tilde{v})$. Now we do the temporal averaging $1/T\int_{\bar{t}}^{\bar{t}+T}\mathrm{d}t$ and exchange order with the spatial derivatives. By construction,

(1) terms with only the background velocity remain unchanged

(2) all terms that are linear with respect to the fluctuation are zero.

This simplifies to

$$-\mu\Delta\bar{v} + \rho(\bar{v}\cdot\nabla)\bar{v} + \rho\nabla\cdot\langle\tilde{v}\otimes\tilde{v}\rangle_T + \nabla\bar{p} = 0\,. \tag{4.11}$$

For simplicity the subscript for temporal averages is to be dropped from this point on. If $\tilde{v}_i$ and $\tilde{v}_j$ for $i \neq j$ are independent (uncorrelated) on each other, then

$$\langle\tilde{v}_i\tilde{v}_j\rangle = \langle\tilde{v}_i\rangle\langle\tilde{v}_j\rangle = 0\,.$$

But if they were correlated, the average does not have to be zero. Equation (4.11) is the mean momentum equation averaged over time.

Now let us return to the specific problem of a turbulent flow in a channel. The turbulent velocity profile takes the form $\bar{v} = \bar{v}_1(x_2)e_1$. A coordinate system is used where the x_1-direction is the streamwise direction, the x_2-direction is the wall-normal direction and the x_3-direction is the spanwise direction. δ denotes the half channel

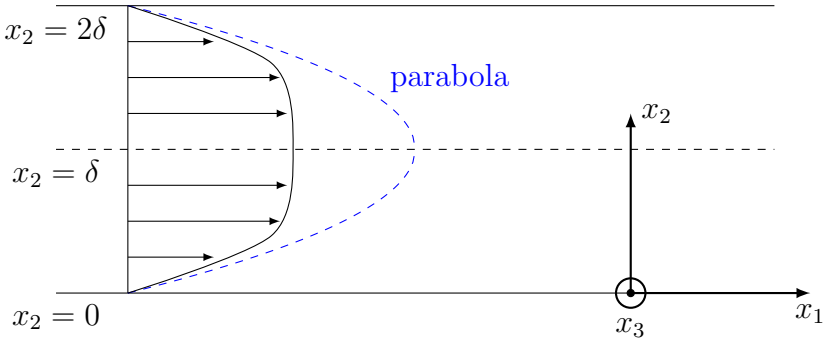


Figure 4.13: Cross-section of the 3D channel flow perpendicular to the spanwise direction. Shown are the definition of δ, the turbulent and laminar velocity profile in streamwise direction.

height. The turbulent profile satisfies $\bar{v}_1(x_2 = 0) = 0$ and $\bar{v}_1(x_2 = 2\delta) = 0$ and by symmetry with respect to the middle plane

$$\left.\frac{\mathrm{d}\bar{v}_1(x_2)}{\mathrm{d}x_2}\right|_{x_2=\delta} = 0\,.$$

It is much flatter than the laminar profile which would be parabolic at the same Reynolds number. At the wall no-slip boundary conditions are imposed. All this information is inserted into the time-averaged mean momentum equation (4.11):

$$\begin{aligned}\rho\left[\bar{v}_1\partial_{x_1}\bar{v}_1+\bar{v}_2\partial_{x_2}\bar{v}_1\right] &= -\partial_{x_1}\bar{p}+\mu\Delta\bar{v}_1-\rho\partial_{x_1}\langle\tilde{v}_1^2\rangle-\rho\partial_{x_2}\langle\tilde{v}_1\tilde{v}_2\rangle\\ \rho\left[\bar{v}_1\partial_{x_1}\bar{v}_2+\bar{v}_2\partial_{x_2}\bar{v}_2\right] &= -\partial_{x_2}\bar{p}+\mu\Delta\bar{v}_2-\rho\partial_{x_1}\langle\tilde{v}_1\tilde{v}_2\rangle-\rho\partial_{x_2}\langle\tilde{v}_2^2\rangle\ .\end{aligned}$$

In this flow we have no variation in the x_1-direction apart from the pressure and $\bar{v}_2=0$. It remains

$$\begin{aligned}0 &= -\partial_{x_1}\bar{p}+\mu\partial_{x_2}^2\bar{v}_1-\rho\partial_{x_2}\langle\tilde{v}_1\tilde{v}_2\rangle\\ 0 &= -\partial_{x_2}\bar{p}-\rho\partial_{x_2}\langle\tilde{v}_2^2\rangle\ .\end{aligned}$$

By further manipulation of these equation one can derive the important relations

- $$\rho u_\tau^2=\mu\partial_{x_2}\bar{v}_1=\tau_w \tag{4.12}$$

where τ_w is the *mean shear stress at the wall* and u_τ is the *friction velocity*. The mean shear stress and friction velocity can be numerically computed with the viscous force exerted at the surface, $F=\mu n\times\zeta$. At the bottom wall $n=-e_2$ and the mean force is

$$\bar{F}=-\mu e_2\times\bar{\zeta}=-\mu\bar{\zeta}_3e_1=\mu\partial_{x_2}\bar{v}_1e_1=\tau_we_1\ .$$

- $$\frac{\tau_w}{\delta}=-\partial_{x_1}\bar{p}=-\partial_{x_1}p_w \tag{4.13}$$

where p_w is called the wall pressure (at $x_2=0$ or $x_2=2\delta$) being only a function of x_1. The first equality gives a formula for the prescribed pressure gradient in x_1-direction.

- $$-\rho\langle\tilde{v}_1\tilde{v}_2\rangle+\mu\partial_{x_2}\bar{v}_1=\rho u_\tau^2\left(1-\frac{x_2}{\delta}\right) \tag{4.14}$$

In a further analysis of this equation x_2 is replaced by y. (4.14) is non-dimensionalized by introducing the coordinate $y^*=y/\delta\ (=x_2/\delta)$ and becomes

$$-\frac{\langle\tilde{v}_1\tilde{v}_2\rangle}{u_\tau^2}+\mathrm{Re}_\tau^{-1}\frac{\partial}{\partial y^*}\left(\frac{\bar{v}_1}{u_\tau}\right)=1-y^*$$

with the *friction Reynolds number*

$$\mathrm{Re}_\tau=\frac{\delta u_\tau}{\nu}\ . \tag{4.15}$$

Another coordinate is the *viscous* or *wall coordinate* $y^+=yu_\tau/\nu$. The scaled equations then read

$$-\frac{\langle\tilde{v}_1\tilde{v}_2\rangle}{u_\tau^2}+\frac{\partial}{\partial y^+}\left(\frac{\bar{v}_1}{u_\tau}\right)=\left(1-\mathrm{Re}_\tau^{-1}y^+\right)\ .$$

$\bar{v}_1/u_\tau$ as a function of y^* is denoted by $u^*(y^*)$. $\bar{v}_1/u_\tau$ as a function of y^+ is denoted by $u^+(y^+)$. In the viscous sublayer, towards the wall, the fluctuations have to go to zero. In this region $y^+ \sim 1$ the leading order terms are $\partial u^+/\partial y^+ = 1$ which has the solution

$$u^+(y^+) = y^+ \ . \tag{4.16}$$

Therefore the turbulent velocity profile the viscous sublayer increases linearly away from the wall. By further calculations one can derive a law in the intermediate region

$$y^* \ll 1 \quad \text{and} \quad y^+ \gg 1 \quad \text{while} \quad \frac{y^*}{y^+} = \mathrm{Re}_\tau \ .$$

A matched asymptotic expansion gives

$$u^*(y^*) = \frac{1}{\kappa}\log(y^*) - 1 \tag{4.17}$$

$$u^+(y^+) = \frac{1}{\kappa}\log(y^+) + 5 \tag{4.18}$$

with constants determined by experiments, in particular the *von Kármán constant* $\kappa = 0.4$. This dependency of the profile is called *logarithmic law of the wall*.

4.6.2 Computational setup

There is a vast collection of publications on computations performed and statistics collected; for instance in [KMM87, MKM99, HJ08] where reference databases on mean velocity profile, its derivative, and correlations of velocity fluctuations for different friction Reynolds numbers are provided. The problem has been more recently investigated by [JR07, VK14, KFWK17]. It has furthermore been added to the benchmark collection since the latest editions of HiOCFD, [3DT18], where our setup is based upon.

The computational domain is $\Omega = (0, 2\pi\delta) \times (0, 2\delta) \times (0, \pi\delta)$, $\delta = 1$, with periodic boundary conditions in the streamwise (x_1-) and spanwise (x_3-) direction. At the walls no-slip boundary conditions are used. The mesh is graded towards the no-slip boundaries to improve resolution of near-wall turbulent structures according to the hyperbolic tangent mesh mapping

$$\begin{aligned} [0,1] &\to [0, 2\delta] \\ x_2 &\mapsto \delta\left(\frac{\tanh(\gamma(2x_2 - 1))}{\tanh(\gamma)} + 1\right) \end{aligned} \tag{4.19}$$

using the mesh stretching parameter γ. The initial flow field is decomposed as $v_0(x) = \bar{v}_0(x) + \tilde{v}_0(x)$ into the mean part plus a perturbation. No-slip boundary conditions read

$$v(x,t) = g(x,t) = 0 \quad \text{on } x_2 = 0 \text{ and } x_2 = 2\delta \ .$$

The forcing term is chosen to reflect the constant pressure gradient (CPG) drive, [QFH16],

$$f = \frac{\tau_w}{\delta} e_1 \ . \tag{4.20}$$

Other possibilities with respect to the forcing are constant flow rate (CFR) and constant power input (CPI) where in both cases the streamwise component is dynamically adjusted over time. In the former to accomplish the *bulk velocity* of the flow to be constant. This strategy is also utilized in the simulation of flow over a periodic hill, [Per15]. Here the bulk velocity is defined as an average of the turbulent mean profile over streamwise-normal cross-sections

$$u_b = \frac{1}{|\Omega|}\int_\Omega \bar{v}_1(x)\mathrm{d}x \ . \tag{4.21}$$

To the bulk velocity one can associate as well the *bulk Reynolds number*

$$\mathrm{Re}_b = \frac{\delta u_b}{\nu} \ . \tag{4.22}$$

The current bulk velocity at simulation time t^k used for calibration with u_b is then

$$u_{b,sim}(t^k) = \frac{1}{|\Omega|}\int_\Omega v_1(x,t^k)\mathrm{d}x \ . \tag{4.23}$$

In the modeling section the main goal was to derive the turbulent velocity profile (4.16) in the viscous sublayer and (4.18) in the intermediate range. The *Reichardt function* approximates the profile up to the half channel height and matches the behavior at $y^+ \sim 1$ and $y^+ \gg 1$. It is used for the mean initial condition $\bar{v}_0(x)$:

$$\bar{v}_0(x) = u_\tau \left[\frac{1}{\kappa}\log(1+\kappa y^+) + C\left(1 - \exp\left(-\frac{y^+}{11}\right) - \frac{y^+}{11}\exp\left(-\frac{y^+}{3}\right)\right)\right]\cdot e_1 \tag{4.24}$$

with • $\kappa = 0.4$ the von Kármán constant
- C chosen to be of value 7.8
- $y^+ = u_\tau/\nu \ \min(x_2, 2\delta - x_2)$.

A display of the Reichardt function - together with the linear law and a DNS profile obtained by temporal averaging - is shown in figure 4.14.

We set the simulation parameters to $\delta = 1$, $\tau_w = 1$, $\rho = 1$, $\gamma = 2.3$ which leads to $u_\tau = 1$ and $\mathrm{Re}_\tau = 1/\mu$. Further the friction Reynolds number is set to $\mathrm{Re}_\tau = 550$. The computed bulk velocity of this flow has the value around 18 which yields the corresponding Reynolds number $\mathrm{Re}_b = 9900$. The initial velocity field $\bar{v}_0(x)$ is perturbed by adding

$$\begin{aligned}\tilde{v}_0(x) = \ &\epsilon\left(c\,\sin(ax_1)\sin(4\pi(x_2-\delta)^5)\cos(cx_3)\right)e_1 \\ &-\epsilon\left(a\,\cos(ax_1)\sin(4\pi(x_2-\delta)^5)\sin(cx_3)\right)e_3\end{aligned} \tag{4.25}$$

with amplitude $\epsilon = 10^{-2}$, wavenumber $k_x = 2$ and wavelength $a = k_x\,(2\pi)/(2\pi\delta)$ in streamwise direction, wavenumber $k_z = 2$ and wavelength $c = k_z\,(2\pi)/(\pi\delta)$ in spanwise direction. The choice of x_2-dependency is motivated by a simple wall-mode approximation.

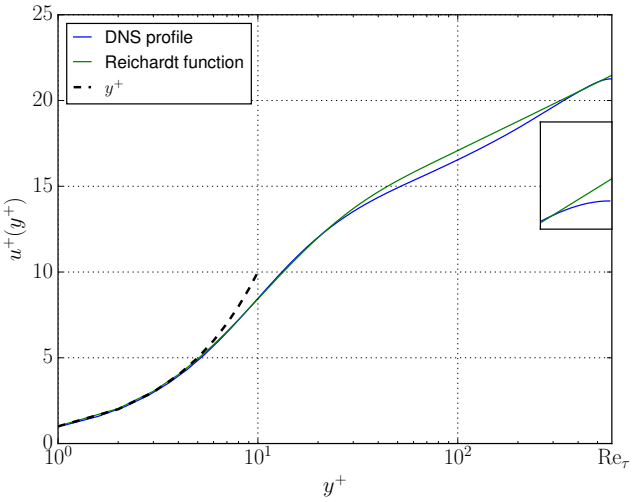


Figure 4.14: Graph of DNS profile, Reichardt function and y^+ in lin-log plot. The zoomed-in plot on the middle-right shows the behavior at $y^+ = \mathrm{Re}_\tau$ or $y^* = 1$. The DNS profile is obtained by temporal averaging over accurate numerical simulation snapshots. It is part of a collection of reference values for given friction Reynolds number. The boundary layer can be confined up to where the deviation from the linear law is small.

Computations have been done on a $128 \times 96 \times 64$ rectangular cuboid mesh that is anisotropically partitioned in wall-normal direction by means of Equation (4.19). Mesh size in x_2-direction of the boundary layer cells is $y^+ \approx 1.10779$. Polynomial degree for the velocity ansatz space is $p = 3$ and $p - 1 = 2$ for the pressure. The variant of the discrete solenoidal projection operator is given by Helmholtz-flux $H(\mathrm{div})$ reconstruction in RT_h^2. Temporal evolution of the incompressible channel flow is done with the RIPCS and Alexander's second order method in the viscous substep. The equations therein are solved with JFNK and flexible GMRes as a linear solver. In presence of the anisotropies due to Equation (4.19) and the mesh stretching parameter γ, the partially matrix-free variant of Gauss-Seidel is used as subdomain solves in the preconditioner. The mean initial condition is downscaled by a fraction. With CPG employed, the flow is accelerated towards the bulk velocity in balance of forces and eventually transitions to turbulence. To the channel flow a viscous time unit $t^+ = \delta/u_\tau$ and a convective time unit $t_b = 2\pi\delta/u_b$ can be associated. A constant time step of $\Delta t = 5 \cdot 10^{-4}\, t^+$ is chosen such that the estimated Courant number is $\mathrm{Cr} \sim 0.7$ when the flow is fully developed. A constant sampling rate avoids further biasing in the ideal surface renewal analysis with discrete structure functions that is carried out in Section 4.7. In total 18623 time steps are taken up to time $T = 11.213 t^+$ with about $1.73 \cdot 10^8$ spatial DOFs. Computations have been done on the BWFORDEV development cluster in Heidelberg.

Figures 4.15 and 4.16 show two time snapshots of the velocity field. The first image demonstrates the onset of boundary layer turbulence. The upper part of this figure gives a three-dimensional vector field visualization with the λ_2-criterion colored by the velocity

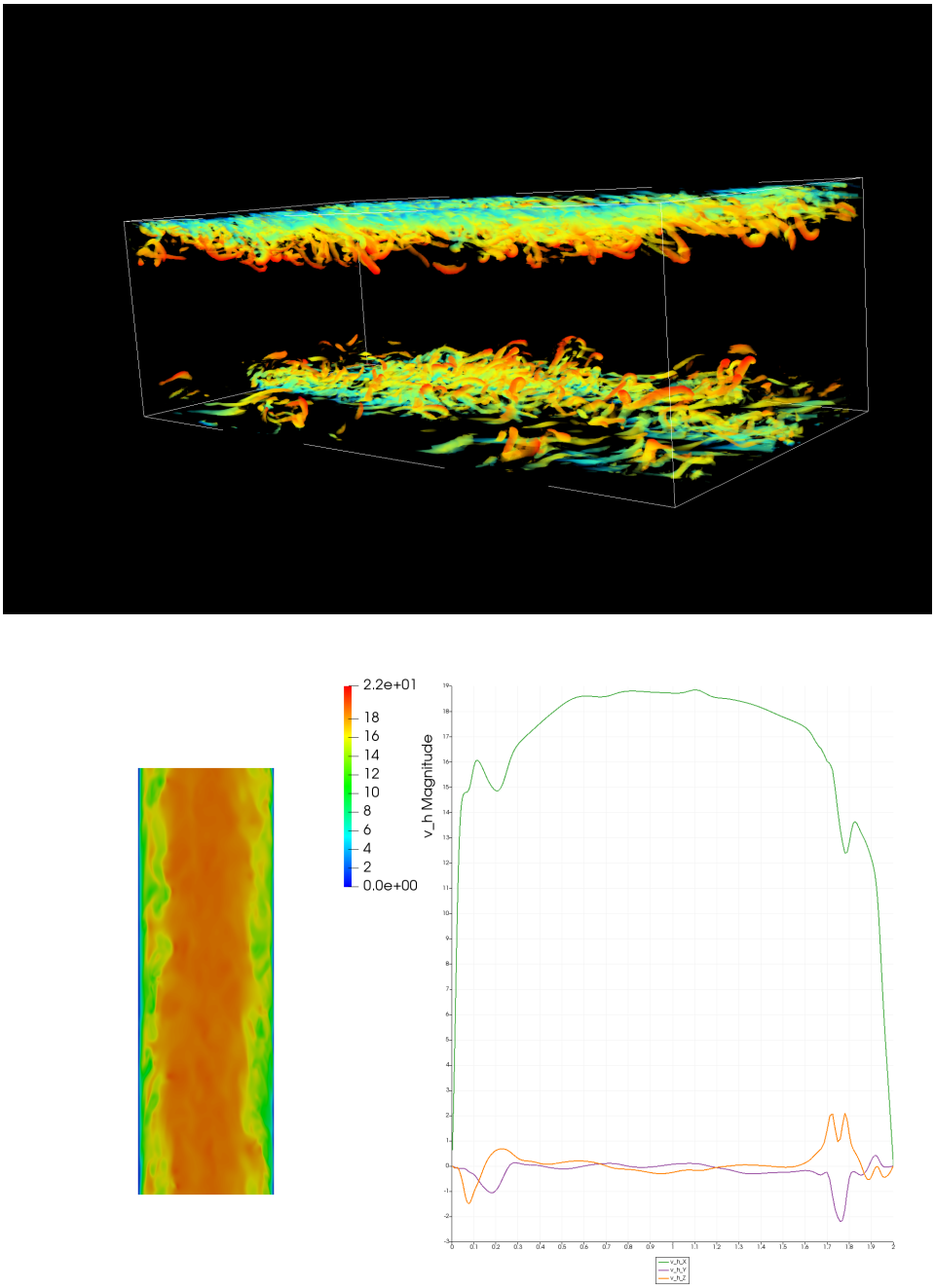


Figure 4.15: Onset of boundary layer turbulence in a three-dimensional channel flow at friction Reynolds number $\mathrm{Re}_\tau = 550$. Shown are evolving hairpin vortices captured by the λ_2-criterion that are colored by velocity magnitude. The bottom row shows a two- and one-dimensional plot of the velocity at the center of the channel where in the former the streamwise direction points upward.

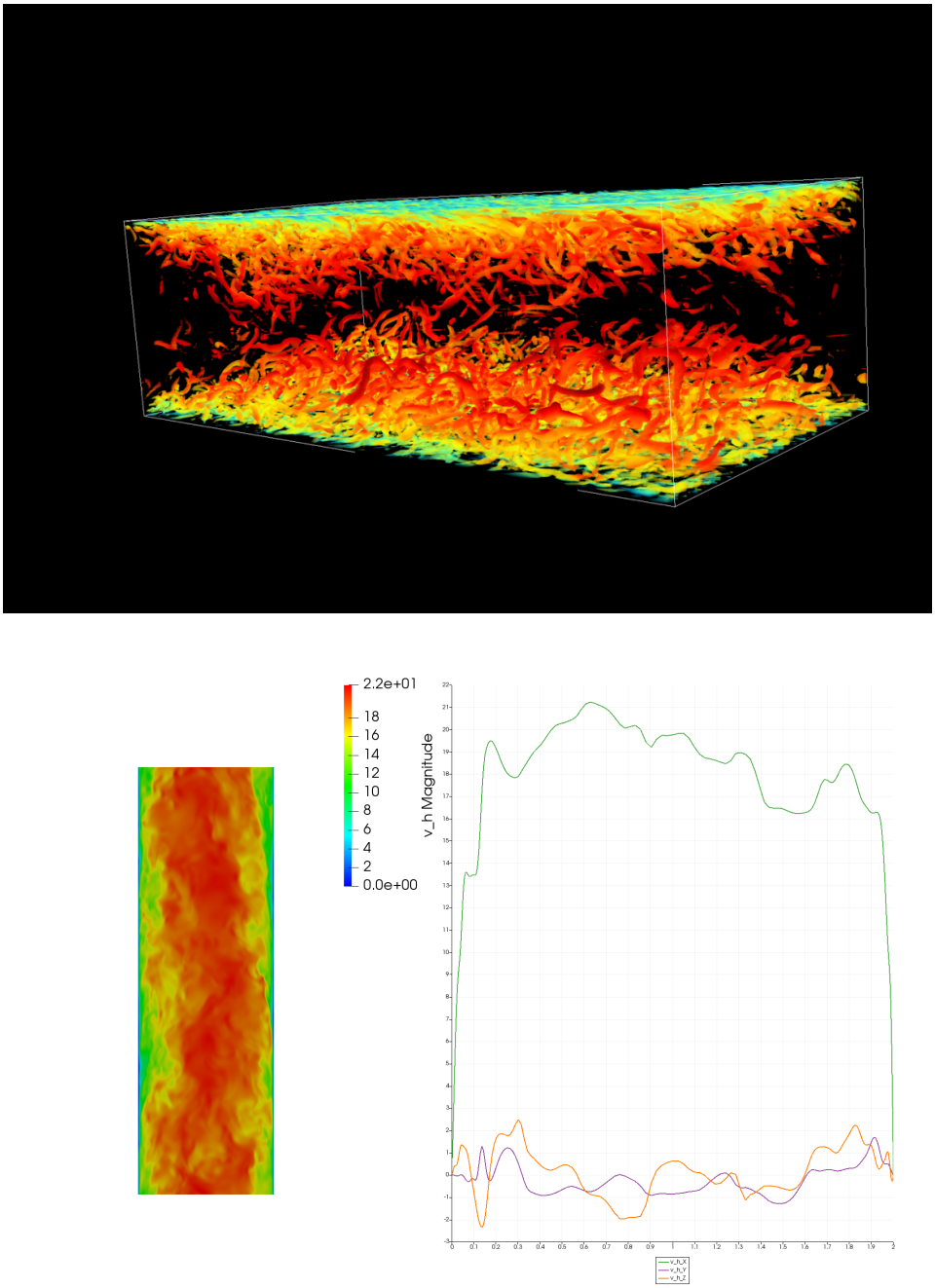


Figure 4.16: Fully developed boundary layer turbulence in a three-dimensional channel at friction Reynolds number $\mathrm{Re}_\tau = 550$. Shown are a visualization with the λ_2-criterion colored by $\|v_h\|_2$. The bottom row shows a two- and one-dimensional plot of the velocity at the center of the channel where in the former the streamwise direction points upward.

magnitude. The development of first *hairpin vortices* at both walls are captured which are the characteristic kinematic structures of boundary layer turbulence. The lower part of this figure shows the velocity magnitude at cross-section $x_3 = \pi\delta/2$ and the velocity components on the line in wall-normal direction with endpoints $(\pi\delta, 0, \pi\delta/2)^T$ and $(\pi\delta, 2\delta, \pi\delta/2)^T$. There the baseline mean profile in x_1-direction is still visible but already interspersed with wall-near oscillations. Note that the display of the two-dimensional cross-section is rotated by 90° such that the streamwise direction points upward. The second image 4.16 shows a corresponding visualization of fully developed boundary layer turbulence with the λ_2-criterion. A two- and one-dimensional cut of the velocity field as for the first snapshot has also been added.

Now let us again examine pointwise divergence and mass conservation errors here in a three-dimensional turbulent flow. According to Theorem 5 the postprocessed tentative velocity field by Helmholtz-flux reconstruction is pointwise divergence-free and satisfies the discrete continuity equation exactly. Figure 4.17 shows the error distribution at cross-section $x_3 = \pi\delta/2$ during the time snapshot taken as in figure 4.16. The pointwise divergence error $\left|\nabla \cdot v_h^{k+1}\right|$ is displayed in logarithmic scale, the maximum errors are located in the wall-near regions whereas the maximum in the middle is of order $2 \cdot 10^{-11}$. Mass conservation $\left|b(v_h^{k+1}, \chi_E) - r(\chi_E; t^{k+1} = 10.3519t^+)\right|$ for each cell E that intersects with the cross-section is displayed in the bottom picture, maximum errors up to $1.7 \cdot 10^{-14}$ are located in the wall-near regions as well. In the middle of the channel the error is by two orders of magnitude lower.

4.7 A model for surface renewal in the Atmospheric Boundary Layer

4.7.1 Description and related work

A model setup that we are to present for study of surface renewal events in the Atmospheric Boundary Layer (ABL), is based upon 3D turbulent channel flow. [HSLR99, KKSI11] have performed direct numerical simulations of turbulence in an open channel with passive scalar transport. At the open end that is also referred to as free surface, a no penetration condition $v \cdot n = 0$ plus free-slip for the tangential directions $\nu \nabla v n \times n = 0$ are imposed to model a shear-free flat interface. In both publications it has been argued that hairpin vortices or bursting events near the solid boundary are the dominant kinematic structure for surface renewal. Buoyancy effects in this setup are negligible because the streamwise velocity component is by one order of magnitude larger than the wall-normal component, c.f. bottom right in figure 4.16. The cited investigations were focused on the surface renewal at the shear-free interface where parcels close to the interface are replaced by fresh parcels coming with coherent structures from below. We are interested however in surface renewal within boundary layer turbulence in the wall-near region that are the viscous sublayer and the intermediate region. [KPCS06] shows and analyzes data of three measurement sites with different surface roughness. One site is located at the Antarctica which represents a smooth surface as the solid walls

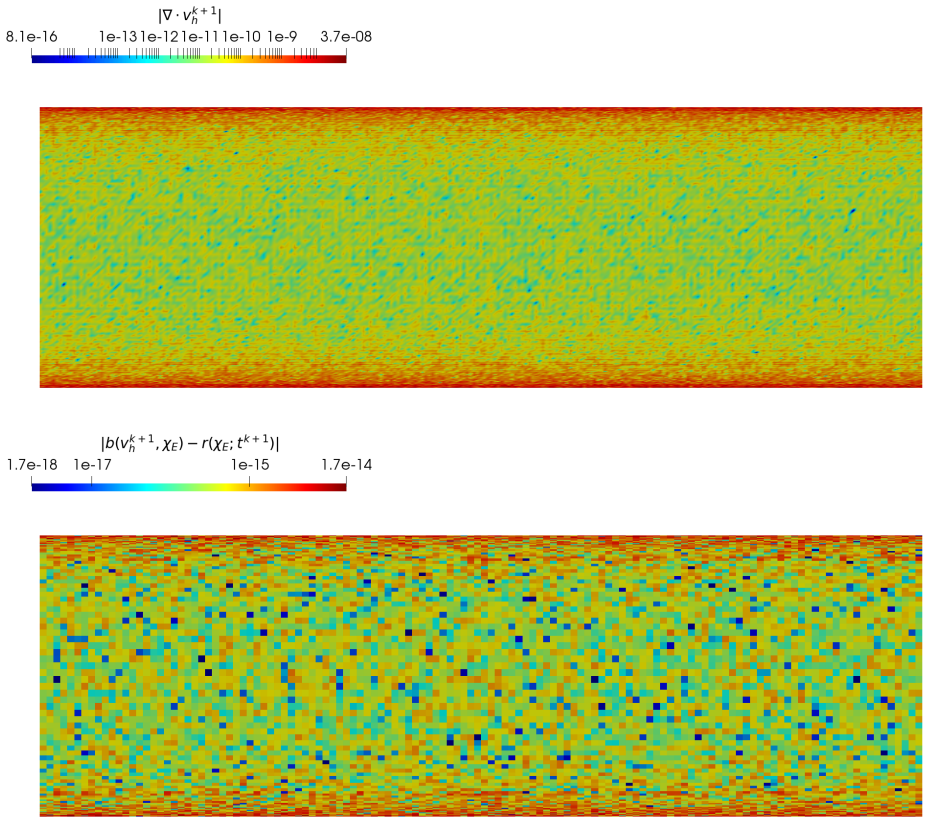


Figure 4.17: Pointwise divergence and mass conservation errors on cross-section $x_3 = \pi\delta/2$ at same time snapshot of boundary layer turbulence as in figure 4.16. Note that both quantities are scaled logarithmically.

in the channel. Another site is a loblolly pine forest as a rough surface example where it is explicitly demonstrated that the time series of a passive scalar exhibits more 'external intermittency' - which is less evident in the active scalar time series and visually absent in the velocity components. By external intermittency the authors name ejections of enriched parcels separated by quiescent periods. Returning to DNS, obtained root-mean-square (rms) profiles of the streamwise velocity $\sqrt{\langle \tilde{v}_1^2 \rangle}/u_\tau$ and concentration $\sqrt{\langle \tilde{c}^2 \rangle}$ have a maximum near the solid boundary. Approaching further the wall fluctuations have to go to zero due to the Dirichlet condition imposed. This region around maximum amount of fluctuations ($y^+ \lesssim 72$) supports observations that surface renewal events have their origin over there. Away from this region the streamwise rms velocity values are significantly lower. Evolution of passive scalar transport in open channel flow requires boundary conditions at the free surface. Either Dirichlet boundary condition or Neumann boundary conditions have been utilized in the direct numerical simulations. A sublayer at the open channel end has been observed for both boundary conditions.

There are noticeable differences around $y^+ = \mathrm{Re}_\tau$ in the mean profile $\langle c \rangle$ for each of the two boundary conditions prescribed since either the concentration is fixed or its normal derivative. This is even more clearly visible in the root-mean-square fluctuations. For Neumann boundary conditions a second local maximum is at the free surface. In contrast the local maximum for Dirichlet boundary conditions is in short distance below the interface since fluctuations eventually have to go to zero. The vicinity of this sharp peak is characterized by strong positive skewness and flatness of the root-mean-square profile. Skewness and flatness of fluctuations over there with Neumann conditions imposed are lower and become even negative.

In our model setup the domain configuration is mirrored at the open end and instead of boundary conditions, we place a sink for the passive scalar in the vicinity of the symmetry plane. The channel half height then models the maximum height of the ABL which captures parcels of high concentration moving away from the boundary layer and is a reservoir of low concentration parcels that travel towards the wall. As an initial condition we use $c_0(x) = 0$ and set the Dirichlet boundary condition value to $c = 0.5$ at the solid walls $x_2 = 0$ and $x_2 = 2\delta$, respectively. Periodic boundary conditions are prescribed in streamwise and spanwise direction. We add a sink term $s(x)c(x,t)$ on the left-hand side of the convection-diffusion equation (1.4) with $s(x) \geq 0$. $s(x)$ only depends on the wall-normal position and is parametrized by a Mexican Hat potential centered at the channel half height. At the roots of the fourth-order polynomial we cut off the progression to stay at zero. The shape of $s(x)$ is displayed in figure 4.18. The

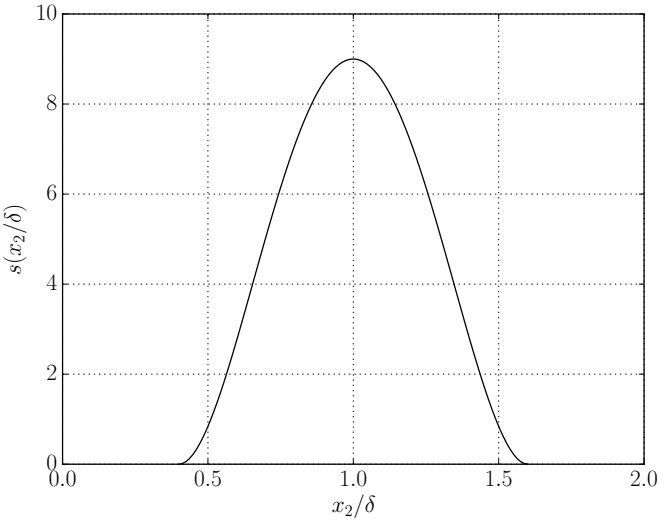


Figure 4.18: Shape of the sink - within the evolution of a passive scalar in 3D turbulent channel flow - as a function of x_2 between the lower and upper plate. The maximum is scaled to have a value of 9.

distribution along x_2-direction is further overlayed with an instantaneous concentration field in figure 4.19. The Schmidt number for scalar transport is set to $\mathrm{Sc} = 1$. We use polynomial degree $p = 3$ for the passive scalar ansatz space. Besides the same

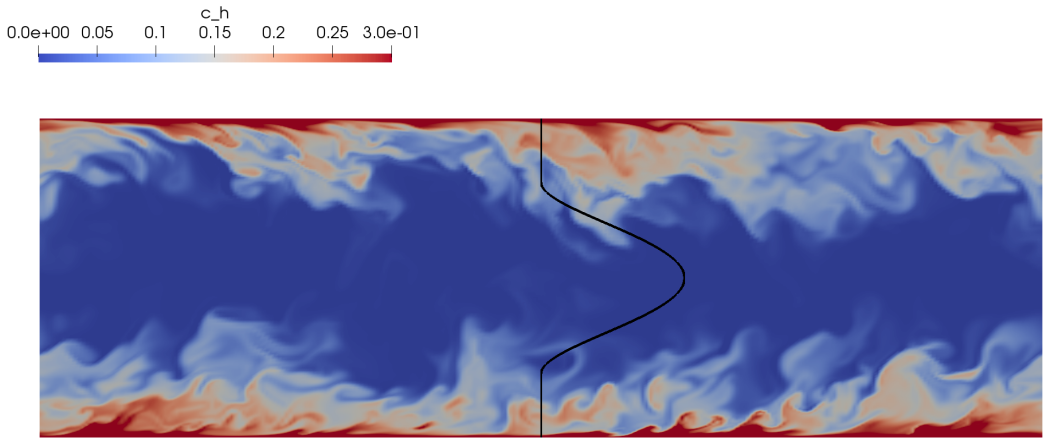


Figure 4.19: Instantaneous concentration field in fully developed boundary layer turbulence on the cross-section $x_3 = \pi\delta/2$, overlayed with the wall-normal sink dependence (scaling in x_1-direction arbitrary units).

discretization settings as in 3D turbulent channel flow are used.

Starting from the initial condition $c_0(x)$ and prescribed further by the Dirichlet boundary condition, the total concentration in the computational domain $\int_\Omega c(x,t)\mathrm{d}x$ increases at first over time. With help of Reynolds transport theorem the rate of change can be written as

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega c(x,t)\mathrm{d}x = \int_\Omega \partial_t c(x,t) + \nabla\cdot(c(x,t)v(x,t))\mathrm{d}x\ . \tag{4.26}$$

It is physically to be expected that for unit Schmidt number and $t > t_0 = 0$ the concentration field is sufficiently smooth such that the partial differential equation holds in its classical form:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega c(x,t)\mathrm{d}x = \int_\Omega D\Delta c(x,t) - s(x)c(x,t)\mathrm{d}x\ . \tag{4.27}$$

During the build-up phase the rate of change is dominated by $\int_\Omega D\Delta c(x,t)\mathrm{d}x > 0$. At later times the total tracer concentration tends to be in balance of production and extraction from the sink $-\int_\Omega s(x)c(x,t)\mathrm{d}x$. This behavior is displayed by the computed total concentration in figure 4.20.

For the ideal surface renewal analysis we place measurement probes at streamwise and spanwise center with different wall-normal positions that are summarized in Table 4.1. We use the techniques that have been employed in Section 1.3.4 to synthetic data, here to the concentration time series. 10000 samples are considered for structure function analysis with constant inverse sampling frequency $\Delta t = 5\cdot 10^{-4}t^+$. Start of the sampling period is indicated by a vertical green line in the total concentration curve, figure 4.20. Comparing the viscous coordinate for each probe in the table with the linear and logarithmic velocity profile in figure 4.14, we note that *c_y0* is in the viscous sublayer.

Table 4.1 Wall-normal measurement positions for time series of concentration in surface renewal. Given are also the corresponding viscous coordinate. All probes are further located at $x_1 = \pi\delta$ and $x_3 = \pi\delta/2$.

	c_y0	*c_y1*	*c_y2*	*c_y3*	*c_y4*	*c_y5*	*c_y6*	*c_y7*
y/δ	1e-2	5e-2	1e-1	2e-1	3e-1	5e-1	0.75	1.0
y^+	5.5	27.5	55	110	165	275	412.5	550

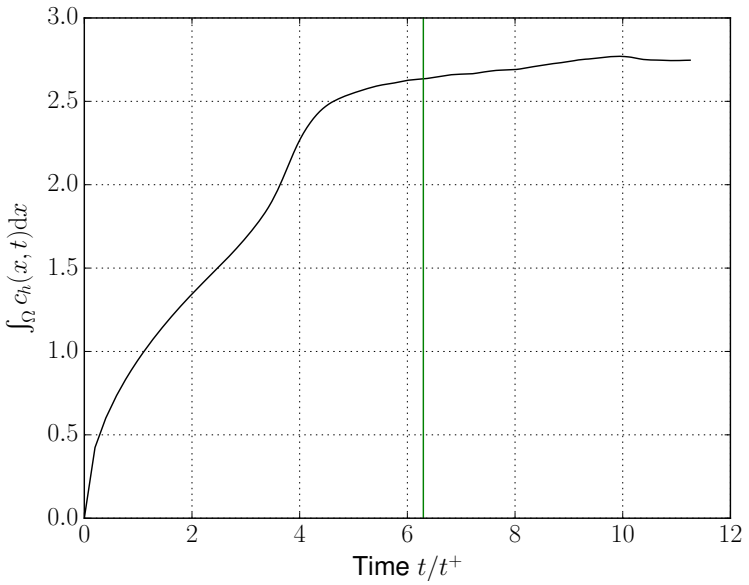


Figure 4.20: Computed total concentration over time in the proposed model for surface renewal. Also included is a vertical line in green color which indicates the start of time series for surface renewal analysis.

This time series is influenced by the constant Dirichlet boundary condition. Furthermore *c_y1* is found in the transition from linear to logarithmic law. *c_y2* and *c_y4* are in the intermediate range. The remaining higher located probes are already in the support of the sink term.

4.7.2 Time series analysis

Figure 4.21 shows the time series of interest split up into two time windows. The most visible ramp events have been marked in magenta color therein upon the data points and on the time axes. For all three positions we will examine the small-lag behavior of the truncated third-order structure function divided by the integer lag while keeping the analytical results (1.87) and (1.100) in mind. Secondly we will apply the linearized VA model and the finite microfront model to estimate ramp amplitude and ramp duration. Thirdly we will compare the obtained parameters with the computed truncated structure function and the ramp patterns in the time series.

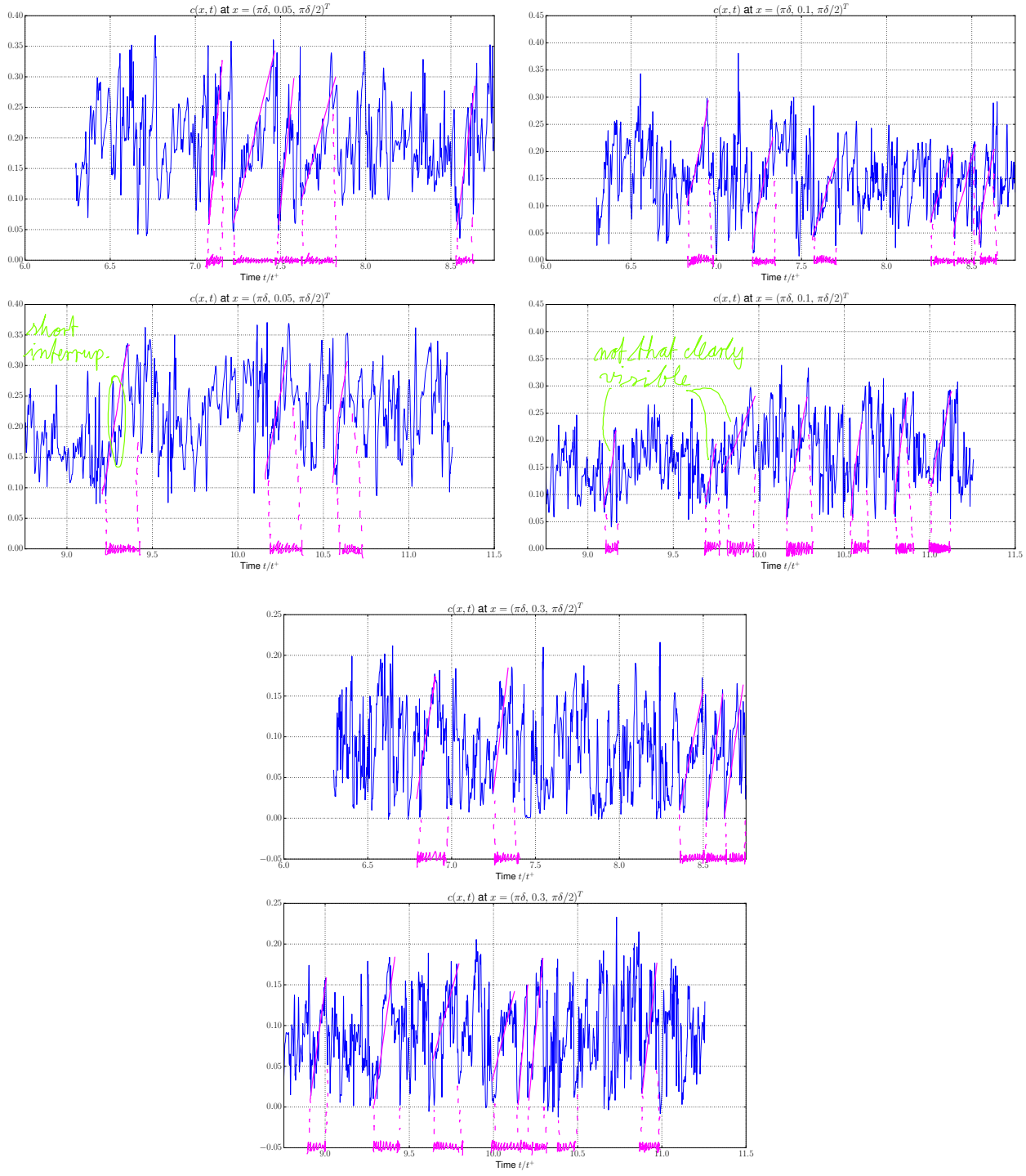


Figure 4.21: Passive scalar concentration time series at three different locations with wall-normal distance $y^+ = 27.5, 55, 165$. Shown is only the time window used for structure function analysis. Most visible ramp signals are further marked in magenta color.

Figure 4.22 shows the recorded structure function at $y^+ = 27.5$. The left part exhibits the typical small-lag behavior for the finite microfront model. $-\widehat{\langle(\delta c_h)^3\rangle}_{tr}/j$ has its maximum value at $j = 9$. The linearized VA model for this lag yields the following estimates of M and τ:

$$(\hat{M}_{VA}, \hat{\tau}_{VA}) = (0.16259, 0.175412t^+) \; . \tag{4.28}$$

The least squares fit to the finite microfront model in the intermediate range returns the values and belonging error bounds

$$\begin{aligned} \hat{M}_{mf} &= 0.12351 \pm 0.000642 \\ \hat{\tau}_{mf} &= 0.05389t^+ \pm 0.001171t^+ \\ \hat{t}_{f,\,mf} &= 0.00168t^+ \pm 0.000212t^+ \; . \end{aligned} \tag{4.29}$$

The right part shows an overlay of the data with the approximating function. It demonstrates that the model describes the third-order structure function well around the maximum, and the decline for $\Delta t < \Delta t_m$ and $\Delta t > \Delta t_m$, respectively.

Figure 4.24 shows the recorded structure function at $y^+ = 55$. Again the left part exhibits the typical small-lag behavior for the finite microfront model. $-\widehat{\langle(\delta c)^3\rangle}_{tr}(j\Delta t)/j$ has its maximum value at $j = 6$. The linearized VA model for this lag yields the following estimates of M and τ:

$$(\hat{M}_{VA}, \hat{\tau}_{VA}) = (0.13307, 0.111292t^+) \; . \tag{4.30}$$

The least squares fit to the finite microfront model in the intermediate range returns the values and belonging error bounds

$$\begin{aligned} \hat{M}_{mf} &= 0.10817 \pm 0.000996 \\ \hat{\tau}_{mf} &= 0.04991t^+ \pm 0.001691t^+ \\ \hat{t}_{f,\,mf} &= 0.00127t^+ \pm 0.000128t^+ \; . \end{aligned} \tag{4.31}$$

The right part shows an overlay of the data with the approximating function. As above it demonstrates that the model describes the third-order structure function well around the maximum, and the decline for $\Delta t < \Delta t_m$ and $\Delta t > \Delta t_m$, respectively. Finally, figure 4.26 shows the recorded structure function at $y^+ = 165$. As in the two previous cases the typical small-lag behavior for the finite microfront model is offered in the left part. $-\widehat{\langle(\delta c)^3\rangle}_{tr}(j\Delta t)/j$ has its maximum value at $j = 7$. The linearized VA model for this lag yields the following estimates of M and τ:

$$(\hat{M}_{VA}, \hat{\tau}_{VA}) = (0.10139, 0.100244t^+) \; . \tag{4.32}$$

The least squares fit to the finite microfront model in the intermediate range returns the values and belonging error bounds

$$\begin{aligned} \hat{M}_{mf} &= 0.08950 \pm 3.58 \cdot 10^{-4} \\ \hat{\tau}_{mf} &= 0.04966t^+ \pm 7.17 \cdot 10^{-4}t^+ \\ \hat{t}_{f,\,mf} &= 0.00166t^+ \pm 8.0 \cdot 10^{-5}t^+ \; . \end{aligned} \tag{4.33}$$

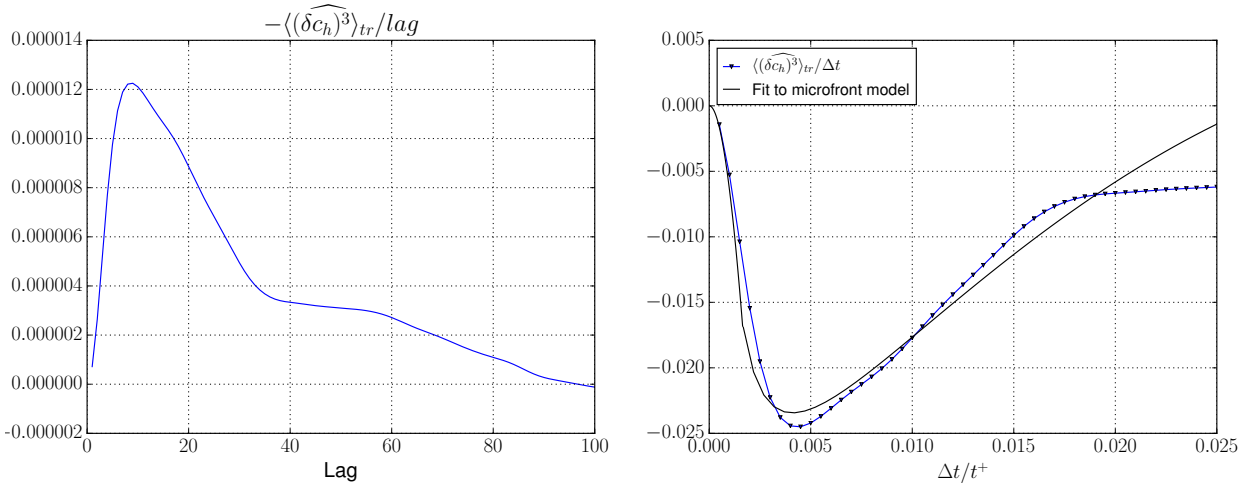


Figure 4.22: Computed third-order structure function at $y^+ = 27.5$. Left part shows the results divided by the integer lag and scaled by minus sign. Right part shows the results in terms of the time lag. Also included is a finite microfront model curve from the least squares fit.

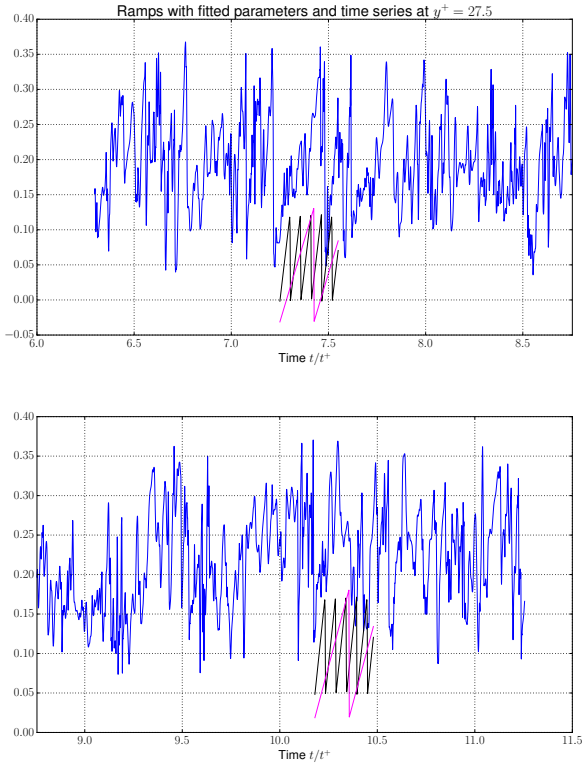


Figure 4.23: Concentration time series at $y^+ = 27.5$ and analytical sawtooth signals with the estimated parameters at selected time intervals. Magenta color shows the linearized VA model, black color the finite microfront model.

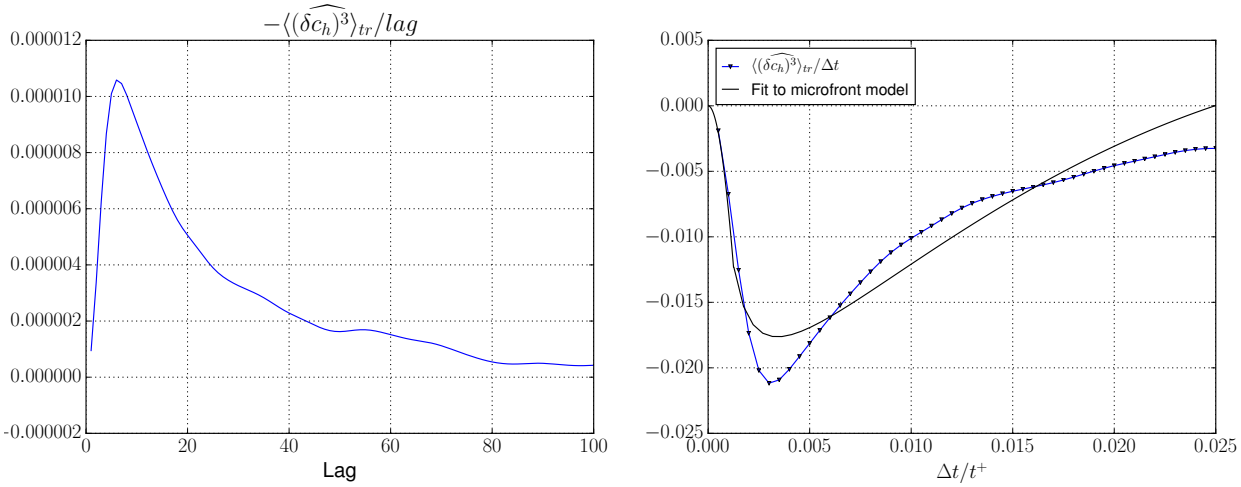


Figure 4.24: Computed third-order structure function at $y^+ = 55$. Left part shows the results divided by the integer lag and scaled by minus sign. Right part shows the results in terms of the time lag. Also included is a finite microfront model curve from the least squares fit.

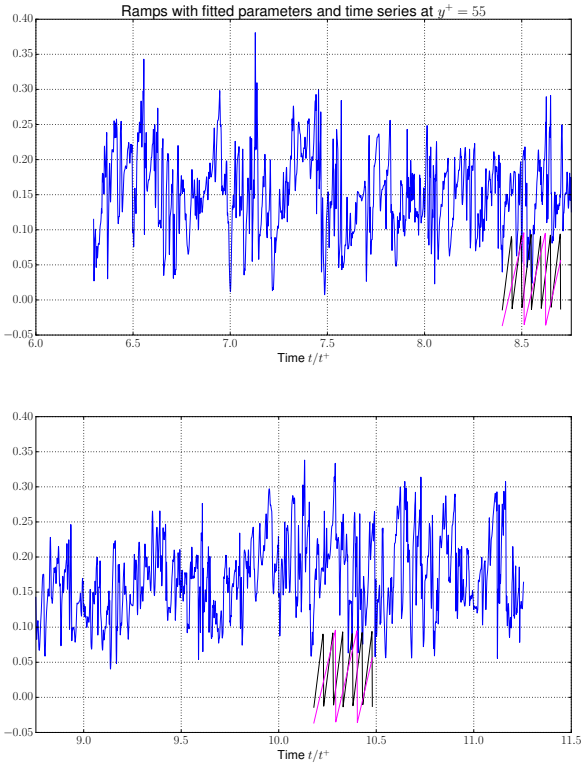


Figure 4.25: Concentration time series at $y^+ = 55$ and analytical sawtooth signals with the estimated parameters at selected time intervals. Magenta color shows the linearized VA model, black color the finite microfront model.

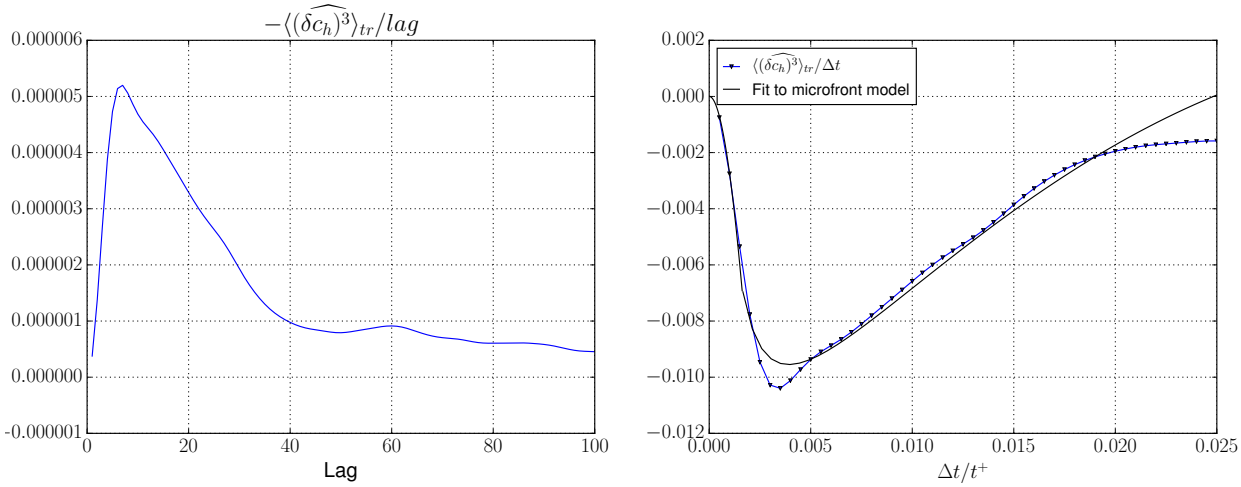


Figure 4.26: Computed third-order structure function at $y^+ = 165$. Left part shows the results divided by the integer lag and scaled by minus sign. Right part shows the results in terms of the time lag. Also included is a finite microfront model curve from the least squares fit.

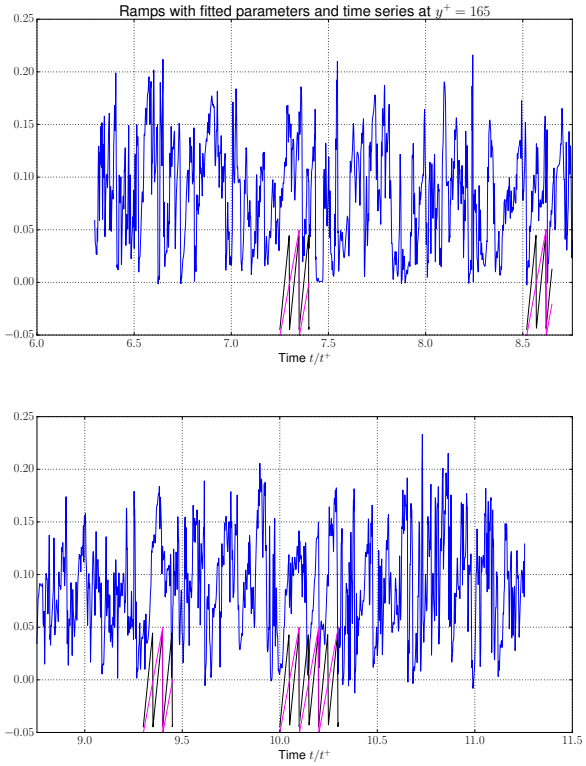


Figure 4.27: Concentration time series at $y^+ = 165$ and analytical sawtooth signals with the estimated parameters at selected time intervals. Magenta color shows the linearized VA model, black color the finite microfront model.

The right part shows an overlay of the data with the approximating function. As already mentioned the model describes the third-order structure function well around the maximum, and the decline for $\Delta t < \Delta t_m$ and $\Delta t > \Delta t_m$, respectively.

The estimated ramp duration in the linearized VA model are approximately three times larger at $y^+ = 27.5$ and two times larger at $y^+ = 55, 165$. This is likely due to the reasons mentioned in [CNBL97a] as the computation of $\hat{\tau}_{VA}$ with (1.93) in the VA model requires the cube of $\hat{M}_{VA}$ and therefore gives error amplification if $\hat{M}_{VA}$ is already defective. Our ratios for $\hat{M}_{VA}/\hat{M}_{mf}$, $\hat{\tau}_{VA}/\hat{\tau}_{mf}$ and $(\hat{M}/\hat{\tau})_{VA}/(\hat{M}/\hat{\tau})_{mf}$ are in agreement with the experimental results from [CNBL97a], Table 1 over there. The results are summarized in Table 4.2. We also observe that the linearized VA model overestimates M by $10\% - 30\%$ and overestimates τ by a factor 2-3. The authors of [CNBL97a] therefore introduce a correction factor for the maximum value of $-\widehat{\langle(\delta c)^3\rangle}_{tr}(\Delta t)/\Delta t$, which can be computed during sampling, to account for the value $M/\tau^{1/3}$. Let γ be

$$\frac{M}{\tau^{1/3}} =: -\gamma \left[\frac{-\widehat{\langle(\delta c)^3\rangle}_{tr}(\Delta t_m)}{\Delta t_m} \right]^{\frac{1}{3}} \tag{4.34}$$

where $-\widehat{\langle(\delta c)^3\rangle}_{tr}(\Delta t_m)/\Delta t_m$ denotes the maximum value of $-\widehat{\langle(\delta c)^3\rangle}_{tr}(\Delta t)/\Delta t$. With Equation (1.93) and the procedure to evaluate this at $\Delta t = \Delta t_m$ we obtain a correction for the linearized VA model:

$$\frac{\hat{M}_{VA}^3}{\hat{\tau}_{VA}} = -\frac{\widehat{\langle(\delta c)^3\rangle}_{tr}(\Delta t_m)}{\Delta t_m} ,$$

(4.34) now becomes

$$\frac{M}{\tau^{1/3}} = \gamma \frac{\hat{M}_{VA}}{\hat{\tau}_{VA}^{1/3}} . \tag{4.35}$$

Obviously $\gamma = 1$ corresponds to not correcting. Inserting the values from the microfront model on the left-hand side of (4.34) respective (4.35) gives corrections in the range 1.06-1.13. The last row of Table 4.2 presents the results for our three measurement locations. This is as well in agreement with values in [CNBL97a]: $\gamma = 1.167$ for the bare soil measurement site, $\gamma = 1.217$ for the straw mulch site, and $\gamma = 1.080$ for the Douglas-fir forest site. Figure 4.23 shows a plot of sawtooth signals based on the parameters (4.28) and (4.29), figure 4.25 sawtooth signals based on the parameters (4.30) and (4.31) and figure 4.27 sawtooth signals based on the parameters (4.32) and (4.33), each at selected time intervals in the corresponding time series where such patterns are visible. Compare with 4.21 upper left, upper right and bottom. The ramps in magenta color display the result of the linearized VA model, the ramps in black color the result of the finite microfront model. For sake of clarity we have chosen the base offset such that the analytical ramp signals are below the measured data points. In the three figures observe that the finite microfront model captures ramp events of shorter duration well within larger structures indicated by the drawings. However it should be noted that the ramps parametrized by linearized VA estimates capture the larger structures.

Table 4.2 Ratios of $\hat{M}$, $\hat{\tau}$ and $\hat{M}/\hat{\tau}$ determined with VA's linearized theory to those determined using the ramp model with finite microfront time.

y^+	27.5	55	165
$\frac{\hat{M}_{VA}}{\hat{M}_{fm}}$	1.316	1.230	1.133
$\frac{\hat{\tau}_{VA}}{\hat{\tau}_{fm}}$	3.255	2.230	2.019
$\frac{(\hat{M}/\hat{\tau})_{VA}}{(\hat{M}/\hat{\tau})_{fm}}$	0.404	0.552	0.561
γ	1.126	1.062	1.116

We close the time series analysis and compare ramp dimensions for the locations presented against each other. Note at first that for both the linearized VA and finite microfront model, the amplitude $\hat{M}$ decreases the further being away from the wall. This is in agreement with the mean concentration profile $\langle c_h \rangle$ and root-mean-square fluctuations $\sqrt{\langle \tilde{c}_h^2 \rangle}$. Both curves are displayed in figure 4.28. The statistical averaging has been done for every 10th time step in the period of structure function analysis. The first order statistical moment drops monotonically when moving away from the wall. As mentioned in [HSLR99] we also observe a logarithmic law for the average concentration which is indicated by a line in log-lin axes. Towards the center of the channel $\langle c_h \rangle$ tends to zero. The second order statistical moment exhibits a peak in the transitional region from linear to logarithmic law. The closest probe to the wall (at $y^+ = 5.5$) is before the maximum amount of fluctuations, the closest probe for consideration (at $y^+ = 27.5$) is just at the maximum before the decrease. The remaining probes are located where the descent of the rms-profile becomes flatter. Towards the center of the channel $\sqrt{\langle \tilde{c}_h^2 \rangle}$ is suppressed by the sink. Comparing the values $\sqrt{\langle \tilde{c}_h^2 \rangle}(y^+)$ for $y^+ \in \{27.5, 55, 165\}$ with the standard deviations of the sawtooth signals with amplitudes $\hat{M}_{VA} \in \{0.16259, 0.13307, 0.10139\}$, we can confine the ratio of surface renewal activity over total fluctuations to $60\% - 70\%$. Ramp duration $\hat{\tau}$ on the other hand is expected to stay constant within the different layers in channel turbulence. This is confirmed by the results. The values in {(4.30), (4.31)} and {(4.32), (4.33)} come from measurements in the range of logarithmic law. They do not change significantly for both models, the error bounds from the finite microfront model do intersect. The value in {(4.28), (4.29)} from the transitional region is higher.

4.7.3 Comparison with experimental data

We consider water vapor measurements that have been analyzed in [Oka17] with the linearized VA model as well. Exact origin of the dataset - in terms of a further publication, location of the measurement site and its surface roughness - has not been reported by

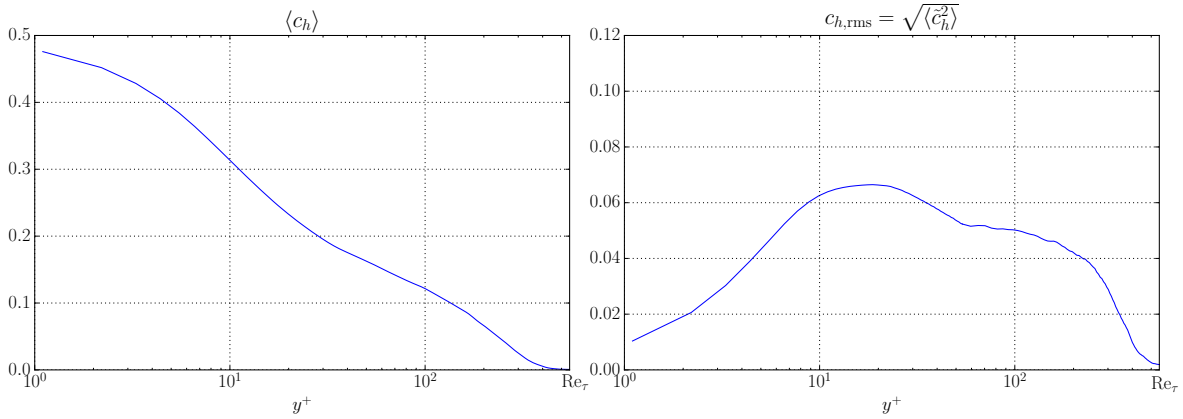


Figure 4.28: Computed average passive scalar concentration (left) and root-mean-square (right) in the proposed model for surface renewal. Both curves are plotted in lin-log axes over the viscous coordinate.

the author. Figure 4.29 shows the the whole time series split up into several windows. The most visible ramp events have been marked in magenta color upon the data points and on the time axes. Units for this record are given in g/m^3 at sampling frequency 20Hz. Figure 4.30 shows the computed structure function. Also the experimental data exhibits the typical small-lag behavior for the finite microfront model as seen in the left part. $-\widehat{\langle(\delta c)^3\rangle}_{tr}(j\Delta t)/j$ has its maximum value at $j = 5$. The linearized VA model for this lag yields the following estimates of M and τ:

$$(\hat{M}_{VA}, \hat{\tau}_{VA}) = (0.96297 \text{ g/m}^3, 8.99948\text{s}) \; . \tag{4.36}$$

The least squares fit to the finite microfront model in the intermediate range returns the values and belonging error bounds

$$\begin{aligned} \hat{M}_{mf} &= 0.92416 \text{ g/m}^3 \pm 0.00333 \text{ g/m}^3 \\ \hat{\tau}_{mf} &= 6.58747\text{s} \pm 0.10904\text{s} \\ \hat{t}_{f,\,mf} &= 0.07227\text{s} \pm 0.00629\text{s} \; . \end{aligned} \tag{4.37}$$

The right part shows an overlay of $\widehat{\langle(\delta c)^3\rangle}_{tr}(\Delta t)/\Delta t$ with the approximating function. It demonstrates that the model describes the third-order structure function well around the maximum, and the decline for $\Delta t < \Delta t_m$ and $\Delta t > \Delta t_m$, respectively. Figure 4.31 shows a plot of sawtooth signals based on the parameters (4.36) and (4.37) at selected intervals in the time series where such patterns are visible, c.f. figure 4.29 top right and bottom right. The ramps in magenta color display the result of the linearized VA model, the ramps in black color the result of the finite microfront model. For sake of clarity we have chosen the base offset such that the analytical ramp signals are below the measured data points. We observe that both models capture ramp events well in the experimental results. There are no significant differences compared to the DNS

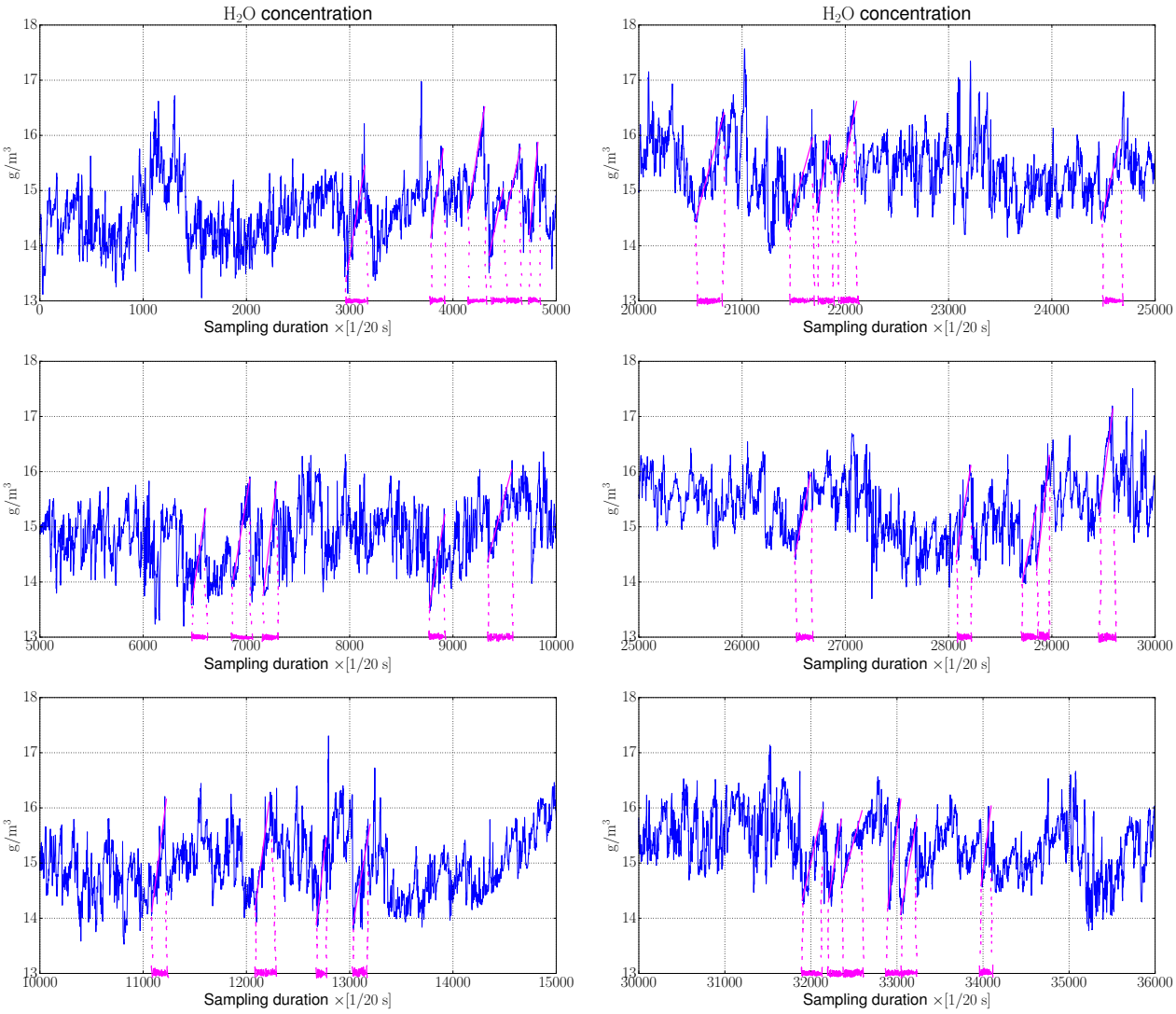


Figure 4.29: Water vapor concentration measurement. Most visible ramp signals in the time series are further marked in magenta color.

data with respect to the visual appearance of (1) the time series and the coherent structures appearing therein and of (2) the postprocessing results with the third-order structure function and its approximation by the finite microfront model. Sensitivities in the parameter triple $(\hat{M}_{mf}, \hat{\tau}_{mf}, \hat{t}_{f,\,mf})$ arise one or two digits, respectively, after the first non-zero decimal number for both the experimental run and numerical runs. Ratios of the parameters coming from the two models are $\hat{M}_{VA}/\hat{M}_{fm} = 1.042$, $\tau_{VA}/\tau_{mf} = 1.366$, and $(\hat{M}/\hat{\tau})_{VA}/(\hat{M}/\hat{\tau})_{mf} = 0.763$. They are closer to unity than the values collected in Table 4.2 but give nevertheless a similar correction factor of $\gamma = 1.065$.

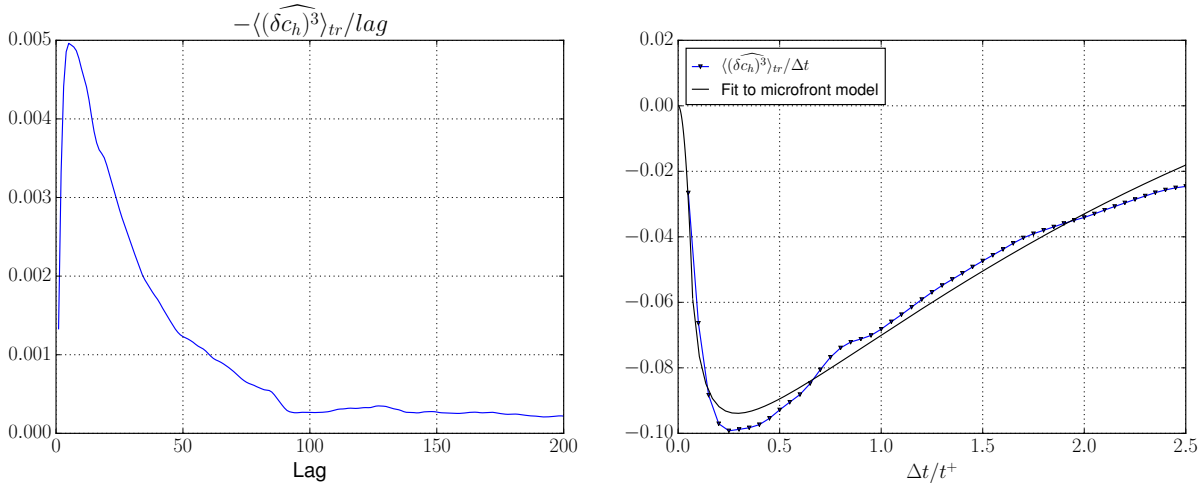


Figure 4.30: Computed third-order structure function from the water vapor time series. Left part shows the results divided by the integer lag and scaled by minus sign. Right part shows the results in terms of the time lag. Also included is a finite microfront curve from the least squares fit.

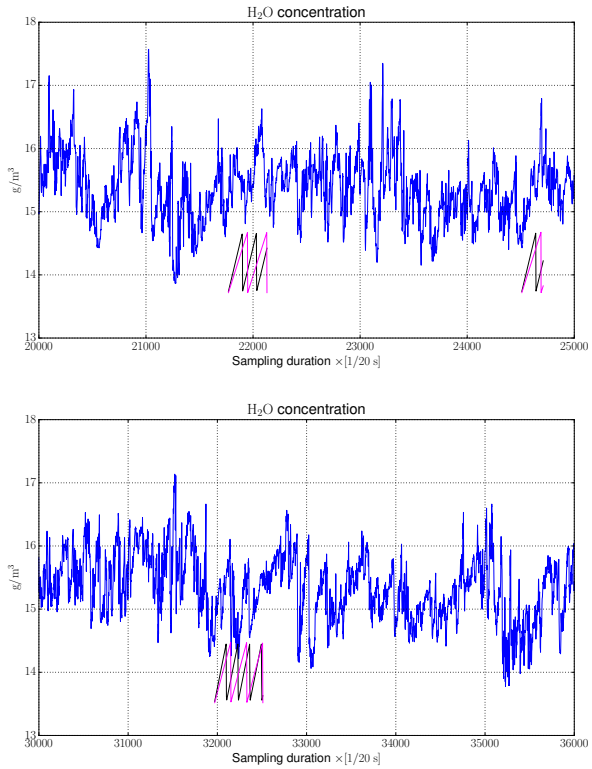


Figure 4.31: Water vapor concentration and analytical sawtooth signals with the estimated parameters at selected time intervals. Magenta color shows the linearized VA model, black color the finite microfront model.

4.7.4 Concentration field analysis

In this subsection we want to give a sketch of the surface renewal mechanism. The passive scalar concentration is studied on spanwise-normal cross-sections around $x_3 = \pi\delta/2$. We want to track the movement of coherent structures from the channel half height that are characterized by low concentration towards the boundary layer. The position of the probes that have been picked for structure function analysis are indicated by black squares in the concentration field. Compare with figures 4.32 and C0.1 which show several distribution snapshots of c_h. Below each, a time window of the recorded series at the probe closest to the wall is appended, $y^+ = 27.5$. A vertical green line in the measurement series shows the current time of display. Figure 4.32 shows three instants of time and demonstrate the movement of coherent structures. In the first subfigure (at $t = 7.32194t^+$) a low concentration parcel in the vicinity of the channel half height is marked by a black circle. It is going to travel to the measurement location in the middle. Its route is indicated by an arrow. The settling is characterized by regions of negative wall-normal velocity which go along the streamwise direction. The second subfigure (at $t = 7.48694t^+$) shows how high concentration has just been replaced by this fresh parcel at the lower probes. This can be seen in the termination of a ramp pattern at the bottom part. During residence time we have evaporation from the boundary. Concentration grows and marks the onset of a next ramp signal, compare the third subfigure at $t = 7.60194t^+$. Bursting events at the boundary layer are visible in the concentration field, also at different positions along the wall, in the other snapshots as well. These patterns in the concentration are generated by hairpin vortices of the flow field. A restricted visualization of the λ_2-criterion to the cross-sections which is not shown here demonstrates that negative values of λ_2 coincide with those patterns. Bursting events take away high concentration from the boundary. In the meantime the low concentration parcel has been enriched and has moved upwards. The black circle indicates that it has joined with another depleted region. Figure C0.1 in the appendix gives a further example of coherent structure movement. In the first subfigure (at $t = 9.98694t^+$) two low concentration parcels in the vicinity of the channel half height are marked by black circles. Their travel route is depicted by black arrows and continued in the uppermost part of the second subfigure. Note that the cross-section at time $t = 10.1019t^+$ is at $x_3 = 1.55\,\delta$ to track the low concentration better until it eventually reaches the measurement locations. Just before the arrival - as indicated by the arrows - the upper concentration parcel forms a tail that is slowed down and joins the lower parcel. The middle and bottom part of the second subfigure finally show the surface renewal event, the third subfigure (at $t = 10.3269t^+$) the subsequent growth of concentration and ejection.

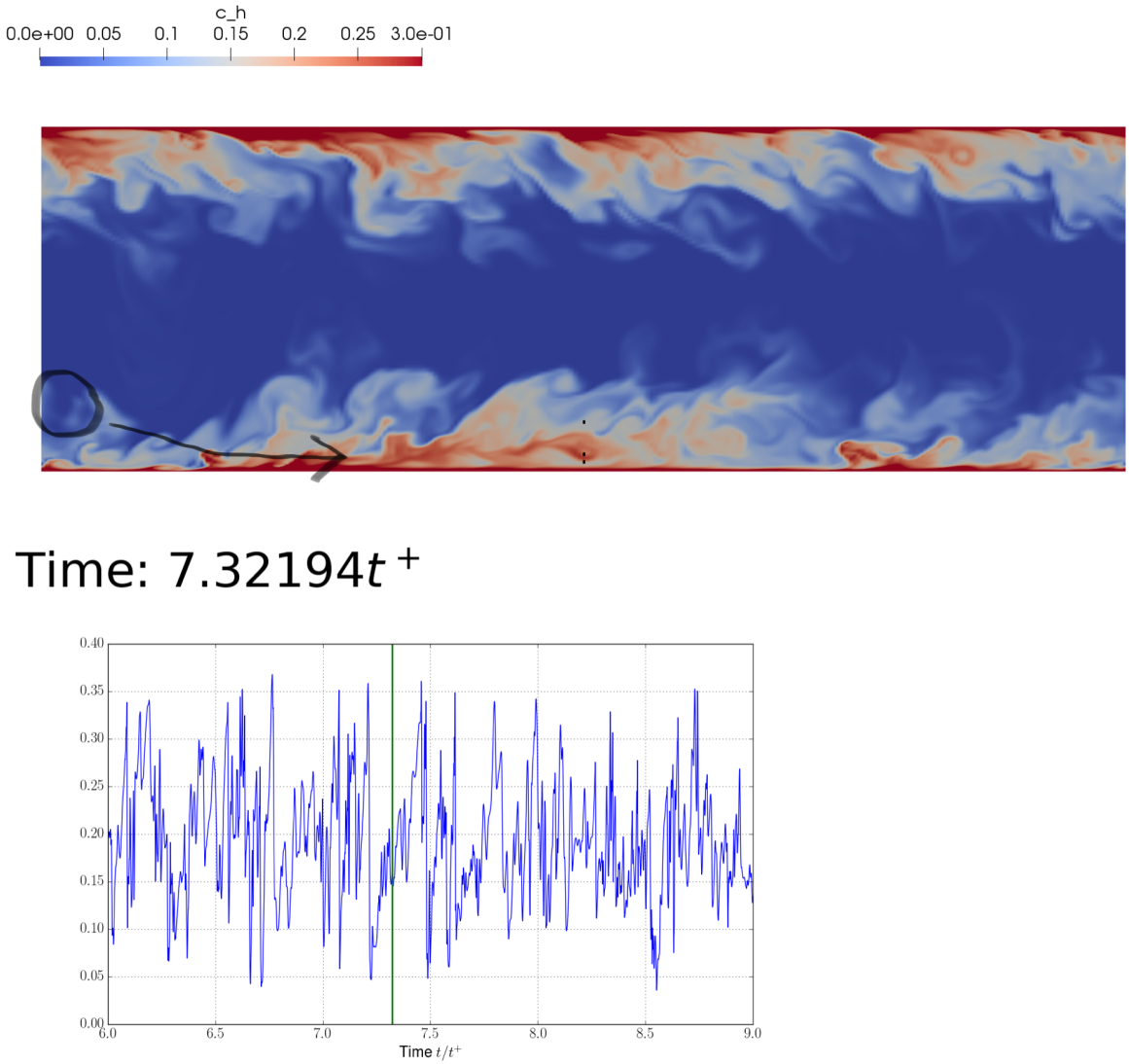


Figure 4.32: Surface renewal event - Part 1: Settling of low concentration parcel indicated by black circle on cross-section $x_3 = \pi\delta/2$. The bottom part shows a time window of the measured concentration at $y/\delta = 0.05$, indicated by the lowermost black square on the cut. The current time of display is marked by a vertical green line.

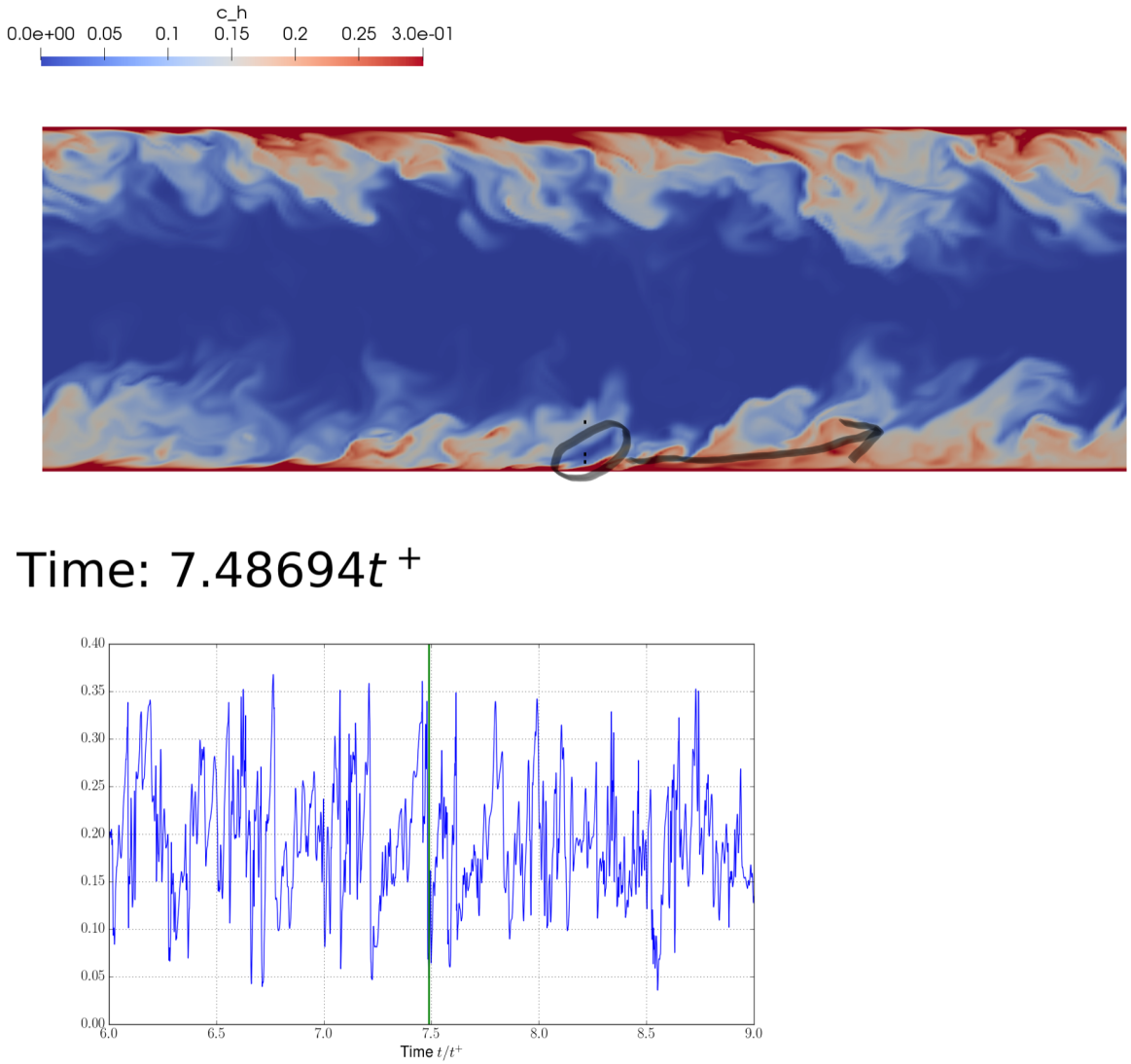


Figure 4.32: Surface renewal event - Part 2: On the cross-section $x_3 = \pi\delta/2$ low concentration parcel has arrived at the lower measurement probes as indicated by the black circle. It has replaced enriched concentration over there. The time series below takes a termination of a ramp pattern. The ejection phase to follow is indicated by a black arrow.

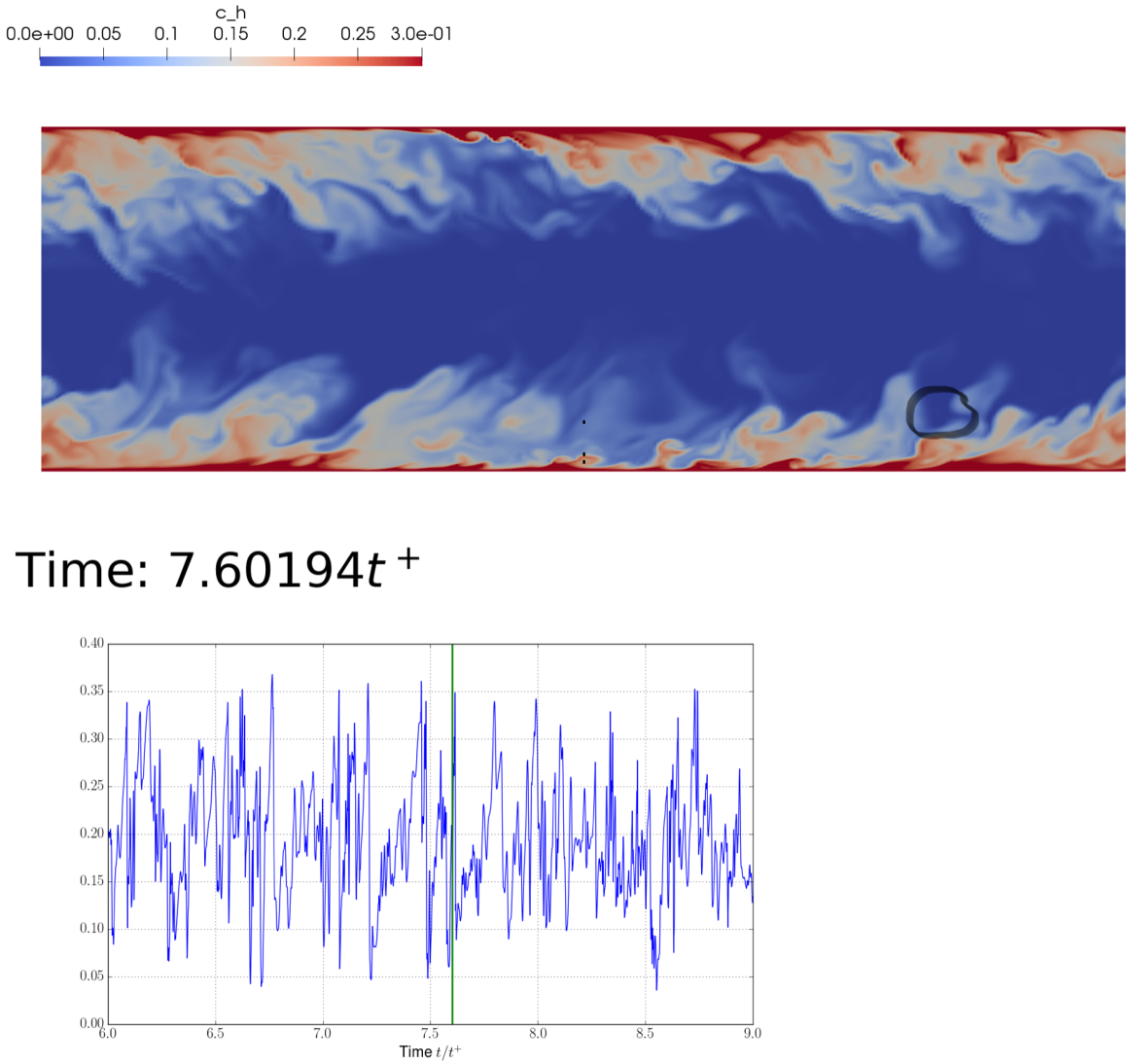


Figure 4.32: Surface renewal event - Part 3: Enrichment of the low concentration parcel during residence time in the reference frame moving in streamwise direction, and ejection. The black circle points out that it has joined with another depleted region from the channel half height. Growth of concentration at the lowermost measurement probe.

4.7.5 Summary of the surface renewal DNS

We have presented a flat model of the ABL to observe the surface renewal effect. It is built upon 3D turbulent channel flow such that the configuration is symmetric with respect to the channel half height. The boundary surface is smooth and flat, and already with unit Schmidt number and friction Reynolds number $\mathrm{Re}_\tau = 550$ surface renewal is evidently present. According to [ZBK17] values of Re_τ greater than 10^5 occur in the atmosphere. We observe coherent structures that are particularly characteristic for surface renewal (I) in the structure functions, (II) in the time series of measurement probes and (III) in two-dimensional visualizations of the concentration field. The small-lag behavior of the third-order structure function is represented well by the finite microfront model. Both the finite microfront model and the instantaneous termination model give a simple description of the activity based on fixed ramp amplitude, duration and quiescent period. Yet they reproduce the sampled structure function progression well for experimental data and also our DNS data. If only turbulent fluctuations in the time series were present, the computed third-order structure function would not resemble the typical small-lag behavior. This has been verified by computing the correlation of a purely fBm(1/3) signal, c.f. Section 1.3.4, but is not shown here. Further recall that in this section the coherent part made up 40% of the synthetic signal. By simply looking at the smoothness of the $-\widehat{\langle(\delta c)^3\rangle}_{tr}(j\Delta t)/j$ curves compared to figure 1.13, this supports the conclusion that surface renewal activity is of higher fraction in 3D turbulent channel flow.

5 Conclusions

In this chapter we discuss results obtained in the whole dissertation. For the numerical solution of the incompressible Navier-Stokes equations (being part also of the Navier-Stokes Boussinesq system) we have considered splitting methods, in particular the incremental and rotational pressure correction schemes, in combination with a high-order discontinuous Galerkin discretization.

- The upwind discretization of the convective term in the momentum equation uses a modified Vijayasundaram numerical flux function that takes into account that the discrete velocity field is not in $H(\mathrm{div})$. This discretization is applicable to high Reynolds number flow whereas the standard (or centered) discretization is limited to small Reynolds numbers.

- We developed postprocessing techniques in the Helmholtz projection step based on $H(\mathrm{div})$ reconstruction. We have shown that the induced projections are indeed discrete projection operators. The velocity field satisfies the discrete continuity equation, is thus locally mass conservative, and is for one variant also pointwise divergence-free.

- Numerical results confirm the conservation properties attained by these operators. The RIPCS is second-order convergent in time for Dirichlet and periodic boundary conditions.

- The fully implicit fractional-step solver has been implemented in the DUNE framework. It is able to run in both matrix-based and matrix-free mode. We have incorporated the SDGM and sum-factorization on quadrilateral/hexahedral meshes. Especially in combination with matrix-free methods algorithmic complexity is thereby reduced and our computations confirm that this leads to a significantly faster solution time even for low polynomial degrees. This observation is also true for the matrix-based method. On-the-fly computations further reduce storage requirements as the matrix does not have to be assembled and stored explicitly.

  For a fast convergent matrix-free iterative method the essential is to have robust and scalable preconditioners. We described matrix-free stationary iterative methods where the diagonal blocks are inverted iteratively on-the-fly. This new approach is used as a preconditioner to linearized problems from Newton's method for instance, likewise as a smoother in a hybrid DG multigrid algorithm with low order subspace correction.

  We have demonstrated the node-level performance of our code as well as scalability up to ~ 6100 cores. The efficient implementation of the numerical method

described in this work required a significant amount of low-level optimizations, leading to sophisticated code which is hard to maintain. Simultaneous work on a Python-based code generator that can transform a very abstract description of the variational form into highly optimized code has been shown to give promising results, [KHMB18].

- We have carried out computations of various types of turbulent flow. The properties of the numerical method have been investigated as a DNS tool and in underresolved turbulence. The resulting spectral distribution of kinetic energy exhibits the behavior predicted by the different flow regimes. Scales below numerical resolution are suppressed in the distribution of modes whereas larger scales are equivalently represented as on a finer resolution.

  We have proposed a model for numerical study of the surface renewal effect in the ABL. With the help of structure functions and parameterizations based on fixed ramp dimensions, we have demonstrated the existence of surface renewal events in silico by direct numerical simulation.

The numerical solver developed herein provides an efficient tool for simulating fluid flow prescribed by the incompressible Navier-Stokes equations or the Navier-Stokes Boussinesq system. However there are some minor limitations in the current version of the code: Only one passive and active scalar each can be coupled to the evolution at the time of writing.

Free-slip boundary conditions have not been incorporated into the weak form. The viscous term and discrete projection operators can be adapted accordingly, though. In the former, consistency and symmetrization variational forms only have to correct with the normal component of $\nabla v n$ on the free-slip boundary. Penalization on this part of the boundary is applied to the normal component of the velocity field.

A possible direction for future work is to apply the tensor product optimizations to a spectral DG solver for viscous compressible flow.

Appendix

A Discretization related

A.1 Multidimensional Calculus - Notation

Let $n, m \geq 1$ and $f : U \to \mathbb{R}^m$, $U \subset \mathbb{R}^n$ open, be differentiable. Then the Jacobian matrix of f in a point $x \in U$ reads

$$\nabla f(x) = \begin{pmatrix} \partial_1 f_1 & \dots & \partial_n f_1 \\ \vdots & & \vdots \\ \partial_1 f_m & \dots & \partial_n f_m \end{pmatrix} (x) \in \mathbb{R}^{m \times n} . \tag{A1.1}$$

As a shorthand notation we will simply write ∇f. With this particular sorting in the Jacobian, components f_i per row and partial derivatives per column, recall the chain rule. Let $f : U \to V$ and $g : V \to \mathbb{R}^p$, $U \subset \mathbb{R}^n$ open, $V \subset \mathbb{R}^m$ open, be differentiable, then the composition $g \circ f$ is also differentiable and it holds

$$\nabla(g \circ f)(x) = \nabla g(f(x)) \nabla f(x) \quad \text{(matrix-matrix product)} . \tag{A1.2}$$

As stated in the formula above $\nabla g(f(x))\nabla f(x)$ denotes the matrix-matrix product, and similarly for the matrix-vector product no operation symbol is used as the meaning should be clear from the context. Furthermore $a \cdot b$ denotes the Euclidean scalar product between the two vectors $a, b \in \mathbb{R}^n$. For two equally sized rectangular matrices $A, B \in \mathbb{R}^{m \times n}$, $A : B = \operatorname{tr}(A^T B)$ is the Frobenius scalar product between A and B.

Let $f : U \to \mathbb{R}$, $U \subset \mathbb{R}^n$ open, be a differentiable scalar function. Based on (A1.1) we have defined ∇f to be a row vector. In the case of scalar functions, we also introduce the transpose of the Jacobian matrix

$$\nabla f := \nabla f^T \in \mathbb{R}^n . \tag{A1.3}$$

Now let $f \in \mathcal{C}^m(U)$. By $\nabla^m f(x)$ we denote the tensor that contains the order m partial derivatives of f at $x \in U$. As for the Jacobian we use the shorthand notation $\nabla^m f$. In particular $\nabla^2 f$ is the Hessian matrix. For $f \in \mathcal{C}^2(U)$ it is known from Schwarz's theorem that the Hessian is symmetric:

$$\nabla^2 f^T = \nabla^2 f , \quad f \in \mathcal{C}^2(U) . \tag{A1.4}$$

Example - Chain rule: Jacobian transformation of a finite element function.
Let $\hat{E}, E \subset \mathbb{R}^n$ open, $\phi : E \to \mathbb{R}$ continuously differentiable, $\mu_E : \hat{E} \to E$ a diffeomorphism. In the finite element context $\hat{\phi}$ and ϕ are assigned by $\hat{\phi} := \phi \circ \mu_E$. The Jacobian of ϕ at $x = \mu_E(\hat{x})$ can be expressed with the Jacobian of $\hat{\phi}$ at $\hat{x} \in \hat{E}$:

$$\begin{aligned} \hat{\nabla}\hat{\phi}(\hat{x}) = \hat{\nabla}\phi(\mu_E(\hat{x})) &\overset{(A1.2)}{=} \nabla\phi(x)\hat{\nabla}\mu_E(\hat{x}) \\ \Leftrightarrow \nabla\phi(x) &= \hat{\nabla}\hat{\phi}(\hat{x})\hat{\nabla}\mu_E(\hat{x})^{-1} . \end{aligned}$$

This formula holds as well for vector-valued ϕ. Taking now the transpose on both sides gives

$$\mathbb{\nabla}\phi(x) = \hat{\nabla}\mu_E(\hat{x})^{-T}\hat{\mathbb{\nabla}}\hat{\phi}(\hat{x}) .$$

Another theorem that uses the Jacobian and derivatives of higher order is the expansion given by Taylor's theorem. For example let $f \in \mathcal{C}^2(U)$, $U \subset \mathbb{R}^n$ open. In the vicinity of $x \in U$, f can be expressed as

$$\begin{aligned} f(x + \Delta x) &= f(x) + \nabla f(x)\Delta x + \frac{1}{2}\Delta x^T \nabla^2 f(x)\Delta x + \text{remainder} \\ &= f(x) + \mathbb{\nabla} f(x) \cdot \Delta x + \frac{1}{2}\Delta x^T \nabla^2 f(x)\Delta x + \text{remainder} . \end{aligned} \tag{A1.5}$$

Particular choice of the remainder is not important here. For $f \in \mathcal{C}^2(U; \mathbb{R}^m)$ a similar expression holds:

$$f(x + \Delta x) = f(x) + \nabla f(x)\Delta x + \sum_{l=1}^{m} \Big(\Delta x^T \nabla^2 f_l(x)\Delta x\Big) e_l + \text{remainder} . \tag{A1.6}$$

Throughout this thesis the divergence operator is used. Therefore let $A : U \to \mathbb{R}^{m\times n}$, $U \subset \mathbb{R}^n$ open, be a differentiable matrix-valued function. Note that m does not necessarily have to be equal to the number of columns n, but the number of columns has to match the dimension of U. Then the divergence of A is defined as

$$\nabla \cdot A := \begin{pmatrix} \sum_{j=1}^{n} \partial_j A_{1j} \\ \vdots \\ \sum_{j=1}^{n} \partial_j A_{mj} \end{pmatrix} . \tag{A1.7}$$

For vector-valued $v : U \to \mathbb{R}^n$, the divergence of v reads

$$\nabla \cdot v := \sum_{i=1}^{n} \partial_i v_i . \tag{A1.8}$$

Example - Divergence of the Jacobian matrix Let $f : U \to \mathbb{R}^m$ be sufficiently smooth. The divergence applied to the Jacobian of f is equal to the Laplacian of f:

$$\nabla \cdot \nabla f = \Delta f . \tag{A1.9}$$

If now $p : U \to \mathbb{R}$ is sufficiently smooth, we have the identities

$$\nabla \cdot \mathbb{\nabla} p = \Delta p = \nabla \cdot \nabla p . \tag{A1.10}$$

Example - Pressure gradient in the Navier-Stokes equations As above let $p : U \to \mathbb{R}$ be differentiable. Consider the pressure gradient ∇p in the Navier-Stokes equations. Based on (A1.7) we have an alternative representation

$$\nabla \cdot (pI) = \nabla p \, . \tag{A1.11}$$

When the definitions (A1.7) and (A1.8) are applied to the divergence of a matrix-vector product, we can readily derive for $\nabla \cdot (Au)$ another product rule. In explicit let $A : U \to \mathbb{R}^{n \times m}$ and $u : U \to \mathbb{R}^m$, $U \subset \mathbb{R}^n$ open, then it holds

$$\nabla \cdot (Au) = (\nabla \cdot A^T) \cdot u + \mathrm{tr}(A \nabla u) = (\nabla \cdot A^T) \cdot u + A^T : \nabla u \, . \tag{A1.12}$$

Example - Integration by parts of the Laplacian Consider the integrand in the variational formulation of the Laplacian. For $v : U \to \mathbb{R}^n$, $u : U \to \mathbb{R}^n$, use formula (A1.12) and insert $A^T = \nabla v$ to obtain

$$\nabla \cdot (\nabla v^T u) = \Delta v \cdot u + \nabla v : \nabla u \, . \tag{A1.13}$$

A.2 Courant and grid Reynolds number estimation

In the Sections 3.3.2, 4.1- 4.7 we have mentioned the choice of the time-step size based on a computed Courant number. This estimation is realized with the updated velocity field during the volume integrals in the variational form, compare with Section 3.2.4. The results are used to calibrate the next simulation step. Recall the notation (3.10) for a corner in E

$$V(E, \hat{k}) := \mu_E(\hat{k})$$

where the position of the corners in the reference element are given by $\hat{k} \in \{0, 1\}^d$ and displayed in figure A2.1. Both numbers are estimated with what corresponds to

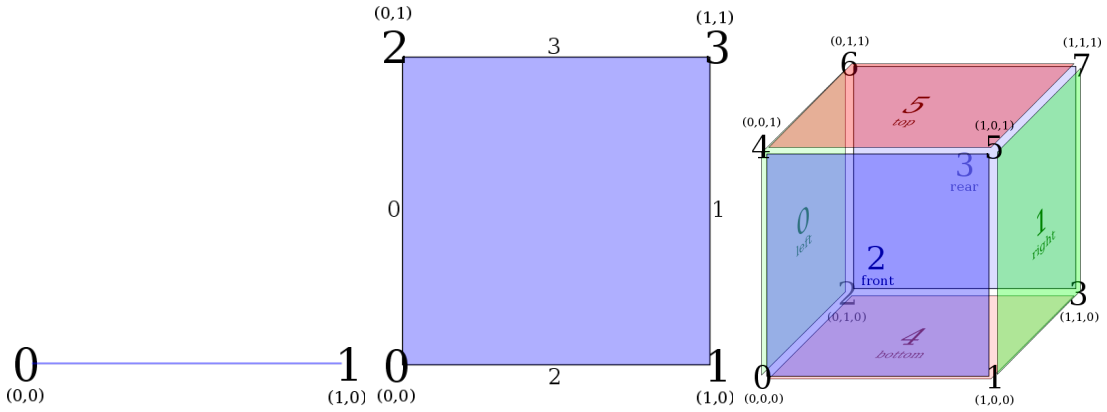


Figure A2.1: Corner numbering (and face numbering) in the reference elements $[0, 1]^d$ in 1D, 2D and 3D.

the extends if E is a parallelepiped, and are already required to set up the geometry transformation. In the following steps:

1. Calculate $V(E,(\delta_{i1},\delta_{i2},\delta_{i3}))-V(E,(0,0,0))$ for $i\in\{1,\dots,d\}$. On an axis-aligned quadrilateral/hexahedral mesh this gives $h_i e_i$.

2. Calculate $\left\|V(E,(\delta_{i1},\delta_{i2},\delta_{i3}))-V(E,(0,0,0))\right\|_2^2$ which gives h_i^2 on an axis-aligned quadrilateral/hexahedral mesh.

3. At each quadrature point in the volume reference element $\hat{E}$ calculate

$$\frac{\left|\hat{v}_h\cdot\Big(V(E,(\delta_{i1},\delta_{i2},\delta_{i3}))-V(E,(0,0,0))\Big)\right|}{\left\|V(E,(\delta_{i1},\delta_{i2},\delta_{i3}))-V(E,(0,0,0))\right\|_2^2}$$

   and

$$\left|\hat{v}_h\cdot\Big(V(E,(\delta_{i1},\delta_{i2},\delta_{i3}))-V(E,(0,0,0))\Big)\right|.$$

   On an axis-parallel grid the first quantity is equal to $\sum_{i=1}^d |(v_h)_i|/h_i$ and the latter to $\sum_{i=1}^d |(v_h)_i|\,h_i$ at the image of the quadrature point in E. On quadrilateral/hexahedral meshes these two quantities serve as an estimation for $\|v_h\|_2/h$ and $\|v_h\|_2 h$ at this point. Taking the maximum over all quadrature nodes in E gives a cell-wise estimation for both quantities.

4. The maximum over all cells estimates $\|v_h\|_2/h$ and $\|v_h\|_2 h$.

With the current time step size Δt, density ρ and viscosity μ, Courant number and grid Reynolds number can be estimated,

$$\mathrm{Cr}=\frac{\|v_h\|_2\,\Delta t}{h} \tag{A2.14}$$

$$\mathrm{Re}_h=\frac{\rho\,\|v_h\|_2\,h}{\mu}\,. \tag{A2.15}$$

A.3 Face embeddings

As in Section 3.2.2 recall the notation:
The embedding of face e into $E^{\mathrm{int}}, E^{\mathrm{ext}}$ is described by maps $\eta_e^{\mathrm{int,ext}}:\hat{e}\to\hat{E}$ of the corresponding reference elements such that $\mu_{E^{\mathrm{int,ext}}(e)}\circ\eta_e^{\mathrm{int,ext}}=\mu_e$ holds. The maps $\eta_e^{\mathrm{int,ext}}:\hat{e}\to\hat{E}$ map coordinate number $q\in\{1,\dots,d-1\}$ in $\hat{e}$ to coordinate number $\pi_e^{\mathrm{int,ext}}(q)\in\{1,\dots,d\}$ in $\hat{E}$. We may extend the map $\pi_e^{\mathrm{int,ext}}$ to $\{1,\dots,d\}$ by requiring $\pi_e^{\mathrm{int,ext}}(d)$ to be the unique coordinate number that face $\hat{e}$ is perpendicular to in the embedded $\hat{E}$. Thereby $\pi_e^{\mathrm{int,ext}}$ is a permutation of $\{1,\dots,d\}$ and the numbers $\pi_e^{\mathrm{int,ext}}(q), q\in\{1,\dots,d-1\}$ are referred to as the *tangential directions* in $\hat{E}$. This implies an extension of the map $\eta_e^{\mathrm{int,ext}}$ to $\hat{e}\times\{0\}$ and the normal component can take the

values $\eta^{\mathrm{int,ext}}_{e,\pi_e^{\mathrm{int,ext}}(d)}(0) \in \{0,1\}$. Finally, since faces in the volume reference element are axis-parallel, the component

$$\left(\eta_e^{\mathrm{int,ext}}(\hat{s})\right)_{\pi_e^{\mathrm{int,ext}}(q)} = \eta^{\mathrm{int,ext}}_{e,\pi_e^{\mathrm{int,ext}}(q)}(\hat{s}) = \eta^{\mathrm{int,ext}}_{e,\pi_e^{\mathrm{int,ext}}(q)}(\hat{s}_q)$$

is a function of one variable and in absence of hanging nodes it is even an isometry.

A.3.1 Example

Let $d = 3$ and consider the trivial embedding in case of a structured cuboid grid: Prescribing the normal directions to 1, 2 or 3, respectively, the permutation $\pi_e^{\mathrm{int,ext}}$ takes the form

$$\begin{aligned}
&\pi_e^{\mathrm{int,ext}}(3) = 1 && \text{so } \{1,2,3\} \mapsto \{2,3,1\}\\
&\pi_e^{\mathrm{int,ext}}(3) = 2 && \text{so } \{1,2,3\} \mapsto \{1,3,2\}\\
&\pi_e^{\mathrm{int,ext}}(3) = 3 && \text{so } \{1,2,3\} \mapsto \{1,2,3\}.
\end{aligned}$$

However $\pi_e^{\mathrm{int,ext}}$ does not describe the entire information of a face embedding. Also one needs to specify the $\eta^{\mathrm{int,ext}}_{e,\pi_e^{\mathrm{int,ext}}(q)}(\hat{s}_q) : (0,1) \to (0,1)$ which is given by the identity here.

Return now to the general case of a face e to embed into elements E^{int}, E^{ext}. By prescribing $\pi_e^{\mathrm{int,ext}}(d)$, the normal direction of e in the associated $\hat{E}$, and $\eta^{\mathrm{int,ext}}_{e,\pi_e^{\mathrm{int,ext}}(d)}(0)$, the sign of the unit outer normal vector on $\partial\hat{E}$, the face of the volume reference element is fixed where $\hat{e}$ is mapped to. Note that both quantities can be easily obtained with the face numbering that comes with the implementation of the reference elements, c.f. figure A2.1. Then there are $(d-1)!$ possibilities left for $\pi_e^{\mathrm{int,ext}}$ to be a permutation of $\{1,\dots,d\}$. Moreover each component can be of the form

$$\begin{aligned}
\eta^{\mathrm{int,ext}}_{e,\pi_e^{\mathrm{int,ext}}(q)}(\hat{s}_q) : (0,1) &\to (0,1)\\
\hat{s}_q &\mapsto \hat{s}_q\\
\text{or } \hat{s}_q &\mapsto 1-\hat{s}_q
\end{aligned}$$

giving another 2^{d-1} possibilities. Hence there are $2^{d-1}(d-1)!$ possibilities in total.

Two embedding variants in three dimensions (for fixed normal direction and fixed sign of the unit outer normal vector) are displayed in figure A3.2. The corners of the reference face are denoted by $0,1,2,3$. Their image in $\hat{E}$ are shown for mirroring and rotation by 90°. The mirroring variant is given such that the orientation of the tangential directions is preserved, $\pi_e(1) < \pi_e(2)$ as in the trivial embedding stated above. Each component is mapped according to

$$\begin{aligned}
\eta_{e,\pi_e(1)}(\hat{s}_1) &= 1-\hat{s}_1\\
\eta_{e,\pi_e(2)}(\hat{s}_2) &= \hat{s}_2\ .
\end{aligned}$$

In contrast, rotation by 90° causes the tangential directions to be swapped, $\pi_e(2) < \pi_e(1)$. Each component is then mapped as per

$$\eta_{e,\pi_e(1)}(\hat{s}_1) = \hat{s}_1$$
$$\eta_{e,\pi_e(2)}(\hat{s}_2) = 1 - \hat{s}_2 \ .$$

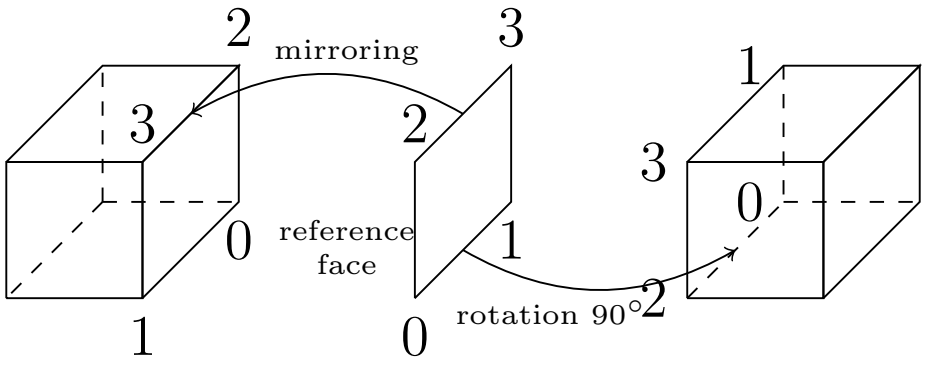


Figure A3.2: Two example embeddings of $\hat{e}$ into $\hat{E}$ in three dimensions. The corners of the reference face (in the middle) are denoted by $0, 1, 2, 3$. To the left the position of these face corners in $\hat{E}$ are shown for mirroring along the $\hat{s}_2$-direction. To the right resulting positions for rotation by 90° can be seen.

A.4 Dimension of local function spaces

Table A4.1 Dimension of element-local function spaces for polynomial degrees $p \in \{1, \dots, 10\}$. The left subtable shows the local size of Q_h^{p-1}, X_h^p and RT_h^{p-1} (second from left to right column) on quadrilateral elements in two space dimensions. Right subtable the corresponding sizes on hexahedral elements in three space dimensions.

p	$\mathbb{Q}_{p-1,2}$	$\mathbb{Q}_{p,2} \times \mathbb{Q}_{p,2}$	RT_E^{p-1}
1	1	8	4
2	4	18	12
3	9	32	24
4	16	50	40
5	25	72	60
6	36	98	84
7	49	128	112
8	64	162	144
9	81	200	180
10	100	242	220

p	$\mathbb{Q}_{p-1,3}$	$\mathbb{Q}_{p,3} \times \mathbb{Q}_{p,3} \times \mathbb{Q}_{p,3}$	RT_E^{p-1}
1	1	24	6
2	8	81	36
3	27	192	108
4	64	375	240
5	125	648	450
6	216	1029	756
7	343	1536	1176
8	512	2187	1728
9	729	3000	2430
10	1000	3993	3300

A.5 Diagonal implicit Runge-Kutta Butcher tableaus

In the following three sections we list Butcher tableaus of different numerical methods for ordinary differential equations. The use of Butcher tableaus for in-time evolution has been introduced in Section 2.2.5 using the example of the Navier-Stokes viscous substep.

A.5.1 One-Step θ-method

$$\alpha = \begin{pmatrix} -1 & 1 \end{pmatrix} \quad \beta = \begin{pmatrix} 1-\theta & \theta \end{pmatrix} \quad \delta = \begin{pmatrix} 0 & 1 \end{pmatrix}$$

with $0 \leq \theta \leq 1$.

A.5.2 Alexander's two-stage method

$$\alpha = \begin{pmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix} \quad \beta = \begin{pmatrix} 0 & \alpha & 0 \\ 0 & 1-\alpha & \alpha \end{pmatrix} \quad \delta = \begin{pmatrix} 0 & \alpha & 1 \end{pmatrix}$$

with $\alpha = 1 - \frac{\sqrt{2}}{2}$.

A.5.3 Fractional-Step θ-method

$$\alpha = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix}, \quad \beta = \begin{pmatrix} \theta(1-\alpha) & \theta\alpha & 0 & 0 \\ 0 & \theta'\alpha & \theta'(1-\alpha) & 0 \\ 0 & 0 & \theta(1-\alpha) & \theta\alpha \end{pmatrix} \quad \delta = \begin{pmatrix} 0 & \theta & 1-\theta & 1 \end{pmatrix}$$

with $\theta = 1 - \frac{\sqrt{2}}{2}$, $\alpha = 2\theta$, $\theta' = 1 - 2\theta = 1 - \alpha = \sqrt{2} - 1$. Note also that $\theta\alpha = \theta'(1-\alpha) = 2\theta^2$.

A.6 Implicit multistep methods

A.6.1 Newmark β-scheme

$$\alpha = \begin{pmatrix} 1 & -2 & 1 \end{pmatrix} \quad \beta = \begin{pmatrix} \beta & 1-2\beta & \beta \end{pmatrix}$$

with

- $\beta = 0$, Central difference scheme
- $\beta = 1/4$, Average acceleration
- $\beta = 1/10$, Linear acceleration
- $\beta = 1/12$, Fox-Godwin method
- $\beta = 1$, BDF2 method for second order ordinary differential equations.

A.6.2 BDF2 scheme

$$\alpha = \begin{pmatrix} \frac{1}{3} & -\frac{4}{3} & 1 \end{pmatrix} \quad \beta = \begin{pmatrix} 0 & 0 & \frac{2}{3} \end{pmatrix}$$

A.6.3 BDF3 scheme

$$\alpha = \begin{pmatrix} -\frac{2}{11} & \frac{9}{11} & -\frac{18}{11} & 1 \end{pmatrix} \quad \beta = \begin{pmatrix} 0 & 0 & 0 & \frac{6}{11} \end{pmatrix}$$

A.6.4 Adams-Moulton2

$$\alpha = \begin{pmatrix} 0 & -1 & 1 \end{pmatrix} \quad \beta = \begin{pmatrix} -\frac{1}{12} & \frac{2}{3} & \frac{5}{12} \end{pmatrix}$$

A.6.5 Adams-Moulton3

$$\alpha = \begin{pmatrix} 0 & 0 & -1 & 1 \end{pmatrix} \quad \beta = \begin{pmatrix} \frac{1}{24} & -\frac{5}{24} & \frac{19}{24} & \frac{3}{8} \end{pmatrix}$$

A.7 IMEX Runge-Kutta Butcher tableaus

A.7.1 IMEX-θ-method

Implicit-explicit counterpart to the one-step θ-method, presented in [Kot08b]:

$$\alpha = \begin{pmatrix} -1 & 1 \end{pmatrix} \quad \beta_I = \begin{pmatrix} 1-\theta & \theta \end{pmatrix} \quad \delta_I = \begin{pmatrix} 0 & 1 \end{pmatrix}$$
$$\beta_E = \begin{pmatrix} 1 & 0 \end{pmatrix} \qquad \delta_E = \begin{pmatrix} 0 & 1 \end{pmatrix}$$

with $\frac{1}{2} \leq \theta \leq 1$.

A.7.2 IMEX trapezoidal rule

Implicit-explicit counterpart to the trapezoidal rule, to be found in [Kot08b, Kot08a]:

$$\alpha = \begin{pmatrix} -1 & 1 & 0 \\ -\frac{1}{2} & -\frac{1}{2} & 1 \end{pmatrix} \quad \beta_I = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 \end{pmatrix} \quad \delta_I = \begin{pmatrix} 0 & 1 & 1 \end{pmatrix}$$
$$\beta_E = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \end{pmatrix} \quad \delta_E = \begin{pmatrix} 0 & 1 & 1 \end{pmatrix}.$$

A.7.3 IMEX Alexander's two-stage method

Implicit-explicit counterpart to the diagonal implicit two-stage method by Alexander, presented [ARS97]:

$$\alpha = \begin{pmatrix} -1 & 1 & 0 \\ -\frac{1}{2} & -\frac{1}{2} & 1 \end{pmatrix} \quad \beta_I = \begin{pmatrix} 0 & \gamma & 0 \\ 0 & 1-\frac{3\gamma}{2} & \gamma \end{pmatrix} \qquad \delta_I = \begin{pmatrix} 0 & \gamma & 1 \end{pmatrix}$$

$$\beta_E = \begin{pmatrix} \gamma & 0 & 0 \\ \delta - \frac{\gamma}{2} & 1-\delta & 0 \end{pmatrix} \quad \delta_E = \begin{pmatrix} 0 & \gamma & 1 \end{pmatrix}$$

with $\alpha = 2 - \sqrt{2},\ \gamma = \alpha/2,\ \delta = 1 - 1/\alpha$.

A.7.4 IMEX Pareschi2

Second order three-stage method presented in [Kot08b]:

$$\alpha = \begin{pmatrix} -1 & 1 & 0 & 0 \\ -\frac{1}{2} & -\frac{1}{2} & 1 & 0 \\ -\frac{1}{3} & -\frac{1}{3} & -\frac{1}{3} & 1 \end{pmatrix} \quad \beta_I = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & -\frac{7}{6} & \frac{2}{3} & 1 \end{pmatrix} \quad \delta_I = \begin{pmatrix} 0 & 1 & \frac{1}{2} & 1 \end{pmatrix}$$

$$\beta_E = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{1}{2} & 0 & 1 & 0 \end{pmatrix} \quad \delta_E = \begin{pmatrix} 0 & 1 & \frac{1}{2} & 1 \end{pmatrix} .$$

A.7.5 IMEX Ascher3

Third order four-stage method presented first in [ARS97] and then in [BPR13]:

$$\alpha = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 \\ -\frac{1}{2} & -\frac{1}{2} & 1 & 0 & 0 \\ -\frac{1}{3} & -\frac{1}{3} & -\frac{1}{3} & 1 & 0 \\ -\frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} & 1 \end{pmatrix} \quad \beta_I = \begin{pmatrix} 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & -\frac{1}{12} & \frac{1}{2} & 0 & 0 \\ 0 & -\frac{13}{18} & \frac{1}{3} & \frac{1}{2} & 0 \\ 0 & \frac{35}{24} & -\frac{7}{4} & \frac{3}{8} & \frac{1}{2} \end{pmatrix} \quad \delta_I = \begin{pmatrix} 0 & \frac{1}{2} & \frac{2}{3} & \frac{1}{2} & 1 \end{pmatrix}$$

$$\beta_E = \begin{pmatrix} \frac{1}{2} & 0 & 0 & 0 & 0 \\ \frac{13}{36} & \frac{1}{18} & 0 & 0 & 0 \\ \frac{25}{54} & -\frac{23}{27} & \frac{1}{2} & 0 & 0 \\ -\frac{17}{72} & \frac{35}{18} & \frac{5}{8} & -\frac{7}{4} & 0 \end{pmatrix} \quad \delta_E = \begin{pmatrix} 0 & \frac{1}{2} & \frac{2}{3} & \frac{1}{2} & 1 \end{pmatrix} .$$

B Energy spectra

In this part of the appendix we revisit the spectral distribution of kinetic energy, helicity and enstrophy that we have briefly introduced in Section 1.2.3. We derive here relations for the practical computation of these spectra.

B.1 Kinetic energy spectrum

For a velocity field, $E(k,t)$ is numerically computed by taking the Fourier transform of the velocity correlation tensor,

$$E(t) = \frac{1}{2}\sum_{i=1}^{d} \mathrm{Corr}(v_i, v_i)(0,t) \overset{(1.55)}{=} \frac{1}{2}\sum_{i=1}^{d} \int_{\mathbb{R}^d} (\mathcal{F}(\mathrm{Corr}(v_i, v_i)))(k,t)\mathrm{d}k.$$

Now since the velocity field is real-valued we replace the Fourier transform of the correlation tensor in the style of (1.58) by $(\mathcal{F}v_i)(k,t)$ and its complex conjugate,

$$\begin{aligned} E(t) &\sim \frac{1}{2}(2\pi)^d \sum_{i=1}^{d} \int_{\mathbb{R}^d} |(\mathcal{F}v_i)(k',t)|^2 \,\mathrm{d}k' \\ &\sim \frac{1}{2}\int_0^\infty \oint_{\|k'\|_2=k} \sum_{i=1}^{d} |(\mathcal{F}v_i)(k',t)|^2 \,\mathrm{d}k'\mathrm{d}k \\ \Rightarrow E(k,t) &\sim \frac{1}{2}\oint_{\|k'\|_2=k} \sum_{i=1}^{d} |(\mathcal{F}v_i)(k',t)|^2 \,\mathrm{d}k' \,. \end{aligned} \tag{B1.1}$$

Obviously (B1.1) confirms the remark that the kinetic energy density is always real and positive.

B.2 Helicity spectrum

Similar to the kinetic energy, helicity equals the velocity-vorticity correlation tensor at zero separation. Using (1.55) we furthermore obtain

$$H(t) = \frac{1}{2}\sum_{j=1}^{d} \mathrm{Corr}(v_j, \zeta_j)(0,t) = \frac{1}{2}\sum_{j=1}^{d} \int_{\mathbb{R}^d} (\mathcal{F}(\mathrm{Corr}(v_j, \zeta_j)))(k,t)\mathrm{d}k \,.$$

Again as for the kinetic energy spectrum, we replace the correlation between velocity and vorticity in the style of (1.58) and use (1.56) afterwards,

$$\begin{aligned} H(t) &\sim \frac{1}{2}(2\pi)^d \sum_{j=1}^{d} \int_{\mathbb{R}^d} \overline{(\mathcal{F}(v_j))(k,t)}(\mathcal{F}\zeta_j)(k,t)\mathrm{d}k \\ &\sim \frac{1}{2} i (2\pi)^d \int_{\mathbb{R}^d} \overline{(\mathcal{F}(v))(k,t)} \cdot (k \times (\mathcal{F}v)(k,t))\mathrm{d}k \ . \end{aligned}$$

Cyclic permutation in the triple product finally gives

$$H(t) \sim \frac{1}{2} i (2\pi)^d \int_{\mathbb{R}^d} k \cdot ((\mathcal{F}v)(k,t) \times \overline{(\mathcal{F}(v))(k,t)})\mathrm{d}k \ .$$

As above, the helicity spectrum can be obtained by integrating over a sphere of radius k,

$$H(k,t) \sim \frac{1}{2} i \oint_{\|k'\|_2 = k} k' \cdot ((\mathcal{F}v)(k',t) \times \overline{(\mathcal{F}(v))(k',t)}) \, \mathrm{d}k' \ . \tag{B2.2}$$

Note that due to the properties of the triple product the helicity spectrum is real-valued.

B.3 Enstrophy spectrum

Similar to the spectra of kinetic energy and helicity, the enstrophy spectrum is derived from the Fourier transform of vorticity correlation tensor. We use (1.58) and (1.56) to arrive at

$$\mathcal{E}(t) \sim \frac{1}{2}(2\pi)^d \int_{\mathbb{R}^d} (k \times \overline{(\mathcal{F}v)(k,t)}) \cdot (k \times (\mathcal{F}v)(k,t))\mathrm{d}k \ .$$

With Lagrange's identity the integrand can be written as $(k \times \overline{(\mathcal{F}v)(k,t)}) \cdot (k \times (\mathcal{F}v)(k,t)) = k \cdot k \, |(\mathcal{F}v)(k,t)|^2 - (k \cdot (\mathcal{F}v)(k,t))(k \cdot \overline{(\mathcal{F}v)(k,t)})$. Now since $\nabla \cdot v = \mathrm{tr}(\nabla v)$, and because of (1.56) the second term in Lagrange's identity vanishes for divergence-free flow. Therefore it turns out that

$$\mathcal{E}(t) \sim \frac{1}{2}(2\pi)^d \int_{\mathbb{R}^d} k \cdot k \, |(\mathcal{F}v)(k,t)|^2 \, \mathrm{d}k$$

and that the enstrophy spectrum is completely determined by the kinetic energy spectrum, $\mathcal{E}(k,t) = k^2 E(k,t)$.

B.4 Implementation details

For the computation of the fast Fourier transform we assume for simplicity that the velocity field v_h is given on an evenly spaced rectangular/cuboid mesh. Each discrete velocity component is evaluated at different grid points that form by themselves the vertices of an axi-parallel equidistant grid. Note that those points are chosen in such a way that the locations do not coincide with the grid cell interfaces where the approximate

solution is discontinuous. These generated d arrays that contain the point values per component are then processed using the library FFTW3.

The discrete Fourier transform assumes input data in Cartesian coordinates and produces output data in the same coordinate frame. In order to perform the surface integral over a sphere of radius k for data $f(k')$ that is given at discrete Cartesian points, we use the quadrature

$$\oint_{\|k'\|_2=k} f(k')\mathrm{d}k' \approx \frac{4\pi k^2}{\#C(k)} \sum_{k'\in C(k)} f(k')$$

where $C(k) := \{k' \mid k - \frac{1}{2}\Delta k \leq \|k'\|_2 < k + \frac{1}{2}\Delta k\}$. Note that (I) in 2D the factor in the enumerator is replaced by $2\pi k$ and that (II) the band thickness Δk is a natural number. For the quadrature approximation we choose $\Delta k = 2$ as in [BMO$^+$83].

C Surface renewal

Recall the discussion on coherent structures movement in Section 4.7.4. The three-page figure C0.1 displays a second example of such motion.

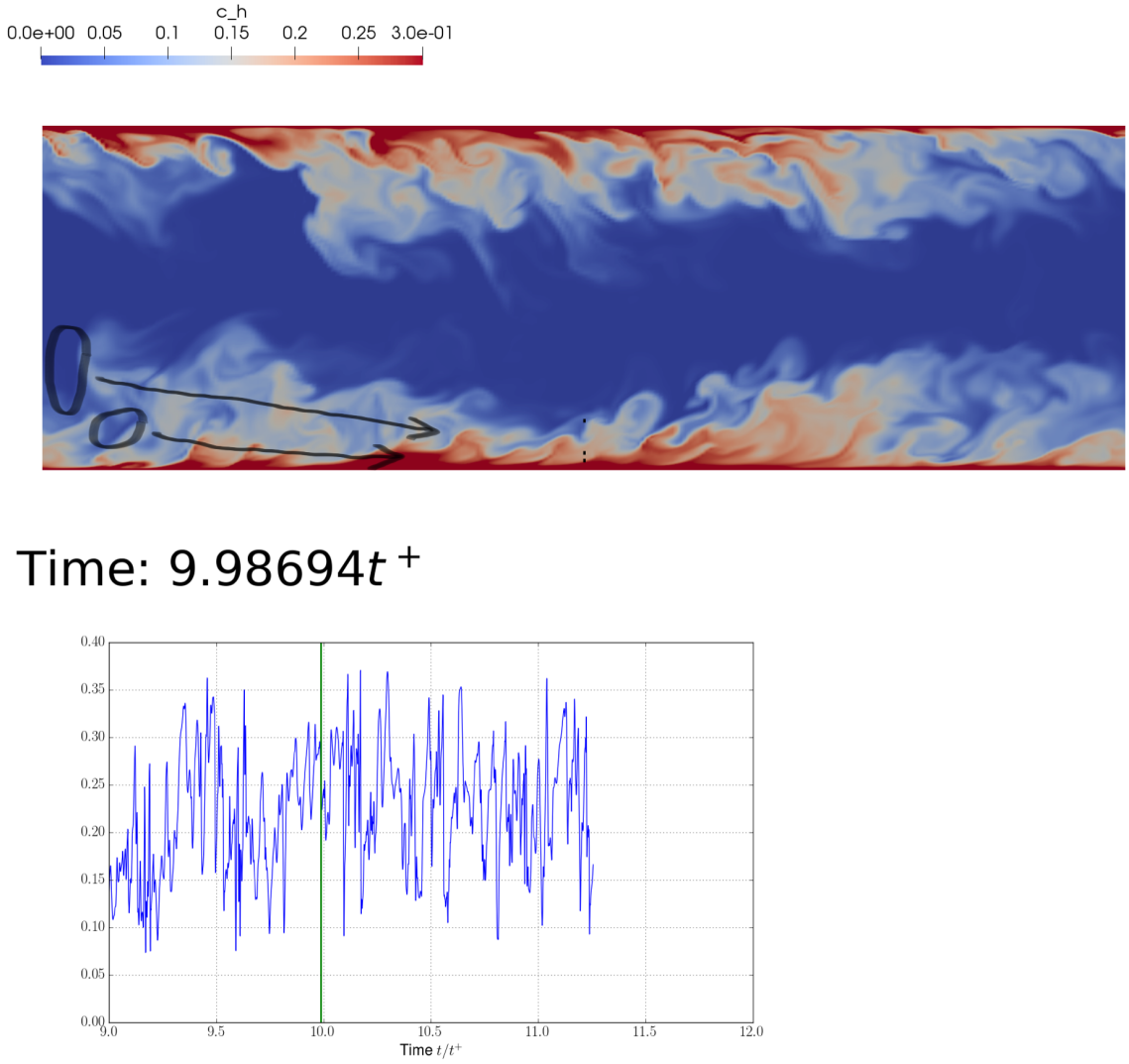


Figure C0.1: Surface renewal event - Part 1: Settling of low concentration parcels indicated by black circles on cross-section $x_3 = \pi\delta/2$. The bottom part shows a time window of the measured concentration at $y/\delta = 0.05$, indicated by the lowermost black square on the cut. The current time of display is marked by a vertical green line.

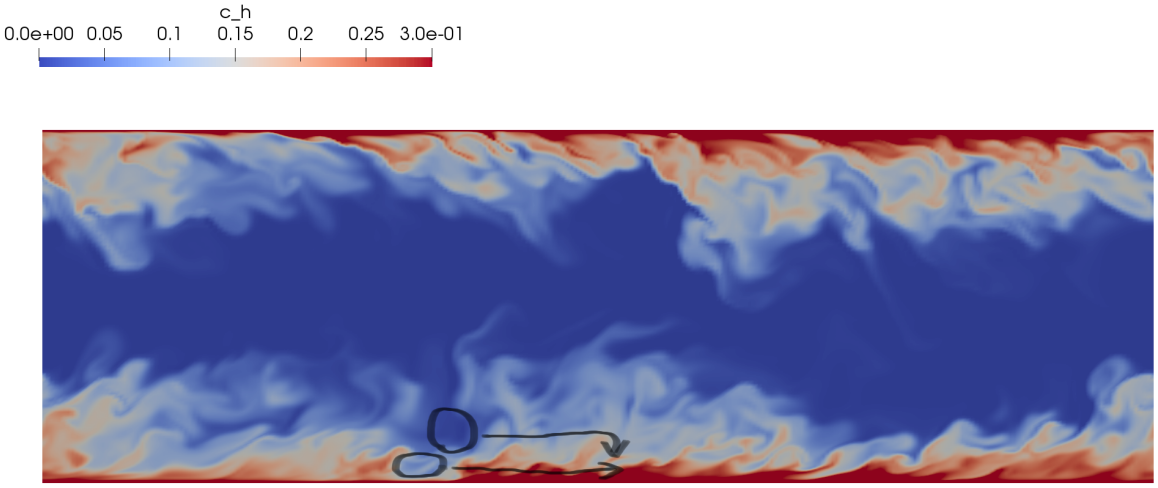


Figure C0.1: Surface renewal event - Part 2: Further settling and joining of low concentration parcels in the top part indicated by black circles and arrows. Here on cross-section $x_3 = 1.55\,\delta$. Middle part shows arrival of concentration parcel at the probe location, again on cross-section $x_3 = \pi\delta/2$. Bottom part displays the corresponding termination of a ramp pattern in the time series.

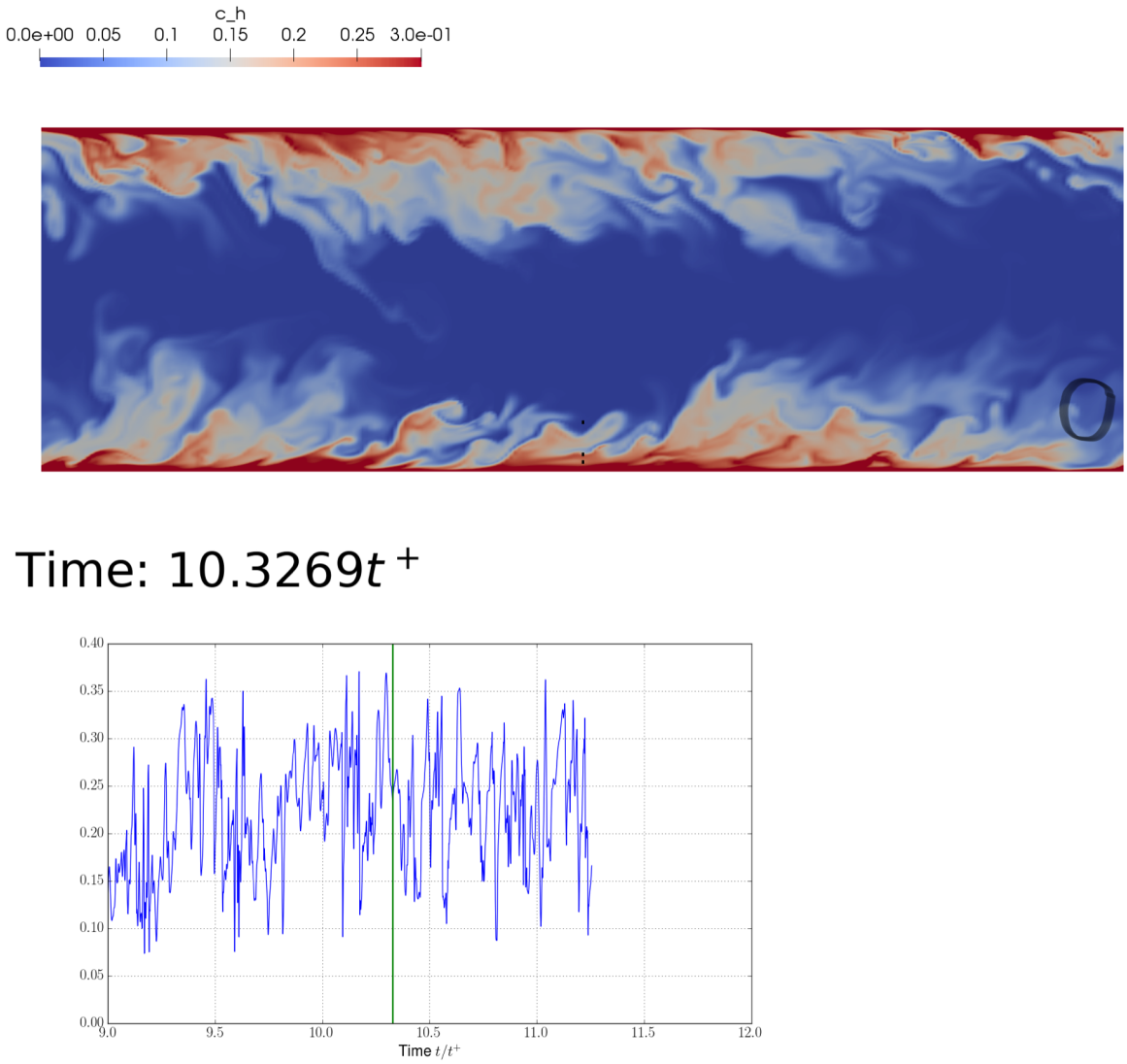


Figure C0.1: Surface renewal event - Part 3: Enrichment of the depleted concentration parcel during residence time in the reference moving in streamwise direction. The black circle indicates the current position after ejection. Meanwhile growth of concentration at fixed position of the lowermost probe.

In the first subfigure (at $t = 9.98694t^+$) two low concentration parcels in the vicinity of the channel half height are marked by black circles. Their travel route is depicted by black arrows and continued in the uppermost part of the second subfigure. Note that the cross-section at time $t = 10.1019t^+$ is at $x_3 = 1.55\,\delta$ to track the low concentration better until it eventually reaches the measurement locations. Just before the arrival - as indicated by the arrows - the upper concentration parcel forms a tail that is slowed down and joins the lower parcel. The middle and bottom part of the second subfigure finally show the surface renewal event, the third subfigure (at $t = 10.3269t^+$) the subsequent growth of concentration and ejection.

Bibliography

[3DT18] *WS2 LES of the plane channel at $Re_\tau = 550$*, January 2018. URL `https://how5.cenaero.be/content/ws2-les-plane-channel-ret550`.

[AE04] Jean-Luc Guermond (auth.) Alexandre Ern: *Theory and Practice of Finite Elements*. Applied Mathematical Sciences 159. Springer-Verlag New York, 1 edition, 2004.

[Ale77] Roger Alexander: *Diagonally Implicit Runge–Kutta Methods for Stiff O.D.E.'s*. SIAM Journal on Numerical Analysis, 14(6):1006–1021, December 1977.

[ARS97] Uri M. Ascher, Steven J. Ruuth and Raymond J. Spiteri: *Implicit-Explicit Runge-Kutta Methods for Time-Dependent Partial Differential Equations*. Appl. Numer. Math, 25:151–167, 1997.

[BBD+08] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, R. Kornhuber, M. Ohlberger and O. Sander: *A generic grid interface for parallel and adaptive scientific computing. Part II: implementation and tests in DUNE*. Computing, 82(2-3):121–138, 2008.

[BBS12] P. Bastian, M. Blatt and R. Scheichl: *Algebraic Multigrid for Discontinuous Galerkin Discretizations*. Numer. Linear Algebra Appl., 19(2):367–388, 2012.

[BF91] Franco Brezzi and Michel Fortin: *Mixed and Hybrid Finite Element Methods*. Springer-Verlag, 1991.

[BHM10] P. Bastian, F. Heimann and S. Marnach: *Generic implementation of finite element methods in the Distributed and Unified Numerics Environment (DUNE)*. Kybernetika, 46(2):294–315, 2010.

[BIB12] Markus Blatt, Olaf Ippisch and Peter Bastian: *A Massively Parallel Algebraic Multigrid Preconditioner based on Aggregation for Elliptic Problems with Heterogeneous Coefficients* . 2012.

[Bla10] Markus Blatt: *A parallel algebraic multigrid method for elliptic problems with highly discontinuous coefficients*. PhD thesis, Heidelberg, Univ., Diss., 2010, 2010.

[BM14] Malte Braack and Piotr Mucha: *Directional Do-Nothing Condition for the Navier-Stokes Equations*. 32:507–521, 09 2014.

[BMMP18] P. Bastian, E. H. Müller, S. Müthing and M. Piatkowski: *Matrix-free multigrid block-preconditioners for higher order Discontinuous Galerkin discretisations*. Submitted to Journal Of Computational Physics, May 2018.

[BMO+83] Marc E. Brachet, Daniel I. Meiron, Steven A. Orszag, B. G. Nickel, Rudolf H. Morf and Uriel Frisch: *Small-scale structure of the Taylor–Green vortex*. Journal of Fluid Mechanics, 130:411–452, 1983.

[BMT11] Evren Bayraktar, Otto Mierka and Stefan Turek: *Benchmark Computations of 3D Laminar Flow Around a Cylinder with CFX, OpenFOAM and FeatFlow*. Technical Report, Fakultät für Mathematik, TU Dortmund, 2011. URL `http://www.featflow.de/en/benchmarks/cfdbenchmarking/flow/dfg_flow3d.html`. Ergebnisberichte des Instituts für Angewandte Mathematik, Nummer 433.

[BNPB13] Harsh Bhatia, Gregory Norgard, Valerio Pascucci and Peer-Timo Bremer: *The Helmholtz-Hodge Decomposition – A Survey*. IEEE Transactions on Visualization and Computer Graphics, 19(8):1386–1404, August 2013. URL `http://dx.doi.org/10.1109/TVCG.2012.316`.

[BPR13] S. Boscarino, L. Pareschi and G. Russo: *Implicit-Explicit Runge–Kutta Schemes for Hyperbolic Systems and Kinetic Equations in the Diffusion Limit*. SIAM Journal on Scientific Computing, 35(1):A22–A51, 2013. URL `https://doi.org/10.1137/110842855`.

[BR03] Peter Bastian and Béatrice Rivière: *Superconvergence and H(div) projection for discontinuous Galerkin methods*. International Journal for Numerical Methods in Fluids, 42(10):1043–1057, 2003. URL `http://dx.doi.org/10.1002/fld.562`.

[Bra91] M.E. Brachet: *Direct simulation of three-dimensional turbulence in the Taylor—Green vortex*. Fluid Dynamics Research, 8(1):1 – 8, 1991. URL `http://www.sciencedirect.com/science/article/pii/016959839190026F`.

[CGS02] Mark A. Christon, Philip M. Gresho and Steven B. Sutton: *Computational predictability of time-dependent natural convection flows in enclosures (including a benchmark solution)*. International Journal for Numerical Methods in Fluids, 40(8):953–980, 2002. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.395`.

[Cho68] Alexandre Joel Chorin: *Numerical Solution of the Navier-Stokes Equations.* Mathematics of Computation, 22(104):745–762, 1968. URL http://www.jstor.org/stable/2004575.

[CHQZ07] Claudio G. Canuto, M. Yousuff Hussaini, Alfio Quarteroni and Thomas A. Zang: *Spectral methods: Evolution to complex geometries and applications to fluid dynamics.* Scientific Computation. Springer, 1 edition, 2007.

[CNBL97a] Wenjun Chen, Michael Novak, T. Andrew Black and Xuhui Lee: *Coherent eddies and temperature structure functions for three contrasting surfaces. Part I: Ramp model with finite microfront time.* 84:99–124, 07 1997.

[CNBL97b] Wenjun Chen, Michael Novak, T. Andrew Black and Xuhui Lee: *Coherent eddies and temperature structure functions for three contrasting surfaces. Part II: Renewal model for sensible heat flux.* 84:125–147, 07 1997.

[DF04] V. Dolejší and M. Feistauer: *A semi-implicit discontinuous Galerkin finite element method for the numerical solution of inviscid compressible flow.* Journal of Computational Physics, 198(2):727 – 746, 2004. URL http://www.sciencedirect.com/science/article/pii/S0021999104000609.

[DHOT09] H. Damanik, J. Hron, A. Ouazzi and S. Turek: *A monolithic FEM-multigrid solver for non-isothermal incompressible flow on general meshes.* Journal of Computational Physics, 228(10):3869 – 3881, 2009. URL http://www.sciencedirect.com/science/article/pii/S0021999109000941.

[DLS90] D. Degani, Y. Levy and A. Seginer: *Graphical visualization of vortical flows by means of helicity.* AIAA Journal, 28:1347–1352, August 1990.

[EL95] Weinan E and Jian-Guo Liu: *Projection Method I: Convergence and Numerical Boundary Layers.* SIAM Journal on Numerical Analysis, 32(4):1017–1057, 1995.

[EL96] Weinan E and Jian-Guo Liu: *Projection Method II: Godunov–Ryabenki Analysis.* SIAM Journal on Numerical Analysis, 33(4):1597–1621, 1996.

[ENV07] Alexandre Ern, Serge Nicaise and Martin Vohralík: *An accurate H(div) flux reconstruction for discontinuous Galerkin approximations of elliptic problems.* Comptes Rendus Mathematique, 345(12):709 – 712, 2007. URL http://www.sciencedirect.com/science/article/pii/S1631073X07004360.

[ES94] C. R. Ethier and D. A. Steinmann: *Exact fully 3D Navier–Stokes solution for benchmarking.* Internat. J. Numer. Methods Fluids, 19:369 – 375, 1994.

[ESW14] Howard C. Elman, David J. Silvester and Andrew J. Wathen: *Finite elements and fast iterative solvers. With applications in incompressible fluid dynamics. 2nd ed.* Oxford: Oxford University Press, 2nd ed. edition, 2014.

[Eva10] L. C. Evans: *Partial Differential Equations.* American Mathematical Society, 2nd edition, 2010.

[FFS03] M. Feistauer, J. Felcman and I. Straskraba: *Mathematical and Computational Methods for Compressible Flow.* Clarendon Press, 2003.

[Fog17] A. Fog: *VCL C++ vector class library*, 2017. URL `http://www.agner.org/optimize/vectorclass.pdf`.

[Fri95] Uriel Frisch: *Turbulence, the legacy of A.N. Kolmogorov.* Cambridge University Press, 1995.

[GB12] Gregor Gassner and Andrea Beck: *On the accuracy of high-order discretizations for underresolved turbulence simulations.* 27, 06 2012.

[GKS11] S. Gottlieb, D.I. Ketcheson and C.W. Shu: *Strong Stability Preserving Runge-Kutta and Multistep Time Discretizations.* World Scientific, 2011.

[GMS05] J. L. Guermond, P. Minev and J. Shen: *Error Analysis of Pressure-Correction Schemes for the Time-Dependent Stokes Equations with Open Boundary Conditions.* SIAM Journal on Numerical Analysis, 43(1):239–258, 2005. URL `http://dx.doi.org/10.1137/040604418`.

[GMS06] J. L. Guermond, P. Minev and Jie Shen: *An overview of projection methods for incompressible flows.* Computer Methods in Applied Mechanics and Engineering, 195:6011–6045, 2006.

[GR86] V. Girault and P.-A. Raviart: *Finite Element Methods for the Navier-Stokes Equations.* Springer, 1986.

[GR09] Christophe Geuzaine and Jean-François Remacle: *Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities.* International Journal for Numerical Methods in Engineering, 79:1309 – 1331, 2009.

[GRMW04] Vivette Girault, Béatrice Rivière, Mary and F. Wheeler: *A discontinuous Galerkin method with nonoverlapping domain decomposition for the Stokes and Navier-Stokes problems.* Math. Comp, pages 53–84, 2004.

[GS03] J. L. Guermond and Jie Shen: *A new class of truly consistent splitting schemes for incompressible flows.* J. Comput. Phys., 192:262–276, 2003.

[GS04] J. L. Guermond and Jie Shen: *On the error estimates of rotational pressure-correction projection methods.* Math. Comp, 73:1719–1737, 2004.

[GSPU89] W. Gao, R. H. Shaw and K. T. Paw U: *Observation of organized structure in turbulent flow within and above a forest canopy.* Boundary-Layer Meteorology, 47(1):349–377, Apr 1989. URL https://doi.org/10.1007/BF00122339.

[Gue15] Jean-Luc Guermond: *Lecture on Mathematical Theory of Finite Elements*, 2015. URL http://www.math.tamu.edu/~guermond/M661_FALL_2015/index.html. Generation of finite elements, Chap4.

[HH08] P. Houston and R. Hartmann: *An optimal order interior penalty discontinuous Galerkin discretization of the compressible Navier-Stokes equations.* J. Comp. Phys., 227:9670–9685, 2008.

[Hil13] Koen Hillewaert: *Development of the discontinuous Galerkin method for high-resolution, large scale CFD and acoustics in industrial geometries.* PhD thesis, Univ. de Louvain, 2013.

[HJ08] Sergio Hoyas and Javier Jiménez: *Reynolds number effects on the Reynolds-stress budgets in turbulent channels.* Physics of Fluids, 20(10):101511, 2008. URL https://doi.org/10.1063/1.3005862.

[HSLR99] R. A. Handler, J. R. Saylor, R. I. Leighton and A. L. Rovelstad: *Transport of a passive scalar at a shear-free boundary in fully developed turbulent open channel flow.* Physics of Fluids, 11(9):2607–2625, 1999. URL https://doi.org/10.1063/1.870123.

[JDS+16] Sumedh M. Joshi, Peter J. Diamessis, Derek T. Steinmoeller, Marek Stastna and Greg 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. URL http://www.sciencedirect.com/science/article/pii/S0045793016301311.

[JH95] Jinhee Jeong and Fazle Hussain: *Hussain, F.: On the identification of a vortex. JFM 285, 69-94.* 285:69 – 94, 02 1995.

[JLM+17] Volker John, Alexander Linke, Christian Merdon, Michael Neilan and Leo G. Rebholz: *On the Divergence Constraint in Mixed Finite Element Methods for Incompressible Flows.* SIAM Review, 59(3):492–544, 2017. URL https://doi.org/10.1137/15M1047696.

[Joh05] Volker John: *On the efficiency of linearization schemes and coupled multigrid methods in the simulation of a 3D flow around a cylinder*. International Journal for Numerical Methods in Fluids, 50(7):845–862, 2005. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.1080.

[JR07] Volker John and Michael Roland: *Simulations of the turbulent channel flow at $Re_\tau = 180$ with projection-based finite element variational multiscale methods*. International Journal for Numerical Methods in Fluids, 55(5):407–429, 2007. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.1461.

[KFWK17] Benjamin Krank, Niklas Fehn, Wolfgang A. Wall and Martin 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. URL http://www.sciencedirect.com/science/article/pii/S0021999117305478.

[KHMB18] Dominic Kempf, René Hess, Steffen Müthing and Peter Bastian: *Automatic Code Generation for High-Performance Discontinuous Galerkin Methods on Modern Architectures*. arXiv e-prints, page arXiv:1812.08075, December 2018.

[KIO91] George Em Karniadakis, Moshe Israeli and Steven A. Orszag: *High-order splitting methods for the incompressible Navier-Stokes equations*. Journal of Computational Physics, 97(2):414 – 443, 1991. URL http://www.sciencedirect.com/science/article/pii/0021999191900078.

[KK04] D.A. Knoll and D.E. Keyes: *Jacobian-free Newton–Krylov methods: a survey of approaches and applications*. Journal of Computational Physics, 193(2):357 – 397, 2004. URL http://www.sciencedirect.com/science/article/pii/S0021999103004340.

[KKSI11] Alireza Kermani, Hamid R. Khakpour, Lian Shen and Takeru Igusa: *Statistics of surface renewal of passive scalars in free-surface turbulence*. Journal of Fluid Mechanics, 678:379–416, 2011.

[KMM87] John Kim, Parviz Moin and Robert Moser: *Turbulence statistics in fully developed channel flow at low Reynolds number*. J. Fluid Mech, 1987.

[Kot08a] Toshiyuki Koto: *IMEX Runge–Kutta schemes for reaction–diffusion equations*. Journal of Computational and Applied Mathematics, 215(1):182 – 195, 2008. URL http://www.sciencedirect.com/science/article/pii/S0377042707001951.

[Kot08b] Toshiyuki Koto: *Stability of IMEX Runge–Kutta methods for delay differential equations*. Journal of Computational and Applied Mathematics, 211(2):201 – 212, 2008. URL http://www.sciencedirect.com/science/article/pii/S0377042706007059.

[KPCS06] G. Katul, A. Porporato, D. Cava and M. Siqueira: *An analysis of intermittency, scaling, and surface renewal in atmospheric surface layer turbulence*. PHYSICA D-NONLINEAR PHENOMENA, 215(2):117–126, 2006.

[KS05] George Em Karniadakis and Spencer J. Sherwin: *Spectral/hp Element Methods for Computational Fluid Dynamics*. Oxford University Press, 2005.

[Les08] Marcel Lesieur: *Turbulence in fluids*. Fluid Mechanics and Its Applications. Springer, 4th, rev. and enlarged ed. edition, 2008.

[Lin14] Alexander Linke: *On the role of the Helmholtz decomposition in mixed methods for incompressible flows and a new variational crime*. Computer Methods in Applied Mechanics and Engineering, 268:782 – 800, 2014. URL http://www.sciencedirect.com/science/article/pii/S0045782513002636.

[LLMS17] Philip L. Lederer, Alexander Linke, Christian Merdon and Joachim Schöberl: *Divergence-free Reconstruction Operators for Pressure-Robust Stokes Discretizations with Continuous Pressure Finite Elements*. SIAM Journal on Numerical Analysis, 55(3):1291–1314, 2017. URL https://doi.org/10.1137/16M1089964.

[LMW12] A. Logg, K.-A. Mardal and G. N. Wells (editors): *Automated Solution of Differential Equations by the Finite Element Method*, volume 84 of *Lecture Notes in Computational Science and Engineering*. Springer, 2012. URL http://dx.doi.org/10.1007/978-3-642-23099-8.

[MKM99] Robert D. Moser, John Kim and Nagi N. Mansour: *Direct numerical simulation of turbulent channel flow up to $Re_\tau = 590$*. Physics of Fluids, 11(4):943–945, 1999. URL https://doi.org/10.1063/1.869966.

[Mof69] H. K. Moffatt: *The degree of knottedness of tangled vortex lines*. Journal of Fluid Mechanics, 35(1):117–129, 1969.

[Mor61] J.J. Moreau: *Constantes d'un îlot tourbillonnaire en fluide parfait barotrope*. 252:2810–2812, 01 1961.

[MPB17] Steffen Müthing, Marian Piatkowski and Peter Bastian: *High-performance Implementation of Matrix-free High-order Discontinuous Galerkin Methods*. Submitted to SIAM SISC, November 2017.

[Nab98] G. Nabh: *On high order methods for the stationary incompressible Navier-Stokes equations*. PhD thesis, Universität Heidelberg, 1998.

[Nek17] Nek5000: *Periodic Hill*, 2017. URL `https://nek5000.github.io/NekDoc/tutorials/perhill.html`. Nek5000 Tutorial.

[Oka17] Jane Okalebo: *Surface Renewal Analysis for Energy Flux Exchanges in an Ecosystem: 1: Calculating Ramp Characteristics using R.*, 2017. URL `https://www.r-bloggers.com/surface-renewal-analysis-for-energy-flux-exchanges-in-an-ecosystem-1-calculating-ramp-characteristics-using-r/`. Article.

[PELS16] Daniele A. Di Pietro, Alexandre Ern, Alexander Linke and Friedhelm Schieweck: *A discontinuous skeletal method for the viscosity-dependent Stokes problem*. Computer Methods in Applied Mechanics and Engineering, 306:175 – 195, 2016. URL `http://www.sciencedirect.com/science/article/pii/S0045782516301189`.

[Per15] *DNS and LES of flow over 2D periodic hill*, January 2015. URL `https://www.grc.nasa.gov/hiocfd/`.

[PMB18] M. Piatkowski, S. Müthing and 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. URL `https://www.sciencedirect.com/science/article/pii/S0021999117308732`.

[QFH16] M. Quadrio, B. Frohnapfel and Y. Hasegawa: *Does the choice of the forcing term affect flow statistics in DNS of turbulent channel flow?* European Journal of Mechanics B Fluids, 55:286–293, January 2016.

[QNE13] *2D Periodic Hill Flow*, July 2013. URL `http://qnet-ercoftac.cfms.org.uk/w/index.php/Abstr:2D_Periodic_Hill_Flow`.

[Ran92] Rolf Rannacher: *On chorin's projection method for the incompressible navier-stokes equations*, pages 167–183. Springer Berlin Heidelberg, Berlin, Heidelberg, 1992. URL `http://dx.doi.org/10.1007/BFb0090341`.

[Ran17] Rolf Rannacher: *Numerik 3: Probleme der Kontinuumsmechanik und ihre numerische Behandlung*. Heidelberg University Publishing, 2017.

[RB06] Patrick Rabenold and Elias Balaras: *Parallel Adaptive Mesh Refinement for the Incompressible Navier-Stokes Equations*. 2006.

[RG06] Béatrice Rivière and Vivette Girault: *Discontinuous finite element methods for incompressible flows on subdomains with non-matching*

interfaces. Computer Methods in Applied Mechanics and Engineering, 195(25–28):3274 – 3292, 2006. URL `http://www.sciencedirect.com/science/article/pii/S0045782505002690`. Discontinuous Galerkin Methods.

[Saa93] Yousef Saad: *A flexible inner-outer preconditioned GMRES algorithm*. SIAM Journal on Scientific Computing, 14(2):461–469, 1993.

[Saa03] Y. Saad: *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2nd edition, 2003.

[Sch13] Ben Schweizer: *Partielle Differentialgleichungen, Eine anwendungsorientierte Einführung*. Springer-Verlag, 2013.

[SJL+18] P. W. Schroeder, V. John, P. L. Lederer, C. Lehrenfeld, G. Lube and J. Schöberl: *On reference solutions and the sensitivity of the 2D Kelvin-Helmholtz instability problem*. ArXiv e-prints, March 2018.

[SO88] Chi W. Shu and Stanley Osher: *Efficient implementation of essentially non-oscillatory shock-capturing schemes*. J. Comput. Phys., 77:439–471, 1988.

[SS86] Youssef Saad and Martin H. Schultz: *GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems*. SIAM Journal on Scientific and Statistical Computing, 7(3):856–869, 1986.

[SS12] Omer San and Anne E. Staples: *High-order methods for decaying two-dimensional homogeneous isotropic turbulence*. Computers & Fluids, 63:105 – 127, 2012. URL `http://www.sciencedirect.com/science/article/pii/S0045793012001363`.

[SSL13] D.T. Steinmoeller, M. Stastna and 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. URL `http://www.sciencedirect.com/science/article/pii/S0021999113004026`.

[ST05] Zdeněk Strakoš and Petr Tichý: *Error Estimation in Preconditioned Conjugate Gradients*. BIT Numerical Mathematics, 45(4):789–817, 2005.

[Tay13] *2nd International Workshop on High-Order CFD Methods, Cologne, Germany*, May 2013. URL `http://www.dlr.de/as/hiocfd`.

[Tay16] *BS1 DNS of the Taylor-Green vortex at Re=1600*, June 2016. URL `https://how4.cenaero.be/content/bs1-dns-taylor-green-vortex-re1600`.

[TG37] G. I. Taylor and A. E. Green: *Mechanism of the Production of Small Eddies from Large Ones.* Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, 158(895):499–521, 1937. URL `http://www.jstor.org/stable/96892`.

[TL72] H. Tennekes and J. L. Lumley: *A First Course in Turbulence.* MIT Press, 1972.

[TMVDV96] L. J. P. Timmermans, P. D. Minev and F. N. Van De Vosse: *An approximate projection scheme for incompressible flow using spectral elements.* International Journal for Numerical Methods in Fluids, 22(7):673–688, 1996.

[TPUSSS] Kyaw Tha Paw U, Richard L. Snyder, Donatella Spano and Hong-Bing Su: *Surface Renewal Estimates of Scalar Exchange.*

[Tre06] L. N. Trefethen: *Spectral Methods in Matlab.* Springer-Verlag, New York, 2006.

[Té69] R. Témam: *Sur l'approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires (II).* Archive for Rational Mechanics and Analysis, 33(5):377–385, 1969. URL `http://dx.doi.org/10.1007/BF00247696`.

[Té87] Roger Témam: *Navier-Stokes Equations. Theory and numerical analysis.* North Holland, Amsterdam, 1987.

[UBC+92] Kyaw Tha Paw U, Yves Brunet, Serge Collineau, Roger H. Shaw, Toshihiko Maitani, Jie Qiu and Lawrence Hipps: *On coherent structures in turbulence above and within agricultural plant canopies.* Agricultural and Forest Meteorology, 61(1):55 – 68, 1992. URL `http://www.sciencedirect.com/science/article/pii/016819239290025Y`.

[VA77] C. W. Van Atta: *Effect of coherent structures on structure functions of temperature in the atmospheric boundary layer.* 29:161–171, 01 1977.

[VK14] A. W. Vreman and J. G. M. Kuerten: *Comparison of direct numerical simulation databases of turbulent channel flow at $Re_\tau = 180$.* Physics of Fluids, 26(1):015102, 2014. URL `https://doi.org/10.1063/1.4861064`.

[Wan09] Wen Wang: *A non-body conformal grid method for simulations of laminar and turbulent flows with a compressible large eddy simulation solver.* PhD thesis, Iowa State University, 2009.

[ZBK17] E. Zorzetto, A. D. Bragg and G. Katul: *Extremes, intermittency and time reversibility of atmospheric turbulence at the cross-over from production to inertial scales.* ArXiv e-prints, October 2017.

[Zie77] O. C. Zienkiewicz: *A new look at the newmark, houbolt and other time stepping formulas. A weighted residual approach.* Earthquake Engineering & Structural Dynamics, 5, 1977.

7011deb