Development and application of high performance software for mantle convection modeling

Jens Weismüller

München 2015

Entwicklung und Anwendung von Hochleistungs-Software für Mantelkonvektionssimulationen

Jens Weismüller

Dissertation
an der Fakultät für Geowissenschaften
der Ludwig–Maximilians–Universität
München

vorgelegt von
Jens Weismüller
aus Ulm

München, den 19. Oktober 2015

Erstgutachter: Prof. Dr. Hans-Peter Bunge
Zweitgutachter: Prof. Dr. Ulrich Rüde
Tag der mündlichen Prüfung: 08. Febraur 2016

Abstract

The Earth mantle convects on a global scale, coupling the stress field at every point to every other location at an instant. This way, any change in the buoyancy field has an immediate impact on the convection patterns worldwide. At the same time, mantle convection couples to processes at scales of a few kilometers or even a few hundred meters. Dynamic topography and the geoid are examples of such small-scale expressions of mantle convection. Also, the depth of phase transitions varies locally, with strong influences on the buoyancy, and thus the global stress field. In order to understand these processes dynamically it is essential to resolve the whole mantle at very high numerical resolutions.

At the same time, geodynamicists are trying to answer new questions with their models, for example about the rheology of the mantle, which is most likely highly nonlinear. Also, due to the extremely long timescales we cannot observe past mantle states, which calls for simulations backwards in time. All these issues lead to an extreme demand in computing power. To cater to those needs, the physical models of the mantle have to be matched with efficient solvers and fast algorithms, such that we can efficiently exploit the enormous computing power of current and future high performance systems.

Here, we first give an extensive overview over the physical models and introduce some numerical concepts to solve the equations. We present a new two-dimensional software as a testbed and elaborate on the implications of realistic mineralogic models for efficient mantle convection simulations. We find that phase transitions present a major challenge and suggest some procedures to incorporate them into mantle convection modeling. Then we give an introduction to the high-performance mantle convection prototype HHG, a multigrid-based software framework that scales to some of the fastest computers currently available. We adapt this framework to a spherical geometry and present first application examples to answer geodynamic questions. In particular, we show that a very thin and very weak asthenosphere is dynamically plausible and consistent with direct and indirect geological observations.

Zusammenfassung

Konvektion im Erdmantel findet auf globalen Skalen statt und koppelt das Spannungsfeld an jedem beliebigen Punkt mit jedem anderen ohne Zeitverzögerung. Dadurch hat jede Änderung im Auftriebsfeld eine sofortige Wirkung auf die Konvektionsströme weltweit. Gleichzeitig gibt es Wechselwirkungen zwischen Mantelkonvektion und Prozessen auf Skalen von wenigen Kilometern oder sogar einigen hundert Metern. Dynamische Topographie und Geoid sind Beispiele für solche kleinskaligen Ausprägungen der Mantelkonvektion. Auch variiert die Tiefe von Phasenübergängen lokal, was einen starken Einfluss auf das Auftriebsfeld und damit das globale Spannungsfeld hat. Grundvorraussetzung, um diese Prozesse dynamisch verstehen zu können, ist eine sehr hohe numerische Auflösung im gesamten Mantel.

Derzeit wenden sich Geodynamiker mit ihren Modellen neuen Fragen zu, zum Beispiel der der Rheologie des Mantels, welche mit hoher Wahrscheinlichkeit stark nichtlinear ist. Auch kann man aufgrund der extrem langen Zeitskalen den Zustand des Mantels in der Vergangenheit nicht beobachten, weshalb Simulationen rückwärts in der Zeit nötig werden. All dies führt zu einer extremen Nachfrage an Rechenleistung. Um dem gerecht zu werden, müssen den physikalischen Modellen des Mantels effiziente Löser und schnelle Algorithmen zur Seite gestellt werden, so dass die enorme Rechenleistung von aktuellen und zukünftigen Hochleistungsrechnern effizient genutzt werden kann.

Wir beginnen diese Arbeit mit einem ausführlichen Überblick über die physikalischen Modelle und einer Einführung in einige der numerischen Konzepte zur Lösung der Gleichungen. Wir stellen dann eine neue zweidimensionale Software als Testbett vor und diskutieren die Implikationen realistischer mineralogischer Modelle für effiziente Mantelkonvektionssimulationen. Wir stellen heraus, dass Phasenübergänge eine groe Herausforderung darstellen, und wir diskutieren einige Möglichkeiten, diese anzugehen, und Phasenübergänge in Mantelkonvektionsmodelle einzubinden. Darauf folgt eine Einführung in den Hochleistungscode HHG als Prototyp für einen Mantelkonvektionscode. HHG ist ein mehrgitterbasiertes Software-Framework, das auf einige der aktuell schnellsten Computer der Welt skaliert. Wir adaptieren HHG auf eine sphärische Geometrie und präsentieren ein erstes Anwendungsbeispiel, in dem wir geodynamische Fragestellungen beantworten. Konkret zeigen wir, dass eine sehr dünne und mechanisch sehr schwache Asthenosphäre dynamisch plausibel und konsistent mit direkten sowie indirekten geologischen Beobachtungen ist.

Contents

Notation

Scalars are typeset in normal letters, 2d and 3d vectors are underlined, and tensors are underlined twice. For vectors and matrices with $n \gg 3$ (mostly in the FEM context) we use boldface. The absolute of a vector is denoted by the same letter as the vector, but is not underlined, except for some exceptions where both meanings are listed separately (j, n, t).

For thermodynamic derivatives, we use the standard notation, where e.g. $\left(\frac{\partial S}{\partial T}\right)_p$ gives the change in entropy with temperature at constant pressure.

Sometimes we split quantities into contributions from a reference state and a deviation from this state. We use the subscripts $\circ_{\mathrm{ref}}$ and $\circ_{\mathrm{dev}}$ to distinguish between them.

The subscript $\circ_{\mathrm{bnd}}$ gives a quantity at any boundary, $\circ_{\mathrm{upper}}$ or $\circ_0$ at the upper boundary, and $\circ_{\mathrm{lower}}$ at the lower boundary. $\circ_{\mathrm{init}}$ denotes a quantity at initial time.

In the finite element section, we use the standard symbols from set theory.
Since chapter 5 is a re-print of a journal article, symbols in this chapter are not listed here and are not consistent with the rest of this thesis.

Units

Since the scales on which mantle convection takes place are far from every day intuitive scales, it is sometimes convenient to use non-SI units. Here we list all non-SI units used in this work, together with a conversion factor.

a	year	$31\,556\,926\,\mathrm{s} \approx \pi \cdot 10^7\,\mathrm{s}$
cm	centimeter	$0.01\,\mathrm{m}$
h	hour	$3600\,\mathrm{s}$
km	kilometer	$1000\,\mathrm{m}$

Abbreviations

ccNUMA	cache-coherent non-uniform memory access, a shared-memory computer architecture
CFL	Courant-Friedrichs-Lewy criterion, limiting the maximal time step
CMB	core-mantle boundary
CG	conjugate gradients, a Krylov subspace method
D”	Perovskite/postperovskite phase transition in the lower mantle
DFG	Deutsche Forschungsgemeinschaft, the German federal research council
FEM	finite element method
Flatland	a 2d mantle convection code for the annulus
Flop/s	floating point operations per second
GMRES	generalized minimum residual, a Krylov subspace method
HHG	hierarchical hybrid grids, a 3d FEM solver framework
HPC	high performance computing
IOCB	inner-outer core boundary
LAB	lithosphere-asthenosphere boundary
LAPACK	Linear Algebra Package, a direct solver
LU decomposition	decomposition of a matrix into lower and upper triangular matrices
MINRES	minimum residual, a Krylov subspace method
MOHO	Mohorovic discontinuity, the crust-mantle boundary
MPI	Message passing interface, a standard parallelization library for distributed memory architectures
OpenMP	Open Multi-Processing, a standard parallelization library for shared memory architectures
P1-P1	linear-linear finite element pair defined on triangular or tetrahedral elements
PARDISO	Parallel Sparse Direct Solver, a direct solver package
PSPG	pressure stabilizing Petrov-Galerkin method
PVM	Parallel Virtual Machine, a programming model for parallel applications
Q1-P0	mixed finite element pair with bilinear velocity on a hexagonal grid and constant pressure
Q1-Q1	linear-linear finite element pair defined on quadrilateral elements
SPPEXA	Priority program Exa-scale computing by the DFG
TERRA	legacy mantle convection code
Terra-neo	DFG funded project to develop a high performance mantle convection code

Upper Case Latin Symbols

$\mathcal{A}$	viscous operator $\left[\frac{\mathrm{Pa\,s}}{\mathrm{m^2}}\right]$
$\mathrm{A}(\circ,\circ)$	bilinear form representing the viscous operator $\mathcal{A}$ $\left[\frac{\mathrm{Pa\,s}}{\mathrm{m^2}}\right]$
$\mathbf{A}$	finite element matrix representing the viscous operator $\mathcal{A}$ $\left[\frac{\mathrm{Pa\,s}}{\mathrm{m^2}}\right]$
$\underline{A}$	area [m^2]
$\mathcal{B}$	negative of the gradient operator [m^{-1}]
$\mathrm{B}(\circ,\circ)$	bilinear form representing the gradient operator $\mathcal{B}$ [m^{-1}]
$\mathbf{B}^T$	finite element matrix representing the gradient operator $\mathcal{B}$ in the Boussinesq approximation [m^{-1}]
$\mathbf{B}$	finite element matrix representing the divergence operator $\mathcal{D}$ [m^{-1}]
$\mathrm{C}_\circ(\circ,\circ)$	bilinear form for the stabilization of the discrete Stokes system [m^{-1}]
$\mathcal{D}$	divergence operator, includes a density scaling for the TALA [m^{-1}]
$\mathrm{D}(\circ,\circ)$	bilinear form representing the divergence operator $\mathcal{D}$ [m^{-1}]
D_t	Lagrangian derivative $\partial_t + \underline{v} \cdot \mathrm{grad}\,(\circ)$ [s^{-1}]
Ek	Ekman number [-]
$\underline{F}$	force [N]
$\mathrm{F}(\circ)$	linear form representing the buoyancy term $\underline{f}$ $\left[\frac{\mathrm{Pa}}{\mathrm{m}}\right]$
$\mathrm{G}_\circ(\circ)$	linear form for the stabilization of the discrete Stokes system [m^{-1}]
H	volumetric radiogenic heat production rate $\left[\frac{\mathrm{J}}{\mathrm{m^3\,s}}\right]$
$H_0^1(\circ)$	Sobolev space H_0^1 on the domain given as parameter
K	bulk modulus [Pa]
K'	bulk modulus' derivative with respect to pressure [-]
$L^2(\circ)$	Hilbert space of all square integrable functions on the domain given as parameter
$\mathbb{N}$	set of natural numbers not including zero
Nu	Nusselt number [-]
Pe	Peclet number [-]
Pr	Prandtl number [-]
Q_ν	solution space for approximated pressure on element ν
R	right hand side of the heat equation [$\mathrm{K\ s}^{-1}$]
$\mathrm{Ra_b}$	Rayleigh number describing bottom heating [-]
$\mathrm{Ra_i}$	Rayleigh number describing internal heating [-]
Re	Reynolds number [-]
$\mathbf{S}$	Schur complement, $\mathbf{S} := -\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^T$ [$\mathrm{Pa}^{-1}\ \mathrm{s}^{-1}$]
T	temperature [K]
$\mathcal{T}$	triangulation of the domain
$\mathcal{T}_{(\circ)}$	elements of the discretized domain
$\mathcal{T}_{(\circ)}^\circ$	interior of the elements
U	internal energy [J]
V	volume of a sample mass m [m^3]
$\underline{W}_\mu$	solution space for approximated velocity on element μ

Lower Case Latin Symbols

$a_{\mu(,i)}$, $b_{\mu(,i)}$, $c_{\mu(,i)}$, $d_{\mu(,i)}$	FEM coefficients, related to the degrees of freedom [-]
c_p	specific heat at constant pressure $\left[\frac{\mathrm{J}}{\mathrm{kg\ K}}\right]$
$\mathrm{curl}\,(\circ)$	curl $[\mathrm{m}^{-1}]$
$\mathrm{div}\,(\circ)$	divergence $[\mathrm{m}^{-1}]$
$\hat{e}_\circ$	unit base vector [-]
$\underline{f}$	buoyancy term $\left[\frac{\mathrm{Pa}}{\mathrm{m}}\right]$
$\underline{\mathbf{f}}$	finite element vector, representing the buoyancy term $\left[\frac{\mathrm{Pa}}{\mathrm{m}}\right]$
$\underline{g}$	gravitational acceleration, pointing radially outwards $\left[\frac{\mathrm{m}}{\mathrm{s}^2}\right]$
$\mathrm{grad}\,(\circ)$	scalar argument: gradient; vectorial argument: vector gradient $[\mathrm{m}^{-1}]$
h	mantle thickness [km]
i, j, k, l, m	spatial indices [-]. $i, j, k, l, m \in \{1, 2, 3\}$ in three dimensions, and $i, j, k, l, m \in \{1, 2\}$ in two dimensions
$\underline{j}$	conductive heat flux $\left[\frac{\mathrm{J}}{\mathrm{m}^2\,\mathrm{s}}\right]$
k	size of a matrix [-]
k	thermal conductivity $\left[\frac{\mathrm{W}}{\mathrm{m\ K}}\right]$
m	sample mass [kg]
n	number of elements of a mesh, $n \in \mathbb{N}$ [-]
n_{level}	number of regular refinements of a mesh [-]
n_r	number of radial nodes of a mesh (annulus or sphere) [-]
n_t	number of tangential nodes of a mesh (annulus or sphere) [-]
$\underline{n}$	unit normal vector, orthogonal to the surface [m]
p	total pressure [Pa]
p'	total pressure when no reference state is assumed, deviatoric pressure otherwise [Pa]
p_ν	discrete version of p' on element ν [Pa]
$\underline{\mathbf{p}}$	coefficient vector, representing the discrete pressure p_ν [Pa]
q	test functions for the pressure [Pa]
q_ν	discrete test functions for the pressure on element ν [Pa]
q'_ν	shape function for the pressure on element ν [Pa]
r	radius / radial coordinate [km]
$\underline{\mathbf{r}}$	right hand side of the pressure correction system, $\underline{\mathbf{r}} := -\mathbf{B}\mathbf{A}^{-1}\underline{\mathbf{f}}$
t	time [a]
$\underline{t}$	tangential vector, unit vector parallel to the surface in 2d [m]
$\underline{u}_\mu$	shape function on element μ $\left[\frac{\mathrm{cm}}{\mathrm{a}}\right]$
$\underline{v}$	velocity $\left[\frac{\mathrm{cm}}{\mathrm{a}}\right]$
$\underline{v}_\mu$	discrete velocity on element μ $\left[\frac{\mathrm{cm}}{\mathrm{a}}\right]$

$\underline{\mathbf{v}}$	coefficient vector, representing the discrete velocity $\underline{v}_\mu$ [Pa]
v_{rms}	root mean square velocity $\left[\frac{\text{cm}}{\text{a}}\right]$
$\underline{w}$	test functions for the velocity $\left[\frac{\text{cm}}{\text{a}}\right]$
$\underline{w}_\mu$	discrete test functions for the velocity on element μ $\left[\frac{\text{cm}}{\text{a}}\right]$
x_i	ith coordinate in any given coordinate system [m], $i \in \{1,2,3\}$ in three dimensions, or $i \in \{1,2\}$ in two dimensions
x	first cartesian coordinate [m]
y	second cartesian coordinate [m]
z	third cartesian coordinate [m]

Upper case Greek Symbols

Δ	Laplace operator [m^{-2}]
Δ_p	Polar Laplacian [m^{-2}], $\Delta_p \circ = \frac{1}{r}\partial_r\left(r\partial_r \circ\right) + \frac{1}{r^2}\partial_\varphi^2 \circ$
ΔE	energy change in a control volume $\left[\frac{\text{J}}{\text{s}}\right]$
ΔT	temperature difference between lower and upper boundary [K]
$\underline{\Delta x}$	spatial displacement [m]
Δx	spatial resolution [m]
$\underline{\Psi}$	stream function $\left[\frac{\text{m}^2}{\text{s}}\right]$
Ω	Domain. Annulus in 2d [m^2], spherical shell in 3d [m^3]

Lower Case Greek Symbols

α	thermal expansion coefficient [K^{-1}]
$\underline{\underline{\alpha}}$	generic tensor
$\underline{\underline{\beta}}$	generic tensor
δ	infinitesimal change
δ_{ij}	Kronecker delta [-]
$\underline{\underline{\varepsilon}}$	strain tensor [-]
$\underline{\underline{\dot{\varepsilon}}}$	strain rate tensor [s^{-1}]
ζ_i	element-local coordinate [m], $i \in \{1,2,3\}$
η	dynamic viscosity [Pa s]
θ	dilatation (average of the diagonal of the strain tensor) [-]
λ	vorticity [s^{-1}]
μ	element index, $\mu \in \mathbb{N}$
ν	element index, $\nu \in \mathbb{N}$
$\underline{\underline{\xi}}$	deviatoric strain [-]

π	pi, ratio of a circle's circumference to its diameter [-]
ρ	density $\left[\frac{\mathrm{kg}}{\mathrm{m}^3}\right]$
$\underline{\underline{\sigma}}$	stress tensor [Pa]
τ	deviatoric stress tensor [Pa]
φ	tangential coordinate in polar coordinates [-]
χ	time stepping index [-]
ψ	iteration index [-]
ω	Earth's angular velocity [h^{-1}]

Other Symbols

$(\cdot)$	between vectors: scalar product in the scientific notation for decimals ($\circ \cdot 10^{(\circ)}$): multiplication between scalars
$(:)$	dyadic double dot product, $\underline{\underline{\alpha}} : \underline{\underline{\beta}} = \sum_{i,j} \alpha_{ij}\beta_{ij}$
$\circ$	placeholder for any quantity
$\propto$	proportionality
∂_t	partial derivative with respect to time [s^{-1}]
∂_r	partial derivative in radial direction [m^{-1}]
$\dot{\circ}$	partial derivative with respect to time [s^{-1}]
$\partial\Omega_{(u/l)}$	domain boundary. Circles in 2d [m], spherical surfaces in 3d [m^2]. Index u stands for upper, index l for lower boundary.

1 Introduction

It is a common view that the Earth's interior consists of three layers: crust, mantle and core. This is based on the two main seismic reflectors within the solid Earth: The Mohorovic discontinuity at $5-10\,\mathrm{km}$ depth below the ocean floor and at $20-90\,\mathrm{km}$ depth below the continents, and the core-mantle boundary at 2890 km depth. Additional significant reflectors have been observed at the transition zone between 410 km and 660 km, at the perovskite/post-perovskite transition roughly 200 km above the core mantle boundary, and at the inner-outer core boundary at 5150 km depth (*Dziewonski and Anderson*, 1981).

For the Earth's internal dynamics, a different structure seems relevant: The outermost part, where brittle behavior is dominant, is called the lithosphere. Below a depth of roughly $50-250\,\mathrm{km}$, the mechanically weak asthenosphere begins. Below $50-600\,\mathrm{km}$ of asthenosphere, at a depth between 200 and 660 km, the mechanically stronger lower mantle starts. The core mantle boundary separates the solid mantle from the liquid outer core, and at the center of the Earth there is the solid inner core. Figure 1.1 gives a schematic representation of this interior structure.

The material within the mantle is solid, and can transmit seismic shear waves. The high pressures below the lithosphere however make it ductile and thus behave like a fluid on long time scales: material can deform permanently without rupturing by migrating defects through the crystal structure. The characteristic time scale that determines the transition from elastic to fluid behavior is called the Maxwell time, and is on the order of 1000 years for the mantle (*Ricard*, 2007). The viscous behavior can be inferred from the response of the solid Earth to the melting of glaciers: the modified load yields a new isostatic state, and from the way the Earth adjusts to this new state one can infer elastic and viscous properties of the underlying material (*Haskell*, 1935; *Cathles*, 1975; *Mitrovica*, 1996).

Radioactive decay and latent heat from the solidification of the inner core are constantly adding heat to the Earth (*Lay et al.*, 2008; *Korenaga*, 2011), heat that has to be transported out through the surface. The efficiency of conductive cooling depends on the length scale: the time to transport a given amount of heat between two points of given temperature increases with the square of the distance between the points. This makes it impossible to cool a body of the size of the Earth conductively, leaving convection as the only possibility to remove heat.

Convection of the Earth's mantle has shaped the surface of the Earth. It is the main driving force behind plate tectonics (*Richardson and Solomon*, 1976; *Bercovici*, 2003) and as such responsible for processes like mountain building, ocean formation, earthquakes and volcanism. The exact mechanisms that couple mantle convection to plate tectonics are debated (see e.g. *Richter*, 1973; *Forsyth and Uyeda*, 1975; *Hager and O'Connell*, 1981; *Lithgow-Bertelloni and Silver*, 1998), but it is clear that buoyancy differences within the mantle are responsible for sustaining tectonic processes.

The mantle is not very accessible to observations. Firstly, it is deeper down into the Earth

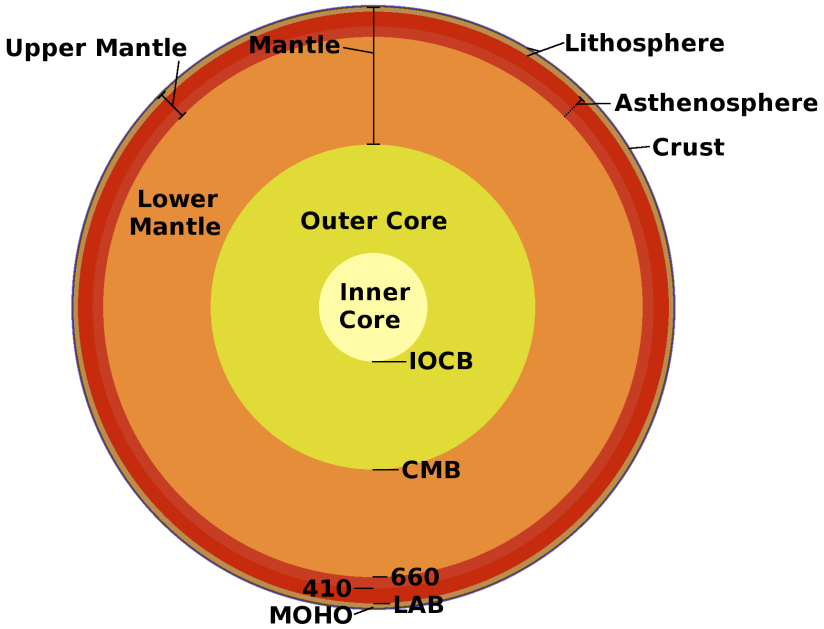

Figure 1.1: Schematic representation of the Earth's interior (to scale). In the upper part, the layers as derived from the main seismic reflectors are shown: The crust, the upper mantle (consisting of the lithospheric mantle and the asthenosphere), the lower mantle, and the outer and inner core. In the lower part, the major seismic discontinuities and dynamic transitions are shown: The Mohorovic discontinuity (MOHO) at $5 - 90$ km depth, the lithosphere-asthenosphere boundary (LAB) at $50 - 250$ km depth, the 410 km and the 660 km phase transitions, the core-mantle boundary at 2890 km depth and the inner-outer core boundary (IOCB) at 5150 km depth.

than anyone has ever been able to drill (*Fuchs et al.*, 1990). Secondly, with typical velocities of a few centimeters per year, the processes are so slow that within a human lifetime we can at best get a peek at one single geological time slice. There is however a number of indirect observations that we can use to infer the current and past state of the mantle, out of which we would like to mention two in particular: For one, seismograms can be used to deduce the traveltimes of seismic waves generated by earthquakes. These can be used to generate tomographic models of wavespeeds within the Earth (*Grand et al.*, 1997; *Ritsema et al.*, 2011). Using models about composition and mineralogical properties of the mantle (*Workman and Hart*, 2005; *Stixrude and Lithgow-Bertelloni*, 2005, 2011; *Chust et al.*, 2015), these wavespeeds can be transformed into estimates of densities and temperatures. Secondly, when oceanic crust forms from upwelling magma at the mid-ocean-ridges, the Earth's magnetic field (generated by the outer core) imprints a residual magnetization. Since the magnetic field changes its orientation periodically, one can use maps of the magnetic orientation of the sea floor to reconstruct past plate motion back in time to approximately 150 Ma (*McElhinny and Lock*, 1996; *Müller et al.*, 2008). This limit comes from the fact that when oceanic crust gets subducted, the magnetic information stored on it gets lost.

Seismic tomographic models now give us information on the current state of the mantle, while paleomagnetic reconstructions can tell us about the evolution of the surface over time. Together, these indirect observations and inferences give us terminal and boundary conditions, which can be incorporated into global mantle circulation models. Other observations like i.e. geodetic information (*Larson et al.*, 1997; *GRACE*, 2002; *Brockmann et al.*, 2014), inferred

dynamic topography (*Winterbourne et al.*, 2009) and past sea levels (*Haq et al.*, 1998) are typically used as independent constraints to ascertain the quality of the simulations.

In order to infer the evolution of the Earth's mantle from terminal and boundary conditions, we also need to know about the governing physics. We have already established that the mantle is solid, not liquid, but behaves like a fluid on long time scales. We can therefore employ the well-established theory of fluid dynamics and use the Navier-Stokes equations (*Landau and Lifshitz*, 1959). Some additional information has to be provided, like i.e. material properties from rheology (*Haskell*, 1935; *Cathles*, 1975; *Mitrovica*, 1996; *Hirth and Kohlstedt*, 2003; *Lenardic et al.*, 2006; *Cordier et al.*, 2012; *Alisic et al.*, 2012), thermodynamical behavior from mineralogy (*Irifune and Tsuchiya*, 2007; *Nakagawa et al.*, 2009; *Stixrude and Lithgow-Bertelloni*, 2005, 2011; *Chust et al.*, 2015) and information about the heating mode (internal heating through radioactive decay and bottom heating from the core, *Davies and Richards*, 1992; *Bunge et al.*, 1997; *Lay et al.*, 2008). Incorporating all this information we can then solve the Navier-Stokes equations numerically with the given constraints (*Gunzburger*, 1989). It is worth mentioning that the very slow speeds within the mantle render inertial forces negligibly small. This allows the simplification of the full Navier-Stokes equations to a Stokes-type set of equations. In this approximation, all changes in the boundary conditions or in the driving buoyancy term propagate through the whole domain instantly (*Iaffaldano and Bunge*, 2015). It is therefore important to consider flow on a global scale, since all local or regional models introduce artificial boundaries, which immediately affect the whole convection pattern.

Mantle convection has been described early with the help of boundary layer theory (*Turcotte and Oxburgh*, 1967; *Schubert et al.*, 1975; *Houseman et al.*, 1981). After the ground work for numerical simulations had been laid out (*McKenzie et al.*, 1974; *Jarvis and McKenzie*, 1980), first global convection models have been presented by *Baumgardner* (1985) and *Glatzmaier* (1988). *Baumgardner* (1985) discretized the planet based on a refined icosahedral mesh with first/zeroth order mixed finite elements, yielding approximately 200 000 degrees of freedom. *Glatzmaier* (1988) used a spectral approach, expanding the variables tangentially into spherical harmonics and radial Chebyshev polynomials, for a total of about one million degrees of freedom. These models were able to produce a planform of up- and downwellings similar to the Earth, but could not account for realistic rheologies and Rayleigh numbers for lack of computing power. With the introduction of commercially available supercomputers such as the Cray T3D PVM or the Intel Paragon, and the newly developed programming models PVM (Parallel Virtual Machine) and MPI (Message Passing Interface) in the 1990s, the necessary computing power became available, and the existing codes were adapted for the new parallel architectures (*Tackley et al.*, 1993; *Bunge and Baumgardner*, 1995). Many more mantle convection codes have been written since, and simulations with more realistic Rayleigh numbers, rheologies and mineralogic properties have been presented, and given dynamical explanations for geological observations (*Tackley*, 1996; *Stuhne and Peltier*, 1996; *Stegman et al.*, 2002; *Zhong et al.*, 2007a; *Schuberth et al.*, 2009; *Tackley*, 2012; *Steinberger and Torsvik*, 2012).

Most of these studies run their models forward in time with linear rheologies, and incorporate simplified thermodynamical properties near the phase transitions. To be able to exploit the knowledge from seismology about the current mantle state in its full extent, simulations need to be performed backwards in time, constituting an inverse problem due to the diffusive nature of heat conduction. Nonlinear rheologies are most likely present in the mantle, posing a great challenge on numerical solvers and requiring additional extensive computing power.

The discontinuities that arise from phase transitions further complicate the matter.

Most of todays codes can resolve the globe to at most 10 km. While this is far beyond the resolution of seismic tomography models of a few hundred kilometers (*Grand et al.*, 1997; *Ritsema et al.*, 2011), seismic reflector depths are known to a much higher precision of a few kilometers (*Dziewonski and Anderson*, 1981). Dynamic topography as a surface expression of mantle convection is on the order of a kilometer (*Winterbourne et al.*, 2009), while the geoid, also dominated by mantle structure, doesn't exceed 100 m (*Brockmann et al.*, 2014). If we want to fully exploit these indirect observables, we need to represent the mantle at resolutions of one kilometer or even less.

In the Earth, up- and downwelling regions, strong viscosity contrasts and phase transitions are highly localized, while large regions with mostly homogeneous properties exist in between. Exploiting this, *Burstedde et al.* (2009, 2013) and *Kronbichler et al.* (2012) have introduced codes that employ adaptive mesh refinement. Here, the domain gets re-meshed every few time steps, automatically adapting to resolve regions with strong gradients at high resolutions, while using much coarser grids in the other regions. This approach allows resolutions of up to a few hundred meters locally, without the need to spend computing time on regions where effectively nothing is happening. On the other hand, this comes with a big administrative overhead, as the mesh has to be regenerated constantly, the solvers need to adapt to the modified grid and the load balance on parallel machines has to be adjusted accordingly. Also, the problem becomes much harder for modern compute architectures, which can deal with uniformly repeating instructions and aligned data much better than with constantly adapting central structures. In addition, the CFL-criterion still limits the time step globally, leading to computational overhead in the more uniform regions.

Apart from some notable exceptions (*Burstedde et al.*, 2009, 2013), most current mantle convection codes do not scale beyond $1000 - 10\,000$ processes. Modern supercomputers on the other hand provide up to several million compute cores and up to 50 petaflop/s peak performance (*Strohmaier et al.*, 2015). Furthermore, in 1965 Moore predicted that the number of transistors in an integrated circuit and thus also the computing speed of a computer doubles approximately every one and a half years, correcting it to two years in 1975. While he did not expect this exponential behavior to continue for more than a decade, it still holds true today, although first indications of a moderate slow down to a two and a half year cadence exist (*Clark*, 2015). But even under this assumption we can expect the first computers with a performance of one exaflop/s around the year 2026. These future machines will be even more parallel, with heterogeneous hardware, where many millions of processing units are combined with accelerators and graphics cards. This poses new challenges for software design, as one will need highly scalable algorithms with new mathematical metrics that are able to deal with localized failures at runtime (resilience). Asynchronous execution will play an ever more important role, and an efficient use of resources will be mandatory. Finally, the enormous amount of data generated will have to be stored and processed in an efficient manner.

The Terra-neo project is part of the SPPEXA priority program funded by the German research council DFG, and aims to develop a new software framework to bring mantle convection to these new machines. As part of this program, we will first discuss the equations that govern mantle convection, and introduce some numerical concepts and algorithms for their solution. We will then introduce the 2d-testbed Flatland developed in the course of this work and show some applications to questions on the mineralogy of the mantle, such as the incorporation of phase changes into mantle convection modeling. We will then elaborate on the Hierarchical Hybrid Grids (HHG) software (*Hülsemann et al.*, 2003; *Bergen and Hülsemann*,

2004; *Bergen*, 2005; *Gradl and Rüde*, 2008; *Gmeiner*, 2013; *Gmeiner et al.*, 2014a) as a prototype for a new high performance mantle convection software framework. We further show an application of HHG to the question of whether fast flow velocities in the upper mantle are dynamically feasible.

2 Governing Equations

For the purpose of mantle convection modeling, we describe the Earth mantle's material as a highly viscous fluid. We will thus disregard plastic deformation and brittle fracture in the lithosphere, which is described by lithospheric models (i.e. *Popov and Sobolev*, 2008; *Gerya and Yuen*, 2007). We also neglect elastic behavior in the form of seismic wave propagation (i.e. *Fowler*, 2005). The governing equations that describe convective flow in the Earth's mantle are thus the Navier-Stokes equations, as described e.g. in *Landau and Lifshitz* (1959). For the derivation of the equations in this chapter we loosely follow the books by *Fowler* (2005), *Davies* (1999) and *Cathles* (1975), essays by *Ricard* (2007) and *Ribe* (2007), the theses written by *Baumgardner* (1983), *Yang* (1997) and *Davies* (2008) and the study by *King et al.* (2010).

2.1 Dimensional Analysis

2.1.1 Simplifications from the Full Navier-Stokes Equations

The density of the mantle ρ is roughly $4000\,\frac{\mathrm{kg}}{\mathrm{m}^3}$ (*Dziewonski and Anderson*, 1981), typical flow velocities are on the order of $v \approx 3\,\frac{\mathrm{cm}}{\mathrm{a}}$ (*Davies*, 1999), the viscosity is on the order of $\eta \approx 10^{21}\,\mathrm{Pa\,s}$ (*Haskell*, 1935) and a typical length scale is the thickness of the mantle, $h = 2\,867\,\mathrm{km}$ (*Dziewonski and Anderson*, 1981). The Reynolds number, which is the ratio of inertial over viscous forces, is thus on the order of

$$\mathrm{Re} = \frac{\rho v h}{\eta} \approx 10^{-20}. \tag{2.1}$$

We will therefore neglect inertial forces as well as kinetic energy.

Similarly, the Prandtl number, which is the ratio of momentum diffusion over thermal diffusion, is given by

$$\mathrm{Pr} = \frac{c_p \eta}{k} \approx 10^{24}, \tag{2.2}$$

with a specific heat of $c_p \approx 1000\,\frac{\mathrm{J}}{\mathrm{kg\ K}}$ (*Ricard*, 2007) and a thermal conductivity of $k \approx 1\,\frac{\mathrm{W}}{\mathrm{m\ K}}$ (*Ricard*, 2007). Momentum diffusion through the mantle can thus be considered immediate.

Finally, we look at the Ekman number, which is the ratio of viscous over Coriolis forces. In a latitude-independent definition it is given by

$$\mathrm{Ek} = \frac{\eta}{\rho \omega h^2} \approx 2.5 \cdot 10^9, \tag{2.3}$$

where $\omega = \frac{1}{24\,\mathrm{h}}$ is the angular velocity of the Earth's rotation. This allows us to also neglect the Coriolis force.

The centrifugal force is isostatically adjusted, transforming the Earth's shape from a sphere to an ellipsoid of rotation. However, neglecting this force as well leads to a geometry of a spherical shell, or, in a two dimensional setting, to an annulus.

2.1.2 Characterizing Mantle Flow

In addition to the very large or very small numbers introduced above, other nondimensional numbers are commonly used to characterize the flow of the mantle (as described i.e. in *Ribe*, 2007). We select those numbers that are important for this work and introduce them here.

There are two driving mechanisms for mantle convection: Heating from the core and internal heating from radioactive decay within the mantle. The respective contributions from these two sources to the total heat flux out of the Earth's crust is debated (*Lay et al.*, 2008; *Korenaga*, 2011). For both heating mechanisms, there is an associated Rayleigh number that describes the vigor of the convection as induced by the corresponding heating mechanism. The Rayleigh number for basal heating is defined as

$$\mathrm{Ra_b} = \frac{\rho^2 g \alpha \Delta T h^3 c_p}{\eta k} \approx 10^8, \tag{2.4}$$

with the gravitational acceleration $\underline{g} \approx 10\,\frac{\mathrm{m}}{\mathrm{s}^2}$, the thermal expansion coefficient $\alpha \approx 2 \cdot 10^{-5}\,\mathrm{K}^{-1}$ (*Ricard*, 2007) and the superadiabatic temperature difference between lower and upper boundary $\Delta T \approx 1500\,\mathrm{K}$ (*Ricard*, 2007). Only above a critical Rayleigh number of roughly 1000, instabilities from the lower boundary amplify and lead to convective cooling (*Turcotte and Oxburgh*, 1967; *Houseman et al.*, 1981; *Baumgardner*, 1985). The Rayleigh number for internal heating is given as

$$\mathrm{Ra_i} = \frac{g \rho^2 \alpha H h^5 c_p}{\eta k^2} \approx 10^{11}, \tag{2.5}$$

with the volumetric radiogenic heat production rate $H \approx 3 \cdot 10^{-7}\,\frac{\mathrm{J}}{\mathrm{m}^3\,\mathrm{s}}$ (*Ricard*, 2007). We note that the relation of the two Rayleigh numbers does not directly allow an inference on the relative heat input from the two sources. Notably, when both sources are equally strong, the internal Rayleigh number is larger than the basal Rayleigh number by about two orders of magnitude (*Bunge et al.*, 1997).

Closely related to the Rayleigh number is the Peclet number. It is defined as the ratio of advective to conductive heat transport rate,

$$\mathrm{Pe} = \frac{v h \rho c_p}{k} \approx 10^4. \tag{2.6}$$

While the Rayleigh numbers are defined purely in terms of physical properties of the mantle, the Peclet number uses the velocity to quantify the contributions by advection, thus relying on a quantity that emerges from these properties. The Rayleigh numbers are thus predetermined and can be used as an input parameter in a nondimensionalized version of the equations, while the Peclet number can be determined from the results of a simulation run. As described in detail in *Davies* (1999), one can derive the following relation between the basal Rayleigh number and the Peclet number from a simple box convection model combined with a halfspace cooling model:

$$\mathrm{Pe} \propto \mathrm{Ra_b}^{2/3} \tag{2.7}$$

This simple relationship can be used as a first plausibility test for mantle convection simulations.

Another related number is the global Nusselt number, which comes from boundary layer theory and is defined as the ratio of convective and conductive heat flux across the mantle,

$$\mathrm{Nu} = -\int_{\partial\Omega_u} \frac{h\partial_r T}{\Delta T} d\Omega \approx 100. \tag{2.8}$$

Here, Ω is the computational domain and $\partial\Omega_u$ its upper boundary. Conductive heat flux is given as the heat that would be transported across the mantle assuming fixed boundary temperatures, whereas convective heat flux through the mantle is given in terms of the temperature gradient at the upper boundary, as a measure for the total (purely conductive) heat flux across the boundary. The high temperature gradient (and thus conductive heat flux) at the boundary is sustained by convection through the mantle. Once again, there is a simple scaling with the basal Rayleigh number,

$$\mathrm{Nu} \propto \mathrm{Ra}_{\mathrm{b}}^{1/3}. \tag{2.9}$$

The Nusselt number is commonly used as a benchmarking quantity (*Blankenbach et al.*, 1998; *King et al.*, 2010).

Another quantity that is commonly used for benchmarking is the root mean square velocity,

$$v_{\mathrm{rms}} = \sqrt{\int_{\partial\Omega} \left(\sum_i v_i^2 \right) d\Omega}. \tag{2.10}$$

Finally, to describe the compressibility of the rock material over the depth of the mantle, the dissipation number is introduced,

$$\mathrm{Di} = \frac{\alpha g h}{c_p} \approx 0.5. \tag{2.11}$$

For incompressible fluids, it is equal to zero.

2.2 First Principle Conservation Equations

With the assumptions stated above, the full Navier-Stokes equations reduce to a Stokes flow problem, consisting of the conservation of momentum and mass. They are coupled with an energy equation and an equation of state,

$$-\mathrm{div}\left(\underline{\underline{\sigma}}\right) - \rho \underline{g} = 0 \tag{2.12}$$

$$\partial_t \rho + \mathrm{div}\left(\rho \underline{v}\right) = 0 \tag{2.13}$$

$$D_t U + \Delta E = 0 \tag{2.14}$$

$$\rho = \rho(p, T). \tag{2.15}$$

Here, $\underline{\underline{\sigma}}$ [Pa] is the stress tensor, $D_t = \partial_t(\circ) + \underline{v} \cdot \mathrm{grad}\,(\circ)$ [s^{-1}] the Lagrangian derivative, U [J] the internal energy, p [Pa] the pressure and T [K] the temperature. $\Delta E \left[\frac{\mathrm{J}}{\mathrm{s}}\right]$ is the energy change within a control volume, and includes conduction, radiogenic heat production, shear heating and compressional heat release.

We will now proceed to explain and reformulate these equations, and introduce some additional assumptions in the process.

2.2.1 Conservation of Momentum

We begin by reformulating the conservation of momentum (2.12),

$$-\mathrm{div}\left(\underline{\underline{\sigma}}\right) - \rho\underline{g} = 0. \tag{2.12 revisited}$$

Stress and Strain

The stress tensor $\underline{\underline{\sigma}}$ is given as the force $\underline{F}$ per unit area $\underline{A}$ in all directions,

$$\underline{\underline{\sigma}} = \frac{\partial \underline{F}}{\partial \underline{A}}. \tag{2.16}$$

We can define the pressure p as the negative of the average of the diagonal elements of $\underline{\underline{\sigma}}$ in three dimensions as

$$p = -\frac{1}{3}(\sigma_{11} + \sigma_{22} + \sigma_{33}) = -\frac{1}{3}\sum_k \sigma_{kk}, \tag{2.17}$$

where $k \in \{1, 2, 3\}$ in three dimensions, and $k \in \{1, 2\}$ and $\sigma_{33} = 0$ in two dimensions.

Subtracting out the pressure (or rather adding it, due to the sign convention) gives the deviatoric stress tensor $\underline{\underline{\tau}}$ [Pa] as

$$\tau_{lm} = \sigma_{lm} + p\delta_{lm} = \sigma_{lm} - \frac{1}{3}\sum_k \sigma_{kk}\delta_{lm}, \tag{2.18}$$

with the Kronecker Delta δ_{lm}, and indices $l, m \in \{1, 2, 3\}$ for the three spatial dimensions, or $l, m \in \{1, 2\}$ in two dimensions.

Next we define the strain tensor $\underline{\underline{\varepsilon}}$ [-] as

$$\varepsilon_{lm} = \frac{1}{2}\left(\frac{\partial \Delta x_l}{\partial x_m} + \frac{\partial \Delta x_m}{\partial x_l}\right), \tag{2.19}$$

where Δx_l and Δx_m are the displacement in direction l and m, respectively.

In analogy to (2.17) we now define the dilatation θ [-] as

$$\theta = \frac{1}{3}(\varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33}) = \frac{1}{3}\sum_k \varepsilon_{kk}, \tag{2.20}$$

which we can subtract out of the strain tensor $\underline{\underline{\varepsilon}}$ to obtain the deviatoric strain $\underline{\underline{\xi}}$ [-],

$$\xi_{lm} = \varepsilon_{lm} - \theta\delta_{lm} = \varepsilon_{lm} - \frac{1}{3}\sum_k \varepsilon_{kk}\delta_{lm}. \tag{2.21}$$

Constitutive Relation

To relate stress and strain rate, we need a constitutive relation. In mantle minerals, there are likely to be two dominating mechanisms: diffusion creep and dislocation creep (*Davies*, 1999). Diffusion creep describes the diffusion of atoms or vacancies through the grain crystals, while dislocation creep describes the migration of higher-dimensional dislocations through the

grains. Diffusion creep is associated with linear stress-strain relationships, while dislocation creep requires a nonlinear constitutive relation. *Karato and Wu* (1993) estimated that the contributions of both mechanisms are of similar order in the upper mantle, while the lower mantle is likely dominated by linear diffusion creep.

For the purpose of this thesis, we will only account for diffusion creep, and neglect nonlinear behavior. Furthermore, we will assume that the viscous resistance to compression is negligible, which causes the bulk viscosity to vanish. With this, we get a linear relation between deviatoric stress and strain rate, with a proportionality factor of 2η, where η is the dynamic viscosity:

$$\begin{aligned}\tau_{lm} &= 2\eta \frac{\partial \xi_{lm}}{\partial t} \\ &= 2\eta \frac{\partial \left(\varepsilon_{lm} - \frac{1}{3}\sum_k \varepsilon_{kk}\delta_{lm}\right)}{\partial t} \\ &= 2\eta \left(\dot{\varepsilon}_{lm} - \frac{1}{3}\sum_k \dot{\varepsilon}_{kk}\delta_{lm}\right). \end{aligned} \tag{2.22}$$

Force Balance Equation

Since the rate of displacement is simply the velocity $\underline{v}$,

$$\frac{\partial \underline{\Delta x}}{\partial t} = \underline{v}, \tag{2.23}$$

we get for the strain rate $\underline{\underline{\dot{\varepsilon}}}$ [s^{-1}]:

$$\dot{\varepsilon}_{lm} = \frac{\partial \varepsilon_{lm}}{\partial t} = \frac{1}{2}\left(\frac{\partial v_l}{\partial x_m} + \frac{\partial v_m}{\partial x_l}\right) \tag{2.24}$$

Plugging (2.24) into (2.22) we obtain

$$\tau_{lm} = \eta \left(\frac{\partial v_l}{\partial x_m} + \frac{\partial v_m}{\partial x_l} - \frac{2}{3}\sum_k \frac{\partial v_k}{\partial x_k}\delta_{lm}\right). \tag{2.25}$$

We can now set up the force balance based on the conservation of momentum (2.12):

$$-\sum_m \frac{\partial \sigma_{lm}}{\partial x_m} - \rho g_l = 0 \tag{2.26}$$

$$-\sum_m \frac{\partial (\tau_{lm} - p\delta_{lm})}{\partial x_m} - \rho g_l = 0 \tag{2.27}$$

$$-\sum_m \frac{\partial \tau_{lm}}{\partial x_m} + \frac{\partial p}{\partial x_l} - \rho g_l = 0 \tag{2.28}$$

$$-\sum_m \frac{\partial}{\partial x_m}\left[\eta\left(\frac{\partial v_l}{\partial x_m} + \frac{\partial v_m}{\partial x_l} - \frac{2}{3}\sum_k \frac{\partial v_k}{\partial x_k}\delta_{lm}\right)\right] + \frac{\partial p}{\partial x_l} - \rho g_l = 0 \tag{2.29}$$

This describes the conservation of momentum for our case. It depicts one equation per spatial dimension, one for each component l.

2.2.2 Conservation of Mass

We have formulated the conservation of mass (2.13) as

$$\partial_t \rho + \operatorname{div}(\rho \underline{v}) = 0. \tag{2.13 revisited}$$

Since flow in the mantle is slow, we neglect the changes in density over time, giving

$$\operatorname{div}(\rho \underline{v}) = 0. \tag{2.30}$$

This is exact for incompressible flow. However, it is a necessary assumption also for compressible flow, since the full continuity equation would introduce unphysical waves. It is worth to note that these waves are not seismic waves, since they originate from the viscous equations (2.12 to 2.15).

2.2.3 Conservation of Energy

We will now proceed to restate the energy equation (2.14)

$$D_t U + \Delta E = 0 \tag{2.14 revisited}$$

in terms of temperature.

First we note that this equation does not contain explicit terms for kinetic or potential energy. As described in section 2.1, the velocities in the mantle are so small that kinetic energy is negligible. Potential energy on the other hand arises from density differences along equipotential surfaces, and has to be accounted for. Looking at the momentum equation (2.12) we note that these density differences are described by the buoyancy term $\rho \underline{g}$, where potential energy is converted into heat by viscous dissipation. We account for the resulting heat within the energy change term ΔE in the form of shear heating.

The Energy Change Term

The change in energy of a sample mass can be due to conduction, radiogenic heat production, shear heating or compression. This gives

$$\Delta E = V \left(\operatorname{div}\left(\underline{j}\right) - H - \sum_{i,j} \sigma_{ij} \dot{\varepsilon}_{ij} + p \operatorname{div}(\underline{v}) \right), \tag{2.31}$$

where V [m^3] is the volume of our sample and $\underline{j}$ $\left[\frac{\mathrm{J}}{\mathrm{m}^2\,\mathrm{s}}\right]$ the conductive heat flux per unit area.

We will now reformulate the compressional term, the last one in (2.31). For this purpose, we consider the change in volume V of an arbitrary, fixed sample mass m [kg] with density ρ in a Lagrangian reference frame:

$$D_t V = m D_t \frac{1}{\rho} = -\frac{m}{\rho^2} D_t \rho = -\frac{m}{\rho^2} \left[\partial_t \rho + \underline{v} \operatorname{grad}(\rho)\right] = -\frac{m}{\rho^2} \left[\partial_t \rho + \operatorname{div}(\rho \underline{v}) - \rho \operatorname{div}(\underline{v})\right] \tag{2.32}$$

Using the continuity equation (2.13) this simplifies to

$$D_t V = -\frac{m}{\rho^2} \left[-\rho \operatorname{div}(\underline{v})\right] = \frac{m}{\rho} \operatorname{div}(\underline{v}) = V \operatorname{div}(\underline{v}) \tag{2.33}$$

or

$$\operatorname{div}(\underline{v}) = \frac{1}{V} D_t V \tag{2.34}$$

We thus get for (2.31):

$$\Delta E = V \left(\operatorname{div}(\underline{j}) - H - \sum_{i,j} \sigma_{ij} \dot{\varepsilon}_{ij} \right) + p D_t V. \tag{2.35}$$

First Law of Thermodynamics

The first law of thermodynamics tells us that the change in internal energy δU can be by either work on the sample or by a heat flux across the border of the sample,

$$\delta U = T \delta S - p \delta V, \tag{2.36}$$

where δ denotes an infinitesimal change in the given quantity and $S \left[\frac{\mathrm{J}}{\mathrm{K}}\right]$ denotes the entropy of the sample.

Entropy can be substituted by

$$\delta S = \left(\frac{\partial S}{\partial T} \right)_p \delta T + \left(\frac{\partial S}{\partial p} \right)_T \delta p, \tag{2.37}$$

and further, using Maxwell's relation $(\partial S/\partial p)_T = -(\partial V/\partial T)_p$ by

$$\delta S = \left(\frac{\partial S}{\partial T} \right)_p \delta T - \left(\frac{\partial V}{\partial T} \right)_p \delta p. \tag{2.38}$$

We note that the specific heat at constant pressure c_p is given by

$$c_p = \frac{T}{m} \left(\frac{\partial S}{\partial T} \right)_p, \tag{2.39}$$

and the thermal expansion coefficient α by

$$\alpha = \frac{1}{V} \left(\frac{\partial V}{\partial T} \right)_p. \tag{2.40}$$

Inserting these definitions into (2.38) we get

$$\delta S = \frac{c_p m}{T} \delta T - \alpha V \delta p, \tag{2.41}$$

and further, inserting into (2.36),

$$\begin{aligned} \delta U &= T \left(\frac{c_p m}{T} \delta T - \alpha V \delta p \right) - p\, \delta V && (2.42)\\ &= c_p m\, \delta T - \alpha T V \delta p - p\, \delta V && (2.43)\\ &= V \left(c_p \rho\, \delta T - \alpha T \delta p \right) - p\, \delta V. && (2.44) \end{aligned}$$

Since we are looking at a sample mass in a Lagrangian reference frame, we can identify the infinitesimal change δ with the Lagrangian derivative D_t and get

$$D_t U = V \left[c_p \rho\, \partial_t T - \alpha T \partial_t p + c_p \rho\, \underline{v} \cdot \operatorname{grad}(T) - \alpha T \underline{v} \cdot \operatorname{grad}(p) \right] - p D_t V. \tag{2.45}$$

Conduction

To describe heat conduction, we use Fourier's law,

$$\underline{j} = -k \operatorname{grad}(T). \tag{2.46}$$

Shear Heating

Using the definitions from section 2.2.1, with some manipulations we can express shear heating in terms of the velocity as

$$\sum_{i,j} \sigma_{ij} \dot{\varepsilon}_{ij} = \eta \sum_{i,j} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) \left(\frac{\partial v_i}{\partial x_j} \right) - \frac{2}{3} \eta \left[\operatorname{div}(\underline{v}) \right]^2 - p \operatorname{div}(\underline{v}). \tag{2.47}$$

Temperature Formulation

Inserting (2.35), (2.45) and (2.46) into the energy conservation equation (2.14) we get

$$\begin{aligned} V \Big[& c_p \rho \, \partial_t T - \alpha T \partial_t p + c_p \rho \, \underline{v} \cdot \operatorname{grad}(T) - \alpha T \, \underline{v} \cdot \operatorname{grad}(p) \\ & - \operatorname{div}(k \operatorname{grad}(T)) - H - \sum_{i,j} \underline{\underline{\sigma}}_{ij} \dot{\varepsilon}_{ij} \Big] = 0. \end{aligned} \tag{2.48}$$

We note that the term that accounts for compressional heating, pD_tV, cancels out. This simply represents the fact that the energy needed to compress a sample mass is equal to the heat that is released by the compression.

We now divide by V and re-order for the time derivative of the temperature to get

$$c_p \rho \, \partial_t T = \alpha T \partial_t p - c_p \rho \, \underline{v} \cdot \operatorname{grad}(T) + \alpha T \, \underline{v} \cdot \operatorname{grad}(p) + \operatorname{div}(k \operatorname{grad}(T)) + H + \sum_{i,j} \sigma_{ij} \dot{\varepsilon}_{ij}. \tag{2.49}$$

For ease of notation, the expansion of the shear heating term (2.47) is not included here.

2.3 Equation of State

We will now look at the equation of state (2.15),

$$\rho = \rho(p, T). \tag{2.15 revisited}$$

This equation constitutes a basic material law and is necessary to solve the system of conservation equations discussed above, (2.12) to (2.14). Several formulations have been suggested, most of which are based on linear thermal expansion, thus containing

$$\frac{\rho}{\rho_0} = -\alpha T \tag{2.50}$$

in some form, where $\rho_0 \approx 3300 \, \frac{\mathrm{kg}}{\mathrm{m}^3}$ is the surface density (*Dziewonski and Anderson*, 1981). To include the pressure dependence, several laws have been proposed, the simplest of which is the linear relationship

$$\frac{p}{K} + 1 = \frac{\rho}{\rho_0} + \alpha T. \tag{2.51}$$

Here, $K \approx 10^{11}$ Pa is the bulk modulus (see e.g. *Ricard*, 2007).

As a refinement, *Murnaghan* (1944) proposed that the bulk modulus itself is a function that depends linearly on pressure. This leads to the relationship

$$\left(\frac{K'}{K}p + 1\right)^{\frac{1}{K'}} = \frac{\rho}{\rho_0} + \alpha T, \tag{2.52}$$

where $K' \approx 4$ (*Birch*, 1947) is the derivative of the bulk modulus with respect to pressure.

An expansion of the nonlinear dependence of the bulk modulus on pressure to higher terms has been presented by *Birch* (1947). Comparisons with experimental data suggest that a good fit to measurements can be achieved by an expansion up to third order, giving for the dependency of pressure on density for a given temperature

$$p = \frac{3}{2}K\left[\left(\frac{\rho}{\rho_0}\right)^{\frac{7}{3}} - \left(\frac{\rho}{\rho_0}\right)^{\frac{5}{3}}\right]\left[1 - \frac{3}{4}(K'-4)\left[\left(\frac{\rho}{\rho_0}\right)^{\frac{2}{3}} - 1\right]\right]. \tag{2.53}$$

Note that we cannot re-arrange this for density, so in this study we use a Newton solver to obtain the scaled density $\frac{\rho}{\rho_0}$. We then account for thermal expansion by subtracting αT according to (2.50).

Alternatively, we can use data from solid state physics models (e.g. *Stixrude and Lithgow-Bertelloni*, 2005; *Piazzoni et al.*, 2007) to obtain the density as a function of temperature and pressure.

2.4 Simplifications and Approximations

2.4.1 The Truncated Anelastic Liquid Approximation (TALA) and the Reference State

A common assumption is to drop the term $\alpha T \partial_t p$ in the energy equation (2.49). For parameters common for the Earth mantle one finds that the magnitude of this term is about one percent of the other terms. *Leng and Zhong* (2008) show that by this we ignore the effect of dynamic pressure on buoyancy, leading to a slight imbalance between gravitational potential energy and viscous dissipation. For this study, we will accept this imbalance, so that the system of equations (2.29), (2.30) and (2.49) reads

$$-\sum_m \frac{\partial}{\partial x_m}\left[\eta\left(\frac{\partial v_l}{\partial x_m} + \frac{\partial v_m}{\partial x_l} - \frac{2}{3}\sum_k \frac{\partial v_k}{\partial x_k}\delta_{lm}\right)\right] + \frac{\partial p}{\partial x_l} - \rho g_l = 0 \tag{2.54}$$

$$\operatorname{div}(\rho \underline{v}) = 0 \tag{2.55}$$

$$-c_p \rho\, \underline{v} \cdot \operatorname{grad}(T) + \alpha T\, \underline{v} \cdot \operatorname{grad}(p) + \operatorname{div}(k \operatorname{grad}(T)) + H + \sum_{i,j} \underline{\underline{\sigma}}_{ij} \dot{\varepsilon}_{ij} = c_p \rho\, \partial_t T \tag{2.56}$$

$$\rho = \rho(p, T). \tag{2.57}$$

We will call these the reference state free anelastic / compressible equations.

A further step that is commonly done is to introduce a reference state. This is motivated by the fact that at any point in the mantle, there is an isostatic, adiabatic contribution to pressure, density and temperature from the load of the material above it. We can subtract

out these contributions and set $T = T_{\mathrm{ref}} + T_{\mathrm{dev}}$, $p = p_{\mathrm{ref}} + p_{\mathrm{dev}}$ and $\rho = \rho_{\mathrm{ref}} + \rho_{\mathrm{dev}}$, where the subscript *ref* denotes the adiabatic and *dev* the deviatoric contribution. Since the reference state is static by definition, it does not contribute to the dynamics, and we can re-write the governing equations in terms of deviatoric quantities:

$$-\sum_m \frac{\partial}{\partial x_m}\left[\eta\left(\frac{\partial v_l}{\partial x_m} + \frac{\partial v_m}{\partial x_l} - \frac{2}{3}\sum_k \frac{\partial v_k}{\partial x_k}\delta_{lm}\right)\right] + \frac{\partial p_{\mathrm{dev}}}{\partial x_l} - \rho_{\mathrm{dev}}\, g_l = 0 \tag{2.58}$$

$$\mathrm{div}\,(\rho_{\mathrm{ref}}\,\underline{v}) = 0 \tag{2.59}$$

$$-c_p\rho\,\underline{v}\cdot\mathrm{grad}\,(T) + \alpha T\,\underline{v}\cdot\mathrm{grad}\,(p_{\mathrm{ref}}) + \mathrm{div}\,(k\,\mathrm{grad}\,(T)) + H + \sum_{i,j}\underline{\underline{\sigma}}_{ij}\,\dot{\varepsilon}_{ij} = c_p\rho\,\partial_t T_{\mathrm{dev}} \tag{2.60}$$

$$\rho_{\mathrm{dev}} = \rho_{\mathrm{dev}}(T_{\mathrm{ref}}, p_{\mathrm{ref}}, \rho_{\mathrm{ref}}, T_{\mathrm{dev}}) \tag{2.61}$$

Note that we have introduced two further assumptions here: First, we assume that the conservation of mass (2.59) is dominated by the reference density. Second, we assume that the pressure is dominated by the adiabatic reference pressure, so that the dynamic pressure in the momentum equation does not feed back into the equations at any other place. In the remainder of this study, we will call these equations (2.58 to 2.61) the Truncated Anelastic Liquid Approximation (TALA), or simply the anelastic / compressible equations.

The reference state is chosen along an adiabate, $\delta U_{\mathrm{ref}} = -p_{\mathrm{ref}}\,\delta V_{\mathrm{ref}}$. This gives $T_{\mathrm{ref}}\delta S_{\mathrm{ref}} = \mathrm{const}$ and thus, since we have no constraint on T_{ref}, $\delta S_{\mathrm{ref}} = 0$. With this we get from (2.41)

$$\frac{c_p m}{T_{\mathrm{ref}}}\delta T_{\mathrm{ref}} = \alpha V_{\mathrm{ref}}\,\delta p_{\mathrm{ref}} \tag{2.62}$$

The only change of quantities in the reference state is in the radial direction, thus we get by integration

$$c_p\partial_r(\log(T_{\mathrm{ref}})) = \frac{\alpha}{\rho_{\mathrm{ref}}}\partial_r(p_{\mathrm{ref}}). \tag{2.63}$$

Since the reference state is static by definition, all terms containing a velocity in the momentum equation (2.54) vanish and we get

$$\partial_r p_{\mathrm{ref}} = -\rho_{\mathrm{ref}}\,\underline{g} \tag{2.64}$$

Together with the equation of state (2.57), these two equations (2.63, 2.64) determine the reference state. For a linear equation of state (2.51), this leads to T_{ref}, ρ_{ref} and p_{ref} increasing exponentially with depth.

2.4.2 The Boussinesq Approximation

While the mantle material is compressible, the density variations are dominated by gravitational compression in radial direction (*Dziewonski and Anderson*, 1981). Lateral density changes are caused by either thermal expansion or chemical heterogeneity and usually vary by less than one percent (e.g. *Schuberth et al.*, 2009). For almost incompressible media like this, a common simplification is the Boussinesq approximation, which is based on the assumption that density variations can be disregarded everywhere in the equations except in the buoyancy term (*Landau and Lifshitz*, 1959). Even though compression in radial direction is significant in the mantle, it does not contribute to the buoyancy, and so the Boussinesq approximation is

quite common in geodynamic modeling (e.g. *Tackley*, 2000; *Zhong et al.*, 2007a; *Alisic et al.*, 2012; *Horbach et al.*, 2014). The equations then read

$$-\sum_m \frac{\partial}{\partial x_m}\left[\eta\left(\frac{\partial v_l}{\partial x_m}+\frac{\partial v_m}{\partial x_l}\right)\right]+\frac{\partial p_{\mathrm{dev}}}{\partial x_l}-\rho_{\mathrm{dev}}\, g_l = 0 \tag{2.65}$$

$$\mathrm{div}\,(\underline{v}) = 0 \tag{2.66}$$

$$-c_p\rho_0\,\underline{v}\cdot\mathrm{grad}\,(T_{\mathrm{dev}})+\mathrm{div}\,(k\,\mathrm{grad}\,(T_{\mathrm{dev}}))+H+\sum_{i,j}\underline{\underline{\sigma}}_{ij}\dot{\varepsilon}_{ij} = c_p\rho_0\,\partial_t T_{\mathrm{dev}} \tag{2.67}$$

$$\frac{\rho_{\mathrm{dev}}}{\rho_0} = -\alpha T_{\mathrm{dev}}. \tag{2.68}$$

The $-\frac{2}{3}\mathrm{div}\,(\underline{v})$ term in the momentum equation (2.58, 2.65) drops out, because the velocity field becomes divergence free due to the continuity equation (2.66). Also, with an adiabate as our reference state we get from (2.41) that $c_p\rho_{\mathrm{ref}}\,\mathrm{grad}\,(T_{\mathrm{ref}}) = -\alpha T_{\mathrm{ref}}\mathrm{grad}\,(p_{\mathrm{ref}})$, which simplifies the heat equation (2.67). For simplicity, one usually also replaces T by T_{dev} in the conduction term $\mathrm{div}\,(k\,\mathrm{grad}\,(T))$, assuming that temperature gradients are insignificant except near the boundaries. With this, the only place left where the reference state shows is in the equation of state. With the common assumption of a first order equation of state (2.51), the reference state vanishes altogether, and since we consider an incompressible medium, the equation of state reduces even further to (2.68).

2.4.3 Further Simplifications

In case the viscosity is constant we can further simplify the momentum equation (2.65). The second term vanishes, because after moving the viscosity in front of the sum it contains a divergence, which is zero in the Boussinesq approximation (cf. 2.65). This can be seen as follows:

$$\sum_m \frac{\partial}{\partial x_m}\frac{\partial v_m}{\partial x_l} = \frac{\partial}{\partial x_l}\sum_m \frac{\partial v_m}{\partial x_m} = \frac{\partial}{\partial x_l}\,\mathrm{div}\,(\underline{v}) = 0. \tag{2.69}$$

If we further substitute the equation of state (2.68) into the momentum equation (2.65), neglect shear heating and internal heating and assume constant thermal conductivity, the full system reads

$$-\eta\Delta\underline{v}+\mathrm{grad}\,(p_{\mathrm{dev}}) = \rho_0\alpha\underline{g}T_{\mathrm{dev}} \tag{2.70}$$

$$\mathrm{div}\,(\underline{v}) = 0 \tag{2.71}$$

$$\partial_t T_{\mathrm{dev}} = -\underline{v}\cdot\mathrm{grad}\,(T_{\mathrm{dev}})+\frac{k}{c_p\rho_0}\Delta T_{\mathrm{dev}}, \tag{2.72}$$

with the Laplace operator Δ.

2.4.4 The Static Equations

We note that we can also solve the system in any approximation without a time dependence by dropping the heat equation. In this case, we obtain equations that represent a problem similar to a Stokes flow problem. We then need a buoyancy distribution as input, which corresponds to the prescription of an initial temperature T_{init}.

2.5 Initial and Boundary Conditions

In all approximations, the governing equations need to be supplemented with initial and boundary conditions, so that they have a unique solution. We need initial and boundary conditions for the temperature, boundary conditions but no initial conditions for the velocity and only a single offset value for the pressure to determine a unique solution. Due to its nature that resembles a physical potential, the pressure is fully determined by the Stokes system except for a constant offset. This offset can be prescribed in several ways, e.g. as an average value or as a fixed potential at a given place.

The simplest choice for velocity boundary conditions are homogeneous Dirichlet conditions,

$$\underline{v}(\underline{x}) = \underline{0}, \quad \underline{x} \in \partial\Omega, \tag{2.73}$$

also called no slip conditions. They are commonly used to describe a stagnant lid in a terrestrial planet (*Solomatov*, 1995). To incorporate plate motions as boundary conditions, one commonly uses inhomogeneous Dirichlet conditions,

$$\underline{v}(\underline{x}) = \underline{v}_{\text{bnd}}(\underline{x}), \quad \underline{x} \in \partial\Omega, \tag{2.74}$$

where $\underline{v}_{\text{bnd}}$ are the given boundary velocities from plate reconstructions (e.g. *Müller et al.*, 2008). Another common choice especially for the core mantle boundary is the free slip condition. It combines a no outflow condition in radial direction with a tangential no stress condition, in 3d

$$\underline{v}(\underline{x}) \cdot \underline{n}(\underline{x}) = 0 \quad \wedge \quad \left(\underline{\underline{\sigma}}(\underline{x}) \cdot \underline{n}(\underline{x})\right) \times \underline{n}(\underline{x}) = \underline{0}, \quad \underline{x} \in \partial\Omega, \tag{2.75}$$

with the surface normal vector $\underline{n}$, or in 2d

$$\underline{v}(\underline{x}) \cdot \underline{n}(\underline{x}) = 0 \quad \wedge \quad \left(\underline{\underline{\sigma}}(\underline{x}) \cdot \underline{n}(\underline{x})\right) \cdot \underline{t}(\underline{x}) = 0, \quad \underline{x} \in \partial\Omega, \tag{2.76}$$

with the tangential vector $\underline{t}$.

For the temperature, one usually either wants to describe a boundary temperature or a heat flux. A fixed temperature is given by inhomogeneous Dirichlet conditions,

$$T(\underline{x}) = T_{\text{bnd}}(\underline{x}), \quad \underline{x} \in \partial\Omega, \tag{2.77}$$

while a heat flux is given by inhomogeneous Neumann conditions,

$$\frac{\partial T(\underline{x})}{\partial \underline{n}} = -\frac{\underline{j}_{\text{bnd}}}{k}, \quad \underline{x} \in \partial\Omega, \tag{2.78}$$

with the boundary heat flux $\underline{j}_{\text{bnd}}$.

Finally, we need an initial buoyancy distribution,

$$T(\underline{x}, t = t_{\text{init}}) = T_{\text{init}}(\underline{x}), \tag{2.79}$$

with time t and the initial time t_{init}. An appropriate choice is a topic of current research (see e.g. *Liu and Gurnis*, 2008; *Horbach et al.*, 2014; *Vynnytska and Bunge*, 2014).

3 Numerical Methods

The equations introduced in the previous chapter do not have analytical solutions except for some special cases (*Landau and Lifshitz*, 1959), which is why we employ numerical solution schemes. The unknowns we solve for are the temperature T, the pressure p, the velocity $\underline{v}$ and the density ρ. In case we have a reference state, we solve for the corresponding deviatoric quantities. Density is directly related to temperature and pressure via the equation of state, so the main focus lies on temperature, pressure and velocity.

The basic solution strategy is to first solve the coupled momentum and mass equations for velocity and pressure with a given initial buoyancy field (i.e. a temperature). The resulting velocity field can then be used to advect the temperature field by a given time step via the heat equation (see e.g. *Davies*, 2008; *Zhong et al.*, 2007b).

If we don't use a reference state, there is a consistency problem in this approach: we need the pressure to obtain the density in the right hand side of the momentum equation from the equation of state, yet the pressure is only obtained after the solution of the Stokes system. While in principle, one would have to iterate over the Stokes equations, for the purpose of this work we simply take the pressure from the previous time step to obtain the right hand side. In case we have a reference state, this problem does not occur, since the equation of state takes the reference pressure as input.

3.1 Stokes System in Operator Notation

To have a general formulation of the Stokes system for all approximations, we reformulate the equations of momentum and mass conservation in operator form,

$$\mathcal{A}\underline{v} + \mathcal{B}p' = \underline{f} \tag{3.1}$$

$$\mathcal{D}\underline{v} = 0. \tag{3.2}$$

For the TALA approximation (2.54 and 2.55), we then have

$$(\mathcal{A}\underline{v})_l = -\sum_m \frac{\partial}{\partial x_m}\left[\eta\left(\frac{\partial v_l}{\partial x_m} + \frac{\partial v_m}{\partial x_l} - \frac{2}{3}\sum_k \frac{\partial v_k}{\partial x_k}\delta_{lm}\right)\right] \tag{3.3}$$

$$\mathcal{B} = \operatorname{grad}(\circ) \tag{3.4}$$

$$p' = p \tag{3.5}$$

$$\underline{f} = \rho\underline{g} \tag{3.6}$$

$$\mathcal{D} = \operatorname{div}(\rho\circ), \tag{3.7}$$

or, if we include the reference state (2.58 and 2.59), in deviation from the above

$$p' = p_{\text{dev}} \tag{3.8}$$
$$\underline{f} = \rho_{\text{dev}} \underline{g} \tag{3.9}$$
$$\mathcal{D} = \operatorname{div}(\rho_{\text{ref}} \circ). \tag{3.10}$$

For the Boussinesq approximation (2.65 and 2.66) we get

$$(\mathcal{A}\underline{v})_l = -\sum_m \frac{\partial}{\partial x_m}\left[\eta\left(\frac{\partial v_l}{\partial x_m} + \frac{\partial v_m}{\partial x_l}\right)\right] \tag{3.11}$$
$$\mathcal{B} = \operatorname{grad}(\circ) \tag{3.12}$$
$$p' = p_{\text{dev}} \tag{3.13}$$
$$\underline{f} = \rho_{\text{dev}} \underline{g} \tag{3.14}$$
$$\mathcal{D} = \operatorname{div}(\circ), \tag{3.15}$$

or, for constant viscosity and a first order equation of state (2.70 and 2.71), deviating from the previous case

$$\mathcal{A} = -\eta\Delta \tag{3.16}$$
$$\underline{f} = \rho_0 \alpha \underline{g} T. \tag{3.17}$$

3.2 Finite Element Discretization of the Stokes System

We want to find an approximate solution to (3.1) and (3.2) together with appropriate boundary conditions on a discrete domain. The idea of the Finite Element Method (also FEM or FE method) is to solve a discretized version of the weak form of the partial differential equations (PDEs) instead. The derivations in this chapter are based on the book by *Braess* (2007), the theses by *Davies* (2008) and *Gmeiner* (2013), and articles by *Zhong et al.* (2007b), *Gmeiner et al.* (2014a) and *Gmeiner et al.* (2014b).

For simplicity, we now assume homogeneous Dirichlet (no-slip) boundary conditions (2.73) and fix the average pressure to zero, $\int_\Omega p' d\Omega = 0$.

We multiply equations (3.1) and (3.2) with test functions $\underline{w} \in \left[H_0^1(\Omega)\right]^3$ and $q \in L^2(\Omega)$ and integrate them over the whole domain. We obtain the weak formulation

$$\int_\Omega \underline{w} \cdot \mathcal{A}\underline{v}\, d\Omega + \int_\Omega \underline{w} \cdot \mathcal{B}p'\, d\Omega = \int_\Omega \underline{w} \cdot \underline{f}\, d\Omega, \tag{3.18}$$
$$\int_\Omega q\, \mathcal{D}\underline{v}\, d\Omega = 0. \tag{3.19}$$

Here, the Hilbert space $L^2(\Omega)$ is the space of all square integrable functions on Ω, and the Sobolev space $H_0^1(\Omega)$ is the subset of $L^2(\Omega)$ that contains only functions that vanish on the boundary and have a derivative which is also square integrable.

Next we divide our domain into $n \in \mathbb{N}$ convex subdomains $\mathcal{T}_\mu$ such that

$$\Omega = \bigcup_{\mu=1}^{n} \mathcal{T}_\mu \quad \text{and} \quad \mathcal{T}_\mu^\circ \cap \mathcal{T}_\nu^\circ = \varnothing \text{ for } \mu \neq \nu, \quad \mu, \nu \in \mathbb{N}. \tag{3.20}$$

We call $\mathcal{T}$ our triangulation of the domain Ω and the subdomains $\mathcal{T}_\mu \subset \mathcal{T}$ our elements. $\mathcal{T}_\mu^\circ$ and $\mathcal{T}^\circ{}_\mu$ are the interiors of the corresponding domains $\mathcal{T}_\mu$ and $\mathcal{T}_\nu$.

We can now discretize the system (3.18) and (3.19) by restricting the solution spaces to finite-dimensional spaces, $\underline{W}_\mu(\mathcal{T}_\mu) \cap \left[H_0^1(\Omega)\right]^3$ and $Q_\nu(\mathcal{T}_\nu) \cap L^2(\Omega)$. The actual choice of these spaces will be discussed in section 3.4.

The problem can now be stated as: Find $(\underline{v}_\mu, p_\nu) \in \underline{W}_\mu \times Q_\nu$ such that

$$\textsc{a}(\underline{v}_\mu, \underline{w}_\mu) + \textsc{b}(\underline{w}_\mu, p_\nu) = \textsc{f}(\underline{w}_\mu), \qquad \forall\, \underline{w}_\mu \in \underline{W}_\mu(\mathcal{T}_\mu) \cap \left[H_0^1(\Omega)\right]^3, \tag{3.21}$$

$$\textsc{d}(\underline{v}_\mu, q_\nu) = 0, \qquad \forall\, q_\nu \in Q_\nu(\mathcal{T}_\nu) \cap L^2(\Omega). \tag{3.22}$$

The exact definitions of the forms $\textsc{a}(\circ,\circ)$, $\textsc{b}(\circ,\circ)$, $\textsc{d}(\circ,\circ)$ and $\textsc{f}(\circ)$ depend on the approximation we make. We will now restrict ourselves to the Bousssinesq approximation with constant viscosity and the first order equation of state, (3.12), (3.13), (3.15), (3.16) and (3.17). In this approximation, the bilinear form $\textsc{a}(\circ,\circ)$ reads

$$\textsc{a}(\underline{v}_\mu, \underline{w}_\mu) = \eta \int_\Omega \operatorname{grad}\left(\underline{v}_\mu\right) : \operatorname{grad}\left(\underline{w}_\mu\right)\, d\Omega, \tag{3.23}$$

where grad $(\circ)$ is the vectorgradient, and $(:)$ the double dot product defined as $\underline{\underline{\alpha}} : \underline{\underline{\beta}} = \sum_{i,j} \alpha_{ij}\beta_{ij}$. This is derived from an integration by parts of the expression

$$\textsc{a}(\underline{v}_\mu, \underline{w}_\mu) = -\eta \int_\Omega \underline{w}_\mu \cdot \Delta \underline{v}_\mu \, d\Omega, \tag{3.24}$$

where the surface term vanishes because of our homogeneous Dirichlet boundary conditions. As an example, in 2d cartesian coordinates $\textsc{a}(\circ,\circ)$ evaluates to

$$\textsc{a}(\underline{v}_\mu, \underline{w}_\mu) = \eta \int_\Omega \left(\frac{\partial v_x}{\partial x}\frac{\partial w_x}{\partial x} + \frac{\partial v_y}{\partial x}\frac{\partial w_y}{\partial x} + \frac{\partial v_x}{\partial y}\frac{\partial w_x}{\partial y} + \frac{\partial v_y}{\partial y}\frac{\partial w_y}{\partial y} \right) d\Omega. \tag{3.25}$$

The bilinear form $\textsc{b}(\circ,\circ)$ in this approximation is given as

$$\textsc{b}(\underline{w}_\mu, p_\nu) = -\int_\Omega \underline{w}_\mu \cdot \operatorname{grad}\left(p_\nu\right)\, d\Omega. \tag{3.26}$$

This time, the need for p_ν to have a first derivative contradicts our requirements on $Q_\nu(\mathcal{T}_\nu)$, which once again we resolve by integrating by parts,

$$\textsc{b}(\underline{w}_\mu, p_\nu) = \int_\Omega p_\nu \operatorname{div}\left(\underline{w}_\mu\right)\, d\Omega. \tag{3.27}$$

Once again, the surface term vanishes because $\underline{w}_\mu$ (as all functions from $\underline{W}_\mu(\mathcal{T}_\mu)$) vanishes on the boundary. As mentioned before, this requirement was introduced because of the no-slip conditions. For other boundary conditions, the surface integrals need to be evaluated.

Next we look at the bilinear form $\textsc{d}(\circ,\circ)$, again in the Bousssinesq approximation with constant viscosity and first order equation of state. It reads

$$\textsc{d}(\underline{v}_\mu, q_\nu) = \int_\Omega q_\nu \operatorname{div}\left(\underline{v}_\mu\right)\, d\Omega. \tag{3.28}$$

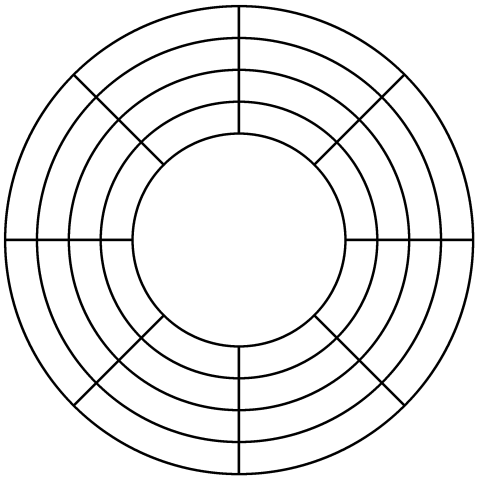

Figure 3.1: Sample triangulation $\mathcal{T}$ of a 2d annulus with $n_r = 5$ and $n_t = 8$. To obtain elements with a better balanced aspect ratio, one will usually use more subdivisions in tangential direction relative to the radial ones than shown here.

We note that $\mathrm{B}(\circ, \circ) = \mathrm{D}(\circ, \circ)$, which makes the system (3.21) and (3.22) a Stokes problem in the strict sense. However, this equality is only true for the Boussinesq approximation, but does not hold for the truncated anelastic approximation (TALA), (2.54) and (2.55).

Finally we consider the linear form $\mathrm{F}(\circ)$, which is given as

$$\mathrm{F}(\underline{w}_\mu) = \int_\Omega \underline{w}_\mu \cdot \underline{f}_\mu \, d\Omega, \tag{3.29}$$

where $\underline{f}_\mu \in \underline{W}_\mu(\mathcal{T}_\mu)$ is a discretized approximation of $\underline{f}$.

3.3 Triangulation

3.3.1 2d Annulus

In two dimensions, we approximate the Earth's mantle as an annulus. We will use a quadrilateral triangulation $\mathcal{T}$, with n_r nodes in radial direction and n_t nodes tangentially. The total number of elements is then $(n_r - 1) \cdot n_t$. An example is shown in figure 3.1.

3.3.2 3d Spherical Shell

In three dimensions, we will regard the Earth's mantle as a thick, spherical shell, neglecting flattening at the poles by rotation and dynamically free deforming surfaces. Similar to *Baumgardner* (1983), *Baumgardner and Frederickson* (1985) and others we will use a triangulation based on an icosahedron, which will be refined tangentially and radially. First, the 20 triangles of the icosahedron are logically combined into 10 diamonds as shown in figure 3.2a. Next, each of the edges is subdivided into n_t nodes, and each triangle is subdivided into $(n_t - 1)^2$ sub-triangles, such that the subdivisions of the edges and the triangles coincide. Finally, each of the new nodes on the triangles and on the edges is projected onto a spherical surface, resulting in meshes as e.g. shown in figure 3.2. To obtain a grid for a thick spherical shell, we now take n_r spherical surfaces with different radii and connect all vertices with their corresponding nodes in the adjacent layers. We now obtain a grid that consists of 10 diamonds, one of which figure 3.3 shows an example of. Each element is now a triangular prismoid. Finally, we divide each prismoid into three tetrahedra as shown in Fig, 3.4. Special

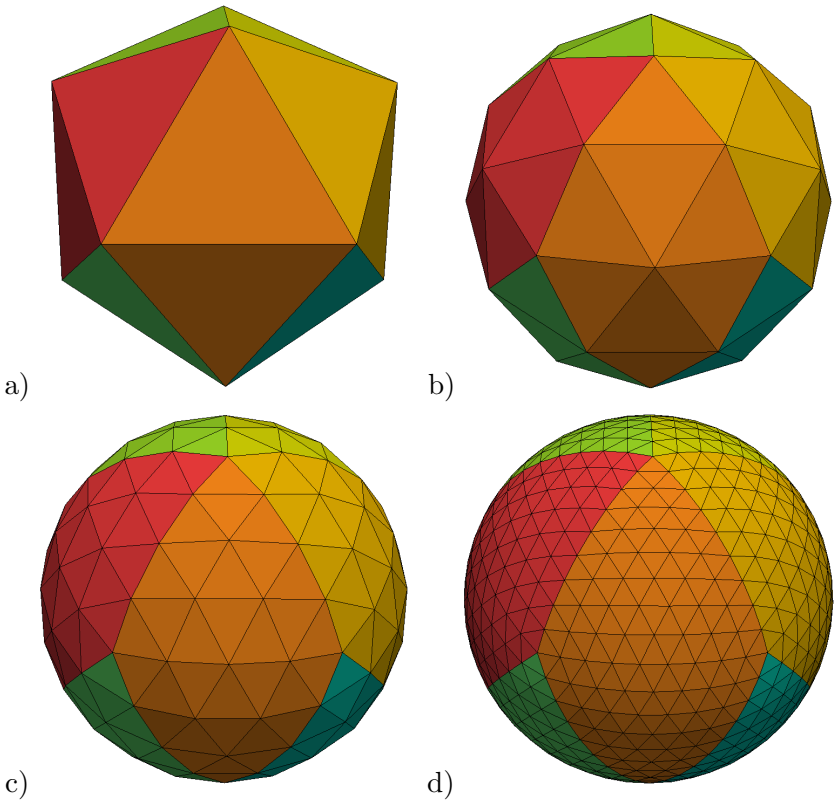

Figure 3.2: Sample triangulations $\mathcal{T}$ of a 3d spherical surface with **a)** $n_t = 2$, **b)** $n_t = 3$, **c)** $n_t = 5$ and **d)** $n_t = 10$. Colored by diamond, where each diamond consists of two triangles on the icosahedron.

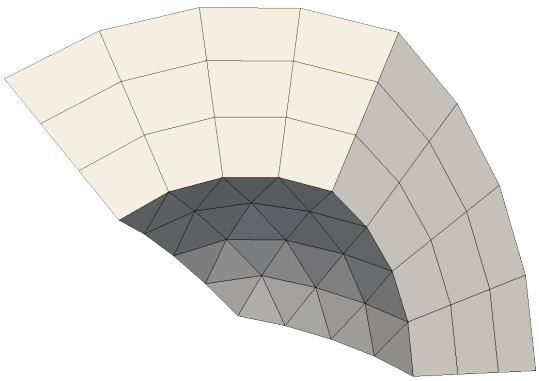

Figure 3.3: Tangentially and radially refined diamond of a mesh with $n_t = 5$ and $n_r = 4$. Equidistance of the radial layers is not required.

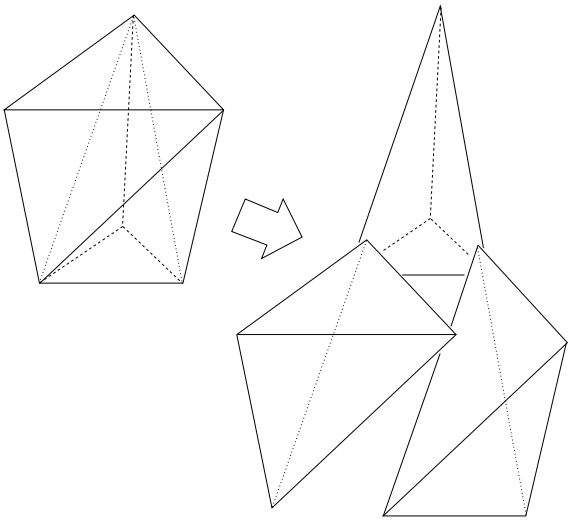

Figure 3.4: Subdivision of a prismoid into three tetrahedra. *Left:* Visualization of the subdivided prismoid. *Right:* Exploded view.

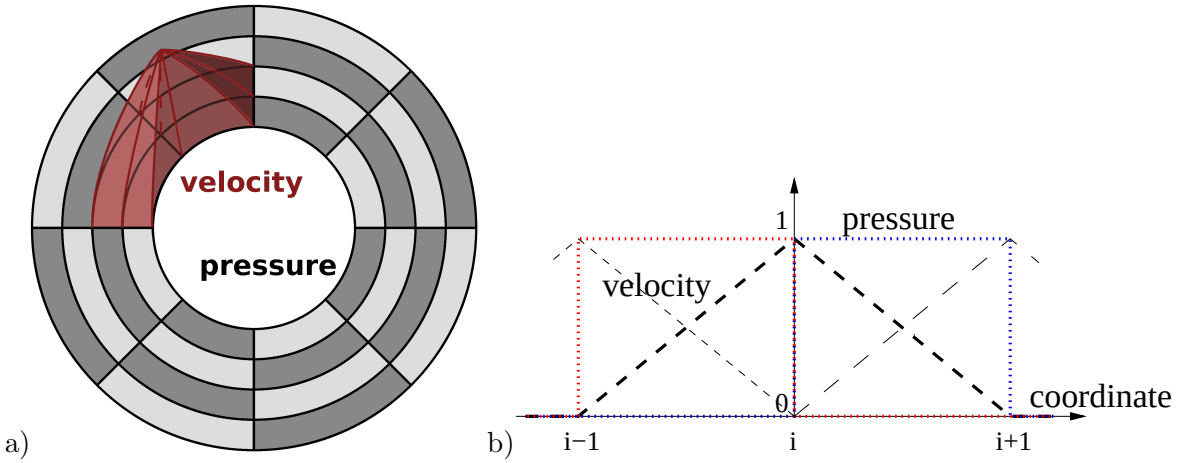

Figure 3.5: Shape functions of the two dimensional Q1-P0 element, spanning the finite element spaces $\underline{W}_\mu(\mathcal{T}_\mu)$ and $Q_\nu(\mathcal{T}_\nu)$. **a)** One component of one velocity shape function is shown in red. The pressure shape functions are drawn in gray shadings, each one has a value of one within exactly one cell and zero everywhere else. **b)** Shape functions in either of the coordinate directions r or φ. Shown is one of the velocity shape functions component (dashed) together with the pressure shape functions (dotted). Nodes are indexed by i.

care has to be taken to make sure that the resulting grid is admissible, i.e. that two adjacent tetrahedra always have common edges.

3.4 Choice of Elements

The quality of the solution and the numerical properties of the solution algorithms strongly depend on the choice of the finite element spaces $\underline{W}_\mu(\mathcal{T}_\mu)$ and $Q_\nu(\mathcal{T}_\nu)$, and there is a vast variety of possible choices in the literature (see e.g. *Zienkiewicz et al.*, 2005; *Braess*, 2007). Here, we will present the element pairs used to obtain the results in the remainder of this thesis: Q1-P0 and Q1-Q1 for the 2d cases, and P1-P1 in three dimensions.

3.4.1 The Q1-P0 Element in 2 Dimensions

In two dimensions, it is convenient to use polar coordinates (r, φ) to define the functions that span our discretized spaces. For the velocity, we will now use functions that are piecewise linear in r and φ, such that for all elements with index $\mu \in \{1, ..., n\}$ we have

$$\underline{W}_\mu(\mathcal{T}_\mu) := \{u_{\mu,1}\hat{e}_1 + u_{\mu,2}\hat{e}_2 \,|\, u_{\mu,i} = a_{\mu,i}r + b_{\mu,i}\varphi + c_{\mu,i} \text{ continuous on } \mathcal{T},\ i \in \{1,2\}\}, \quad (3.30)$$

where $a_{\mu,i}$, $b_{\mu,i}$ and $c_{\mu,i}$ are the coefficients that relate to our degrees of freedom, and $\hat{e}_i$ are the base vectors we use for the two components of the velocity. Note that they do not have to coincide with the polar coordinates we use for the functions that define each of the components. We use the same grid for the pressure and define element-wise constant pressure functions

$$Q_\mu(\mathcal{T}_\mu) := \{q'_\mu \,|\, q'_\mu = \text{const on } \mathcal{T}_\mu\}, \quad \mu \in \{1, ..., n\}. \quad (3.31)$$

We now choose the base vectors that span the velocity space $\underline{W}_\mu(\mathcal{T}_\mu)$ such that they are unequal to zero in at most four adjacent elements, forming a hat shape, while the base

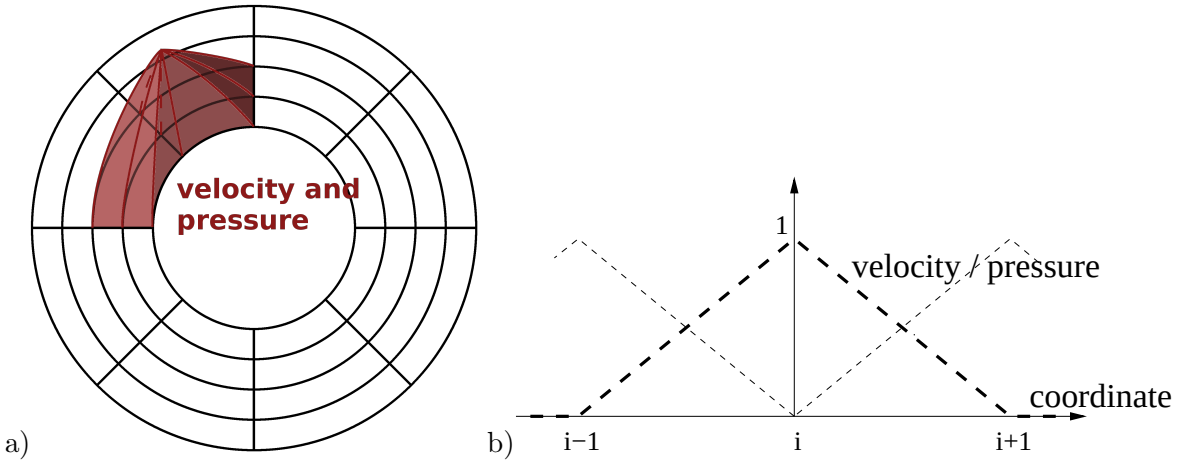

Figure 3.6: Shape functions of the two dimensional Q1-Q1 element, spanning the finite element spaces $\underline{W}_\mu(\mathcal{T}_\mu)$ and $Q_\nu(\mathcal{T}_\nu)$. **a)** The red hat corresponds to one component of a velocity shape function and a single pressure shape function. **b)** Shape functions in either of the coordinate directions r or φ. Shown is one of the shape functions, which can be either a pressure shape function or a single component of a velocity shape function. Nodes are indexed by i.

vectors spanning $Q_\mu(\mathcal{T}_\mu)$ are one in one particular element μ and zero everywhere else. These base vectors are called the shape functions and are visualized in figure 3.5. The test functions $\underline{w}_\mu$ and q_ν are chosen to be equal to the shape functions. The discretized unknowns $\underline{v}_\mu$ and p_μ are expanded into these base functions, which gives us two degrees of freedom per node for the velocity, one for each dimension, and one degree of freedom per element for the pressure. We call this the Q1-P0 element.

3.4.2 The Q1-Q1 Element in 2 Dimensions

For the Q1-Q1 element, we choose the same velocity elements as in the Q1-P0 case, such that again we have for all elements with index $\mu \in \{1, ..., n\}$

$$\underline{W}_\mu(\mathcal{T}_\mu) := \{u_{\mu,1}\hat{e}_1 + u_{\mu,2}\hat{e}_2 \,|\, u_{\mu,i} = a_{\mu,i}r + b_{\mu,i}\varphi + c_{\mu,i} \text{ continuous on } \mathcal{T},\ i \in \{1,2\}\}. \tag{3.30 revisited}$$

For the pressure, we use the same linear elements,

$$Q_\mu(\mathcal{T}_\mu) := \{q'_\mu \,|\, q'_\mu = a_\mu r + b_\mu \varphi + c_\mu \text{ continuous on } \mathcal{T},\ i \in \{1,2\}\}. \tag{3.32}$$

As before we choose a set of local base functions for the shape and the test functions (visualized in figure 3.6). The expansion of $\underline{v}_\mu$ and p_μ into the shape functions now gives us two degrees of freedom per node for the velocity and one degree of freedom per node for the pressure. We call this the Q1-Q1 element.

The Q1-P0 as well as the Q1-Q1 element have the inherent problem that they violate the inf-sup stability condition (see e.g. *Griffiths and Silvester*, 2011), which tends to lead to oscillatory modes in the pressure. While methods to stabilize the system exist (e.g. *Dohrmann and Bochev*, 2004), we find that choosing an uneven number of nodes tangentially n_t is sufficient to suppress the oscillations. Moreover, we are mainly interested in the velocity, which is unaffected by the error in the pressure. More thorough testing of the unstabilized Q1-P0 and Q1-Q1 element pairs is presented in section 4.1.

3.4.3 The P1-P1 Element in 3 Dimensions

For our three dimensional setting, we choose functions that are piecewise linear in an element-local coordinate system spanned by the coordinates ζ_1, ζ_2 and ζ_3. This coordinate system is chosen such that the origin is located at an arbitrary node of the tetrahedron, and the three coordinate axes are oriented along the adjacent tetrahedron edges such that they form a right handed system. We can map any function from these coordinates to a reference tetrahedron defined by the nodes $(0,0,0)$, $(1,0,0)$, $(0,1,0)$ and $(0,0,1)$ in a cartesian reference frame (x_1, x_2, x_3).

For the velocity we now choose our discrete function space such that for all elements with index $\mu \in \{1, ..., n\}$ we have

$$\underline{W}_\mu(\mathcal{T}_\mu) := \{u_{\mu,1}\hat{e}_1 + u_{\mu,2}\hat{e}_2 + u_{\mu,3}\hat{e}_3 \,|\, u_{\mu,i} = a_{\mu,i}\zeta_1 + b_{\mu,i}\zeta_2 + c_{\mu,i}\zeta_3 + d_{\mu,i} \\ \text{continuous on } \mathcal{T},\ i \in \{1,2,3\}\}. \tag{3.33}$$

Since our computational framework HHG (see e.g. *Bergen*, 2005; *Gmeiner*, 2013) restricts us to piecewise linear elements, we choose for the pressure

$$Q_\mu(\mathcal{T}_\mu) := \{q'_\mu \,|\, q'_\mu = a_\mu\zeta_1 + b_\mu\zeta_2 + c_\mu\zeta_3 + d_\mu \text{ continuous on } \mathcal{T}\} \tag{3.34}$$

on the same grid.

Once again we choose a set of local base functions for the shape and the test functions. The expansion of $\underline{v}_\mu$ and p_μ into the shape functions now gives us three degrees of freedom per node for the velocity and one degree of freedom per node for the pressure. We call this the P1-P1 element.

As the Q1-P0 and the Q1-Q1 elements, the P1-P1 element does not satisfy the inf-sup condition and is thus unstable. There are various suggestions for a stabilization in the literature. Here we briefly introduce the two methods that are supported by HHG. Both of them are based on adding additional terms to the discretized momentum equation that are zero in the continuous limit, and thus do not alter the original equations (3.1) and (3.2). The discrete equations (3.21) and (3.22) now read

$$\mathrm{A}(\underline{v}_\mu, \underline{w}_\mu) + \mathrm{B}(\underline{w}_\mu, p_\nu) = \mathrm{F}(\underline{w}_\mu) \tag{3.35}$$

$$\mathrm{D}(\underline{v}_\mu, q_\nu) - \mathrm{C}_\nu(p_\nu, q_\nu) = \mathrm{G}_\nu(q_\nu) \tag{3.36}$$

A common choice for $\mathrm{C}_\nu(p_\nu, q_\nu)$ and $\mathrm{G}_\nu(q_\nu)$ is called the pressure stabilizing Petrov-Galerkin (PSPG) method, where linear least-squares terms are chosen. It has been introduced by *Hughes et al.* (1986), a brief overview of its application in HHG is given e.g. by *Gmeiner et al.* (2014a).

The other approach supported by HHG is described in *Gmeiner et al.* (2014b). It uses a coarser triangulation of the domain Ω for the pressure than for the velocity, thus achieving inf-sup stability. Since HHG uses the multigrid method (see section 3.5.4), the unknowns $\underline{v}$ and p' are already defined on various levels, so this stabilization presents a natural choice.

3.5 Solution Algorithms

For simplicity, we now present the solution strategy for the unstabilized equations (3.21) and (3.22). We can re-write them in matrix-vector form as follows:

$$\begin{pmatrix} \mathbf{A} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \underline{\mathbf{v}} \\ \underline{\mathbf{p}} \end{pmatrix} = \begin{pmatrix} \underline{\mathbf{f}} \\ \underline{\mathbf{0}} \end{pmatrix} \tag{3.37}$$

$\mathbf{A}$, $\mathbf{B}$ and $\mathbf{B}^T$ are matrices that can be calculated by evaluating the forms $\mathrm{A}(\circ,\circ)$, $\mathrm{B}(\circ,\circ)$ and $\mathrm{D}(\circ,\circ)$ with the given shape functions, and factoring out the coefficients that represent the unknowns. They are moved into $\underline{\mathbf{v}}$ and $\underline{\mathbf{p}}$, denoting the coefficient vectors that represent the discrete velocity and pressure $\underline{v}_\mu$ and p_ν, respectively. Note that in the Boussinesq approximation, this procedure leads to two matrices for $\mathrm{B}(\circ,\circ)$ and $\mathrm{D}(\circ,\circ)$ that are the transpose of each other, and we can thus use $\mathbf{B}^T$ and $\mathbf{B}$ to represent them here. Evaluating the linear form $\mathrm{F}(\circ)$ leads to the right hand side vector $\underline{\mathbf{f}}$.

We have now re-formulated the problem in terms of linear algebra and can use computational methods to obtain a solution for $\underline{\mathbf{v}}$ and $\underline{\mathbf{p}}$. These coefficients can then be multiplied with the corresponding shape functions to obtain an approximate solution for the original problem; with all the simplifications made in this chapter, this would be (2.70), (2.71) and (2.73).

3.5.1 Direct Methods

Direct solvers are based on a Gaussian elimination of the matrix in (3.37), which requires $\mathcal{O}(k^3)$ basic operations, where k is the number of rows/columns of the square matrix. In our case, we solve the system by decomposition of the square matrix in (3.37) and subsequent forward- and backward-substitution. In case the matrix does not change in between time steps, we can compute its factorization once and re-use it for all following time steps. This is possible for the Boussinesq approximation and for the TALA with reference state, where constant viscosity in time is required in both cases. Since the computational costs of the substitution only requires $\mathcal{O}(k^2)$ operations, this is much faster than re-computation of the factorization in every time step.

Our 2d mantle convection code Flatland supports two direct solvers, LAPACK (*netlib.org*, 2015) and PARDISO (*Kuzmin et al.*, 2013). While the LAPACK linear algebra package is freely available under a BSD license, it computes an LU decomposition of the full matrix, and thus requires an amount of memory that scales with $\mathcal{O}(k^2)$. Also, it is not parallelized, and can thus not be scaled far beyond the computing power of a typical desktop machine. In practice, this limits calculations to grids with roughly $10,000$ nodes, or about 40 radial layers.

This procedure leaves some potential for optimization, since the test functions have local support, which results in sparse problem matrices. For the Q1-P0 element, an example sparsity pattern is shown in figure 3.7. It is apparent that storing the full matrix is very inefficient with respect to memory usage. Taking advantage of the sparsity, the PARDISO solver package supports the compressed row storage (CRS) format to store the problem matrix. Together with fill-in-reducing permutations of the system, this leads to a greatly decreased memory footprint compared to the straight-forward LU-decomposition. Computationally, PARDISO uses highly involved variants of Gaussian elimination to achieve good runtimes within the $\mathcal{O}(k^3)$ limit for the number of computations. A detailed description of these algorithms is

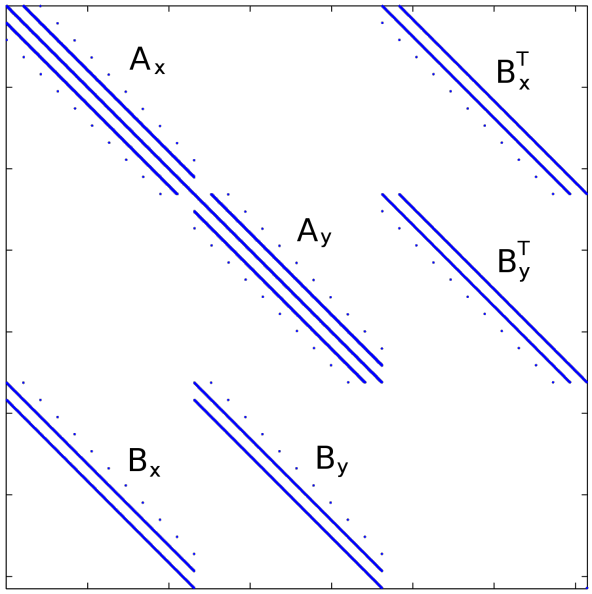

Figure 3.7: Sparsity pattern of a problem matrix from the Q1-P0 element on a grid with $n_t = 11$. Each of the three main bands in $\mathbf{A}$ consists of three adjacent non-zero entries, coupling each node with its nine neighbors. In $\mathbf{B}$ and $\mathbf{B}^T$, each velocity node couples to four pressure cells, two on each of the two bands. The dots outside the main bands stem from the periodic boundary condition in tangential direction. The single dot in the lower right corner represents the regularization of the pressure, fixing one pressure degree of freedom to zero.

given in *Schenk and Gärtner* (2014). Also, the PARDISO solver is parallelized for shared memory architectures. Altogether, it allows us to solve much larger systems than with the LAPACK routines.

3.5.2 Iterative methods

To avoid the $\mathcal{O}(k^3)$ limit for the number of computations, one commonly uses iterative methods. These methods provide an approximate solution through a number of iterative steps, where ideally the error of the solution decreases with each step. There is a huge variety of methods available, most optimized for matrices with certain properties. A very successful method is the multigrid method, which is designed for elliptic problems like the Poisson problem and employs solvers on meshes with varying resolutions. Another common set of methods are the Krylov subspace methods, from which we would like to mention the following: conjugate gradient (CG) methods are designed for symmetric, positive definite matrices. The minimum residual method (MINRES) relaxes this requirement to symmetry only, and the generalized minimum residual method (GMRES) is also able to handle non-symmetric problems. For sparse matrices, all of these methods can find an approximation of the solution within a given error in $\mathcal{O}(k)$ steps. Especially for large systems, they are usually much faster than direct methods.

3.5.3 Pressure Correction Scheme

While HHG is quite flexible with respect to the implementation of new multigrid based solvers, our legacy code TERRA (see e.g. *Baumgardner*, 1983; *Bunge and Baumgardner*, 1995; *Yang*, 1997; *Davies*, 2008) as well as the HHG framework support a pressure correction scheme as introduced by *Verfürth* (1984); *Atanga and Silvester* (1992) and *Ramage and Wathen* (1994). We will briefly discuss this scheme here based on these studies as well as on *Davies* (2008) and *Gmeiner et al.* (2014a).

By applying a block Gaussian elimination step to the system (3.37) we obtain

$$\begin{pmatrix} \mathbf{A} & \mathbf{B}^T \\ \mathbf{0} & -\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^T \end{pmatrix} \begin{pmatrix} \underline{\mathbf{v}} \\ \underline{\mathbf{p}} \end{pmatrix} = \begin{pmatrix} \underline{\mathbf{f}} \\ -\mathbf{B}\mathbf{A}^{-1}\underline{\mathbf{f}} \end{pmatrix}, \tag{3.38}$$

where $\mathbf{S} := -\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^T$ is called the Schur complement. This allows us to isolate a system that we need to solve for pressure only,

$$\mathbf{S}\underline{\mathbf{p}} = \underline{\mathbf{r}}, \tag{3.39}$$

where $\underline{\mathbf{r}} := -\mathbf{B}\mathbf{A}^{-1}\underline{\mathbf{f}}$. Assuming that $\mathbf{S}$ and $\underline{\mathbf{r}}$ are given, this system can then be solved iteratively with a CG method as described by *Verfürth* (1984); *Atanga and Silvester* (1992) or *Ramage and Wathen* (1994). To achieve better convergence, HHG uses a lumped mass matrix as preconditioner (*Gmeiner et al.*, 2014a).

The operations we need to perform within this scheme are the application of $\mathbf{B}$, $\mathbf{B}^T$ and $\mathbf{A}^{-1}$ to different given vectors. We can avoid assembling the matrices altogether by using a matrix-free, stencil based representation of the operators $\mathbf{B}$ and $\mathbf{B}^T$ and $\mathbf{A}$. The inverse of $\mathbf{A}$ is not calculated explicitly, instead we replace each application of $\mathbf{A}^{-1}$ with a geometric multigrid algorithm.

3.5.4 Geometric Multigrid

The systems that remain to be solved are of the form

$$\mathbf{A}\underline{\mathbf{u}} = \underline{\mathbf{g}}, \tag{3.40}$$

where $\underline{\mathbf{g}}$ is a right hand side vector and $\underline{\mathbf{u}}$ the vector of unknowns. The actual definition of these vectors can differ as given by the pressure correction scheme.

One of the central components of the multigrid method is the smoother, e.g. a Jacobi or a Gauss-Seidel relaxation. These smoothers converge to the correct solution for all matrices with a spectral radius of less than one, and are very efficient in eliminating short wavelength errors, but are very slow for low frequency errors. We call an approximate solution to (3.40) after a few smoothing iterations $\underline{\mathbf{u}}_{\text{approx}}$, and note that the algebraic error $\underline{\mathbf{e}}$ is given by

$$\underline{\mathbf{e}} = \underline{\mathbf{u}}^* - \underline{\mathbf{u}}_{\text{approx}}, \tag{3.41}$$

with the exact solution $\underline{\mathbf{u}}^*$. We further define the residual as

$$\underline{\mathbf{r}} = \underline{\mathbf{g}} - \mathbf{A}\underline{\mathbf{u}}_{\text{approx}} \tag{3.42}$$

and find the residual equation

$$\mathbf{A}\underline{\mathbf{e}} = \underline{\mathbf{r}}. \tag{3.43}$$

The central idea of the multigrid method is now to run a few smoothing steps on the initial grid, and then restrict the residual to a coarser mesh. On this mesh we can then solve the residual equation to obtain the error, which we can then prolongate back to the fine grid to correct the unknown $\underline{\mathbf{u}}_{\text{approx}}$. To minimize errors from the prolongation, we once again perform a few smoothing steps. If we now use the same procedure to solve the system on the coarser mesh, we get an iterative scheme that step by step restricts the problem to a much coarser mesh, solves the problem there and then prolongates the solution back, once again

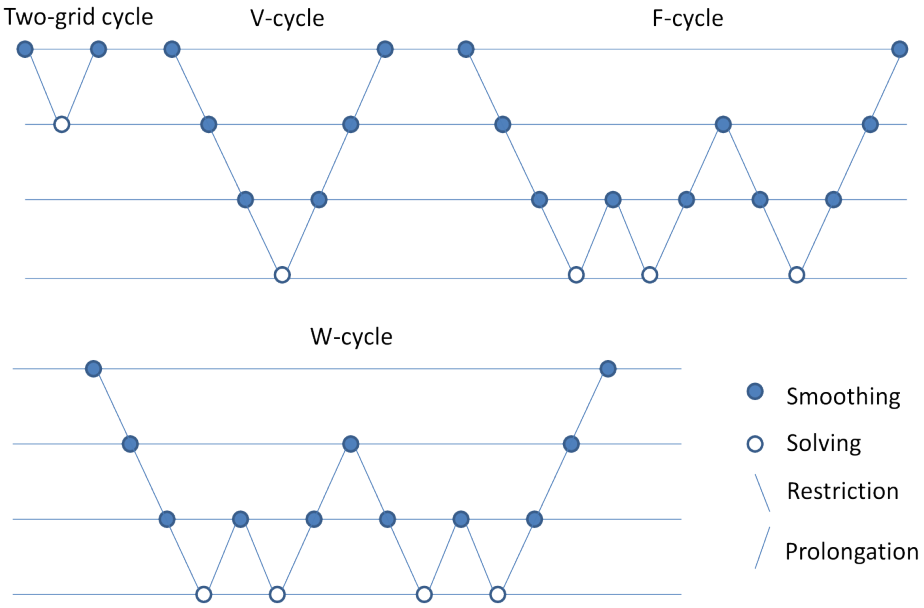

Figure 3.8: Different multigrid cycles with a grid hierarchy of four levels. Reprinted with kind permission from *Gmeiner* (2013).

step by step. On the coarsest mesh, we still need a solver, but since for a three-dimensional domain the problem size is reduced by about a factor of eight for each restriction, this solution process is much more inexpensive than a direct solution of the full system. For the restriction and prolongation, several methods have been proposed, but most of them are based on simple low-order interpolation.

We can traverse the different grid levels in various manners, a selection of common choices is visualized in figure 3.8. The two-grid cycle comprises the simplest case with one restriction and one prolongation, and is mainly used for demonstration and testing purposes. The V-cycle arises from recursively applying the two-grid cycle to the solution process of the coarse grid problem. The W-cycle implements a second recursive call after the first call, resulting in a more accurate coarse grid correction. Like in the V-cycle, for the F-cycle the problem is first restricted to the coarse grid step by step, but additional V-cycles are applied on each level when prolongating back towards the fine grid. We note that all of these schemes except for the two-grid cycle result in a computational complexity of $\mathcal{O}(k)$.

To solve the problem on the coarse grid, once again a conjugate gradient method is applied. This method requires the application of matrix-vector and matrix-matrix products, which poses a challenge in parallel computations, especially since the problem on the coarse grid is typically distributed over a lot more processors than one would usually use for this relatively small-sized problem. This makes the coarse grid solver especially communication-heavy. However, as *Gmeiner et al.* (2014a) have shown, the largest amount of time is spent with the fine grid smoother, so the efficiency of the coarse grid solver does not influence the total efficiency very much.

A more detailed description of the multigrid method can be found e.g. in *Briggs et al.* (2000), *Gmeiner* (2013) and *Gmeiner et al.* (2015), on which this overview is based.

3.6 Discretizing the Heat Equation

In the truncated anelastic liquid approximation (TALA), the heat equation (2.56) reads

$$-c_p\rho\,\underline{v}\cdot\mathrm{grad}\,(T)+\alpha T\,\underline{v}\cdot\mathrm{grad}\,(p)+\mathrm{div}\,(k\,\mathrm{grad}\,(T))+H+\sum_{i,j}\underline{\underline{\sigma}}_{ij}\,\dot{\varepsilon}_{ij}=c_p\rho\,\partial_t T. \quad \text{(2.56 revisited)}$$

As mentioned before, we can assume that velocity and pressure are given from the solution of the Stokes system. The terms that account for volumetric compression (second), internal heating (fourth) and shear heating (fifth) can simply be accounted for by adding a constant value per time step and discrete degree of freedom, so we re-write

$$\partial_t T=-\underline{v}\cdot\mathrm{grad}\,(T)+\frac{1}{c_p\rho}\mathrm{div}\,(k\,\mathrm{grad}\,(T))+\mathrm{const.} \tag{3.44}$$

In case we use a reference state, we need to solve equation (2.60),

$$-c_p\rho\,\underline{v}\cdot\mathrm{grad}\,(T)+\alpha T\,\underline{v}\cdot\mathrm{grad}\,(p_{\mathrm{ref}})+\mathrm{div}\,(k\,\mathrm{grad}\,(T))+H+\sum_{i,j}\underline{\underline{\sigma}}_{ij}\,\dot{\varepsilon}_{ij}=c_p\rho\,\partial_t T_{\mathrm{dev}}. \quad \text{(2.60 revisited)}$$

Splitting the temperature T into a reference temperature T_{ref} and a deviatoric temperature T_{dev}, we can then also absorb the terms that contain the gradient of the reference temperature into the constant and get

$$\partial_t T_{\mathrm{dev}}=-\underline{v}\cdot\mathrm{grad}\,(T_{\mathrm{dev}})+\frac{1}{c_p\rho}\mathrm{div}\,(k\,\mathrm{grad}\,(T_{\mathrm{dev}}))+\mathrm{const}, \tag{3.45}$$

in complete analogy to the reference state-free case (3.44). For the Boussinesq approximation (2.67) we obtain

$$\partial_t T_{\mathrm{dev}}=-\underline{v}\cdot\mathrm{grad}\,(T_{\mathrm{dev}})+\frac{1}{c_p\rho_0}\mathrm{div}\,(k\,\mathrm{grad}\,(T_{\mathrm{dev}}))+\mathrm{const}, \tag{3.46}$$

where the constant contains internal heating and shear heating. For the further simplified case (2.72) we finally have

$$\partial_t T_{\mathrm{dev}}=-\underline{v}\cdot\mathrm{grad}\,(T_{\mathrm{dev}})+\frac{k}{c_p\rho_0}\Delta T_{\mathrm{dev}}. \quad \text{(2.72 revisited)}$$

The conceptional similarity of equations (3.44), (3.45), (3.46) and (2.72) is apparent.

3.6.1 Time stepping

The two terms in the above equations that cannot be accounted for by simple addition of a number are the advective term ($\underline{v}\cdot\mathrm{grad}\,(T)$) and the diffusive term ($\mathrm{div}\,(k\,\mathrm{grad}\,(T))$). The advective term is a local quantity that can be computed from any spatial discretization scheme every time we need to calculate the right hand side of the heat equation. As we have noted in section 2.1, the Peclet number tells us that the advective heat transport rate is larger than its conductive counterpart by four orders of magnitude. We can thus also evaluate the diffusive term locally.

In all cases, the part of the heat equations that is relevant for the temporal discretization can be written as

$$\partial_t T = R(T(x,t), x, t). \tag{3.47}$$

We discretize this equation in time, and introduce the notation T^χ for the temperature field at time step χ. The time between two time steps χ and $\chi + 1$ is noted as Δt. We allow for time steps of different duration, but leave out the index for the Δt, keeping in mind its potential dependency on the time step number χ.

In this notation, the forward Euler scheme can be noted as

$$T^{\chi+1} \approx T^\chi + \Delta t\, R(T^\chi), \tag{3.48}$$

with a local error of $\mathcal{O}(\Delta t^2)$ and a global error of $\mathcal{O}(\Delta t)$.

For the explicit midpoint method, we subdivide the time step between χ and $\chi + 1$ into two equally long time steps between χ, $\chi + 1/2$ and $\chi + 1$, and get the second order scheme

$$T^{\chi+1} \approx T^\chi + \Delta t\, R(T^{\chi+1/2}), \tag{3.49}$$

$$T^{\chi+1/2} \approx T^\chi + \frac{1}{2}\Delta t\, R(T^\chi), \tag{3.50}$$

with a local error of $\mathcal{O}(\Delta t^3)$ and a global error of $\mathcal{O}(\Delta t^2)$. Note that this scheme requires two evaluations of the right hand side R and thus of the Stokes system.

In regions of phase transitions, where densities become strongly dependent on depth and temperature, these explicit methods become unstable and oscillations can be observed. To prevent this, we introduce the implicit Euler scheme

$$T^{\chi+1} \approx T^\chi + \Delta t\, R(T^{\chi+1}), \tag{3.51}$$

with a local error of $\mathcal{O}(\Delta t^2)$ and a global error of $\mathcal{O}(\Delta t)$. Since we are interested in $T^{\chi+1}$, this requires the solution of a nonlinear system of equations. A standard choice for this would be Newton's method, but since we do not know the derivatives of R with respect to T, we cannot apply it here. Instead, we use the secant method, which can be considered a finite difference approximation to Newton's method. In this iteration process, the temperature at the current time step T^χ is given, while we have to iteratively find $T^{\chi+1}$, for which we will use an upper left index to denote the iteration number. Starting with initial temperatures ${}^0T^{\chi+1} = (T^\chi + \text{const})$ and ${}^1T^{\chi+1} = (T^\chi - \text{const})$ we can iterate

$${}^{\psi+1}T^{\chi+1} = {}^{\psi}T^{\chi+1} - \frac{{}^{\psi}T^{\chi+1} - {}^{\psi-1}T^{\chi+1}}{{}^{\psi}T^{\chi+1} - {}^{\psi-1}T^{\chi+1} + R({}^{\psi}T^{\chi+1}) - R({}^{\psi-1}T^{\chi+1})} \tag{3.52}$$

Since $R(T^{\chi+1})$ depends on the velocity at time $\chi + 1$, in principle one would have to introduce another iteration loop around the secant iterations to solve the Stokes system for the new buoyancy field ${}^{\psi_{\max}}T^{\chi+1}$. In practice we simply use the velocity at time χ. Numerical experiments seem to confirm that this inaccuracy is small enough for practical purposes.

These time stepping schemes are standard methods and well known from the numerics of ordinary differential equations. A more detailed description of these methods can be found in any standard introductory book, e.g. in *Butcher* (2008).

4 Flatland

Flatland is a two dimensional finite element code developed in the course of this work. It is written in Fortran, featuring many of the modern software concepts that have been introduced to the language since the release of the 1990 standard (*ISO1539*, 1991). It parallelized with OpenMP and can thus run in parallel on shared memory architectures. It works with the Q1-P0 and the Q1-Q1 element on the annulus (see section 3.4) and includes three different solvers (section 3.5): two direct solvers (LAPACK, *netlib.org* (2015) and PARDISO, *Kuzmin et al.* (2013)) as well as an iterative solver comprised of a CG pressure correction scheme and a multigrid solver. The multigrid solver uses a linear interpolation for the prolongation and the corresponding transpose for the restriction, together with either a Jacobi or a Gauss-Seidel smoother. For the time stepping, the explicit and implicit Euler scheme as well as the explicit midpoint method (section 3.6) are supported.

Flatland supports the compressible (TALA) and the incompressible (Boussinesq) approximation (section 2.4.1 and 2.4.2), and can solve the static and the transient case (section 2.4.4). Homogeneous and inhomogeneous Dirichlet boundary conditions are implemented (section 2.5). For testing, solution of the pure Poisson problem is included as well.

Flatland features set of predefined test cases to ensure the basic functionality. We will now introduce some scenarios that compare the solution of the finite element system against known analytical solutions (section 4.1), and some to test the quality of the advection scheme (section 4.2). Other tests are performed in addition to ensure basic functionality, such as convergence checks, basic tests on the individual operators and tests of the Poisson problem.

4.1 Tests of the Stokes Equations with Manufactured Solutions

To test the quality of a numerical solution, it is desirable to compare it to analytical solutions. While this is not possible in general, we can find special solutions to the Stokes system in the Boussinesq approximation (2.65) and (2.66). We will thus introduce the biharmonic equations as given i.e. in *Davies* (1999).

4.1.1 Biharmonic Equation

To produce an analytic solution, we start with the constant viscosity equations (2.70) and (2.71),

$$-\eta\Delta\underline{v} + \operatorname{grad}(p_{\text{dev}}) = \rho_0\alpha\underline{g}T \qquad \text{(2.70 revisited)}$$

$$\operatorname{div}(\underline{v}) = 0 \qquad \text{(2.71 revisited)}$$

on a spherical shell with $r_{\text{lower}} = 1$ and $r_{\text{upper}} = 2$ and no slip conditions on all boundaries. We replace the right hand side in (2.70) with the arbitrary right hand side vector $\underline{f}$,

$$-\eta\Delta\underline{v} + \operatorname{grad}(p_{\text{dev}}) = \underline{f} \tag{4.1}$$

$$\operatorname{div}(\underline{v}) = 0. \tag{2.71 revisited}$$

In order to find an analytical solution, we now interpret (4.1) and (2.71) as a three dimensional problem with an all-zero third component. We can then define the stream function $\underline{\Psi}$ as

$$\underline{v} = \operatorname{curl}(\underline{\Psi}). \tag{4.2}$$

A stream function that always gives us a zero z-component in the velocity is

$$\underline{\Psi} = (0, 0, \Psi_z). \tag{4.3}$$

This definition leaves only one degree of freedom in the stream function and thus also in the velocity. However, we can show that this reduction is directly linked to automatically fulfilling the continuity equation (2.71): plugging (4.2) into the left hand side of the continuity equation (2.71) we get

$$\operatorname{div}(\underline{v}) = \operatorname{div}(\operatorname{curl}(\underline{\Psi})) = 0, \tag{4.4}$$

because the divergence of the curl of any vector field is always zero. The continuity equation is thus fulfilled.

We now define the vorticity

$$\underline{\lambda} = \operatorname{curl}(\underline{v}) = \operatorname{curl}(\operatorname{curl}(\underline{\Psi})) \tag{4.5}$$

and note that as the stream function $\underline{\Psi}$ it is of the form $\underline{\lambda} = (0, 0, \lambda_z)$. In cylindrical coordinates, this enables us to re-write the only non-zero component of (4.5) as

$$\lambda_z = \Delta_p \Psi_z, \tag{4.6}$$

where Δ_p is the polar Laplacian, given as $\Delta_p \circ = \frac{1}{r}\partial_r(r\partial_r \circ) + \frac{1}{r^2}\partial_\varphi^2 \circ$.

Taking the curl of $\underline{\lambda}$ yields

$$\begin{aligned}\operatorname{curl}(\underline{\lambda}) &= \operatorname{curl}(\operatorname{curl}(\underline{v})) \\ &= \operatorname{grad}(\operatorname{div}(\underline{v})) - \Delta\underline{v} \\ &= -\Delta\underline{v},\end{aligned} \tag{4.7}$$

and inserting into the momentum equation (4.1)

$$-\eta\operatorname{curl}(\underline{\lambda}) + \operatorname{grad}(p_{\text{dev}}) = \underline{f}. \tag{4.8}$$

Taking the curl on this equation the pressure p_{dev} drops out, because the curl of any gradient field is zero, and we get

$$-\eta\operatorname{curl}(\operatorname{curl}(\underline{\lambda})) = \operatorname{curl}\left(\underline{f}\right). \tag{4.9}$$

Once again we exploit that the gradient of the divergence is zero, so we can re-write

$$-\eta\,\Delta\underline{\lambda} = \operatorname{curl}\left(\underline{f}\right), \tag{4.10}$$

and using (4.6), the z-component becomes

$$-\eta\,\Delta_p^2\Psi_z = \frac{1}{r}\partial_r(r f_\varphi) - \frac{1}{r}\partial_\varphi f_r. \tag{4.11}$$

We call this the bi-harmonic equation. It is a transformation of the constant viscosity Boussinesq equations (4.1) and (2.71) into a single fourth order partial differential equation.

4.1.2 Application of the Biharmonic Equation as a Benchmark

Rather than actually solving the Biharmonic equation (4.11), we choose some Ψ_z and calculate the right hand side $\underline{f}$ from it. Let

$$\Psi_z = \frac{1}{2}(r-1)^2(r-2)^2\sin(2\varphi). \tag{4.12}$$

Using (4.2) this gives for the velocity

$$\begin{aligned} v_r &= \frac{1}{r}(r-1)^2(r-2)^2\cos(2\varphi) \\ v_\varphi &= -(r-1)(r-2)(2r-3)\sin(2\varphi), \end{aligned} \tag{4.13}$$

which fulfills the boundary conditions $v(r=1) = v(r=2) = 0$ and $v(\varphi=0) = v(\varphi=2\pi)$. Also the continuity equation (2.71) is fulfilled, since $\operatorname{div}(\underline{v}) = 0$.

Applying the polar Laplacian Δ_p twice gives

$$\Delta_p^2\Psi_z = \frac{9(5r^2-6)}{r^3}\sin(2\varphi), \tag{4.14}$$

which we can then insert into (4.11):

$$-\eta\frac{9(5r^2-6)}{r^3}\sin(2\varphi) - \frac{1}{r}\partial_r(rf_\varphi) + \frac{1}{r}\partial_\varphi f_r = 0. \tag{4.15}$$

The equation is underdetermined, but since we are only interested in one particular solution we can choose $f_\varphi = 0$ and pick

$$f_r = -\eta\frac{9(5r^2-6)}{2r^2}\cos(2\varphi). \tag{4.16}$$

We have now obtained a right hand side that we can plug into Flatland to check whether we obtain a velocity field that satisfies (4.13). Figure 4.1 visualizes the buoyancy term from this test as well as the analytical and an example of a numerical solutions.

4.1.3 Error Analysis

In figure 4.2 we compare the L_2-error between the analytical solution (as given in equation 4.16) and several numerical solutions obtained in Flatland. We test six combinations of discretization and solver: Q1-P0 and Q1-Q1, together with the three solvers described in section 3.5), the two direct solvers LAPACK and PARDISO and an iterative solver. The iterative solver uses the pressure correction conjugate gradient method with a maximum of 30 iterations, or until the residual in the L_2-norm changes by less than one percent. The multigrid scheme applies three W-cycles with five Gauss-Seidel pre- and post-smoothing steps. On the coarse grid we only have one layer with nodes of unknowns, consisting of eight nodes tangentially. This coarse grid problem is solved by 25 Gauss-Seidel smoothing steps. We run a set of tests with increasing resolution for each of the six combinations of discretization and solver. To guarantee a reasonable aspect ratio of the cells, we set the number of radial layers to roughly one eights the number of nodes in tangential direction. In figure 4.2 we plot the error against the radial resolution at an assumed mantle depth of one.

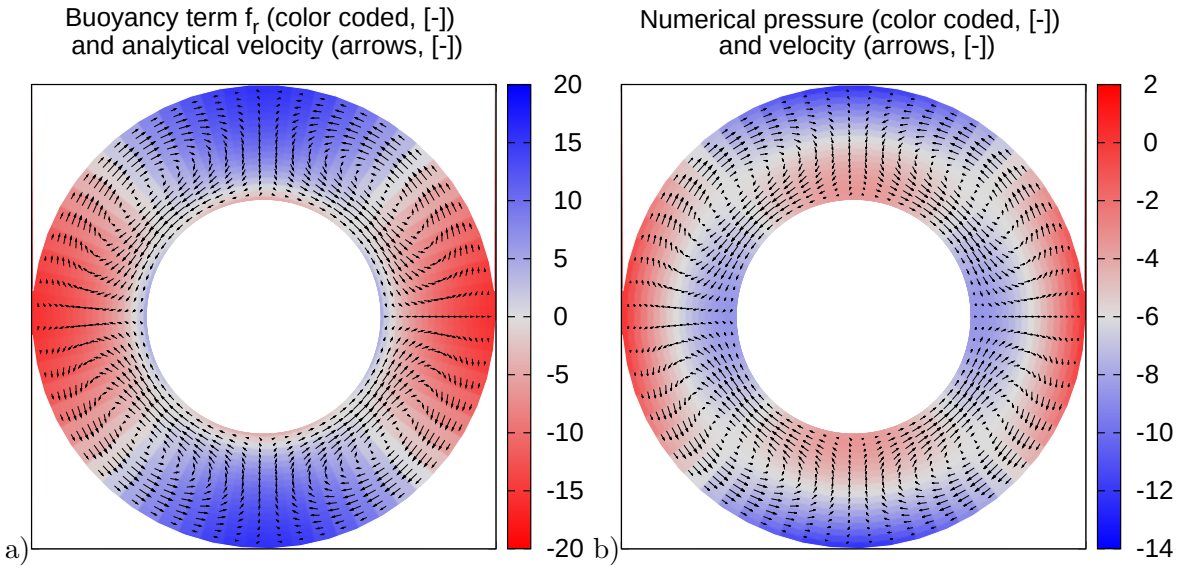

Figure 4.1: Comparison of analytical and numerical solutions of the constant viscosity Stokes problem. (a) Analytically obtained velocity and buoyancy term f_r. (b) Numerically obtained pressure and velocity for $n_r = 19$ and $n_t = 63$, obtained with the Q1-P0 discretization and the LAPACK solver.

The plot shows an almost perfect quadratic convergence for the direct solvers up to the highest resolved case with $\sim 27\,000$ degrees of freedom for the LAPACK solver and $\sim 6.3 \cdot 10^6$ degrees of freedom for the PARDISO solver, respectively. This is a good indication that the error is dominated by the discretization error, which arises because of the difference between discrete and continuous solution. In the iterative scheme, we see good convergence up to $\Delta x = 7.69 \cdot 10^{-3}$ (corresponding to 131 radial layers) for the Q1-P0 discretization, and some deviations from the ideal convergence for Q1-Q1 in this range. For higher resolutions, the solution does not improve significantly, for which the standard explanation would be round-off errors. Since at $\sim 10^{-4}$ the error is still relatively high for double precision calculations, we cannot rule out errors in the details of the implementation. However, since a re-write of the iterative solver is due in any case, no further investigations have been made.

4.2 Tests of the Heat Equation

4.2.1 Tangential Advection

To test the quality of the solution of the advection equation, we set up a velocity field with constant angular velocity and no radial component, corresponding to a solid body rotation. We then advect an initial temperature perturbation once around the annulus for six different cases, as given in table 4.1. The setup and result for the Boussinesq case with an explicit Euler time stepping are shown in figure 4.3. Due to numerical diffusion, the final temperature distribution is smeared out tangentially, such that between 80 and 90 percent of the energy perturbation are diffused away. The amount depends mainly on the time stepping scheme, the approximation (Boussinesq or TALA) does not play a significant role here. The tangentially diffused amount of energy is negligible, which is to be expected, since numerical diffusion should only occur in the direction of advection.

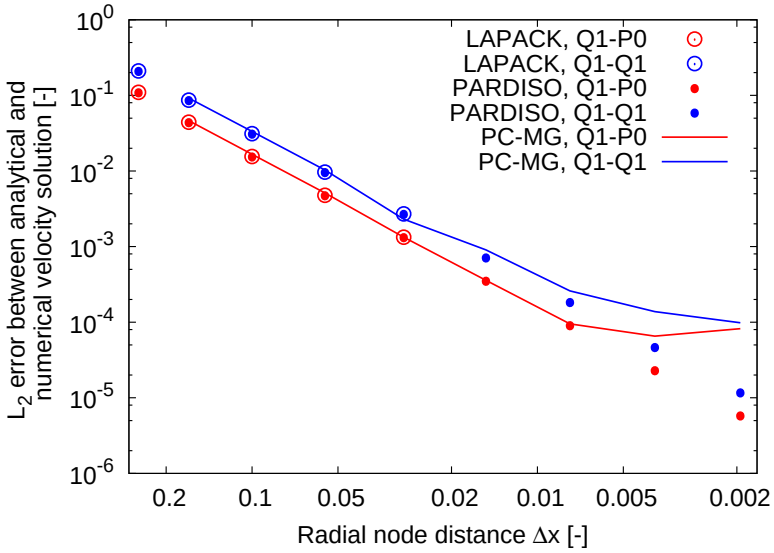

Figure 4.2: L_2 error norm between the analytical solution of the velocity (as given in equation 4.16) and different numerical solutions obtained in Flatland. Shown are solutions for two different discretizations (red: Q1-P0, blue: Q1-Q1) and three different solvers (as described in section 3.5): LAPACK (circles), PARDISO (dots) and an iterative solver based on the pressure correction scheme and a multigrid algorithm (PC-MG, lines).

4.2.2 Radial Advection

To test the quality of the reference state implementation, we advect an initial temperature perturbation outwards and back in (see figure 4.4). While the amount of energy of this perturbation depends on its depth, its energy should be restored after being moved back to its original location. The results are given in table 4.2. The total energy defect, i.e. the difference in total energy in the system before and after the advection is on the order of 10^{-4}, while the energy that is found outside the original angle range is on the order of 10^{-5}.

For reference, values for the Boussinesq case are also provided. Since the reference state is subtracted out here, it is not considered in the calculation of the total energy, leading to the difference in the values given in table 4.2.

4.2.3 TALA with and without reference state

Flatland provides two versions of the TALA, a traditional one in with the reference state is subtracted out of the state variables (equations 2.58–2.61), and another one, where the state variables represent the total quantities (equations 2.54–2.57). To check whether they both yield the same result, we run a test scenario with both versions, advecting and diffusing an initial temperature distribution for 20 time steps. The initial setup is plotted in figure 4.5 a, the final distribution for the two cases is shown in figures 4.5 c and 4.5 d. Figure 4.5 b shows the reference state that is used in the scenario with reference state. At the same time, this reference state is used for initialization of the reference state-free scenario and subtracted from the final state for the evaluation and plotting. For the test case, we use the Q1-P0 discretization and the LAPACK solver, together with an explicit Euler time stepping. The relative root mean square temperature error between the two final states is $2.27 \cdot 10^{-4}$.

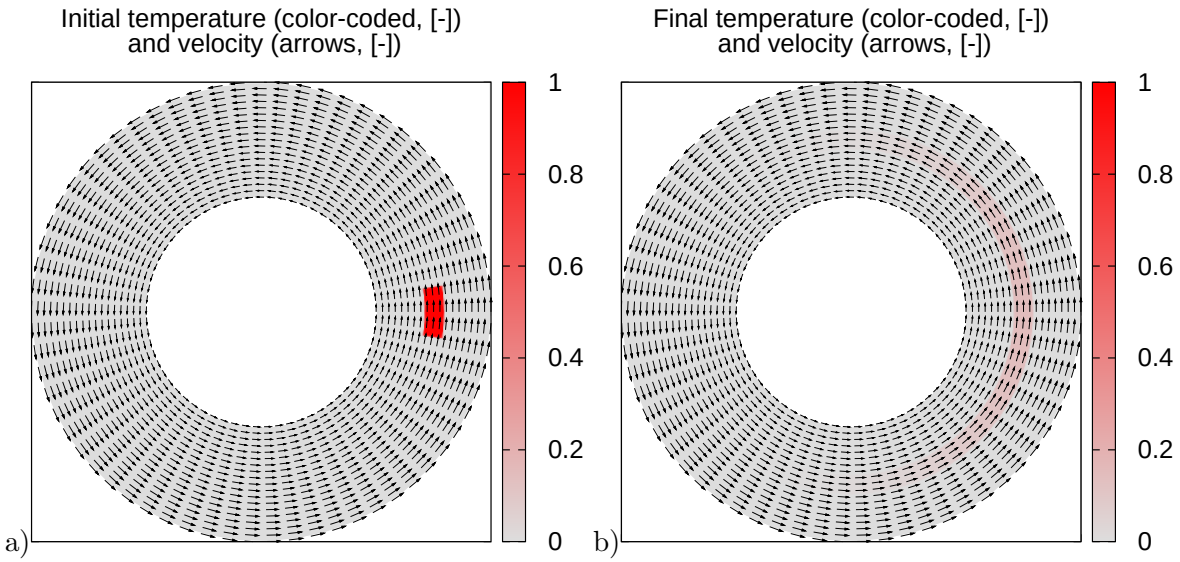

Figure 4.3: Advection of a temperature anomaly once around the annulus for the Boussinesq case with an explicit Euler time stepping scheme. Shown are the initial (a) and final (b) temperature distribution.

Scenario	*Time stepping*	*Radially diffused energy [percent]*	*Tangentially diffused energy [percent]*	*Tangential center of energy [degrees]*
Boussinesq	forward Euler	$3.50 \cdot 10^{-13}$	80.4	$-7.96 \cdot 10^{-6}$
TALA with ref. state	forward Euler	$1.25 \cdot 10^{-13}$	81.0	$1.58 \cdot 10^{-4}$
Boussinesq	explicit midpoint	0	85.3	-0.0221
TALA with ref. state	explicit midpoint	0	86.0	-0.0211
Boussinesq	implicit Euler	$3.52 \cdot 10^{-13}$	88.0	-0.193
TALA with ref. state	implicit Euler	$1.27 \cdot 10^{-13}$	88.6	-0.193

Table 4.1: Numerical errors from the advection of a temperature anomaly outwards and back inwards for different approximations and time stepping schemes. All values would be zero for an ideal solver.

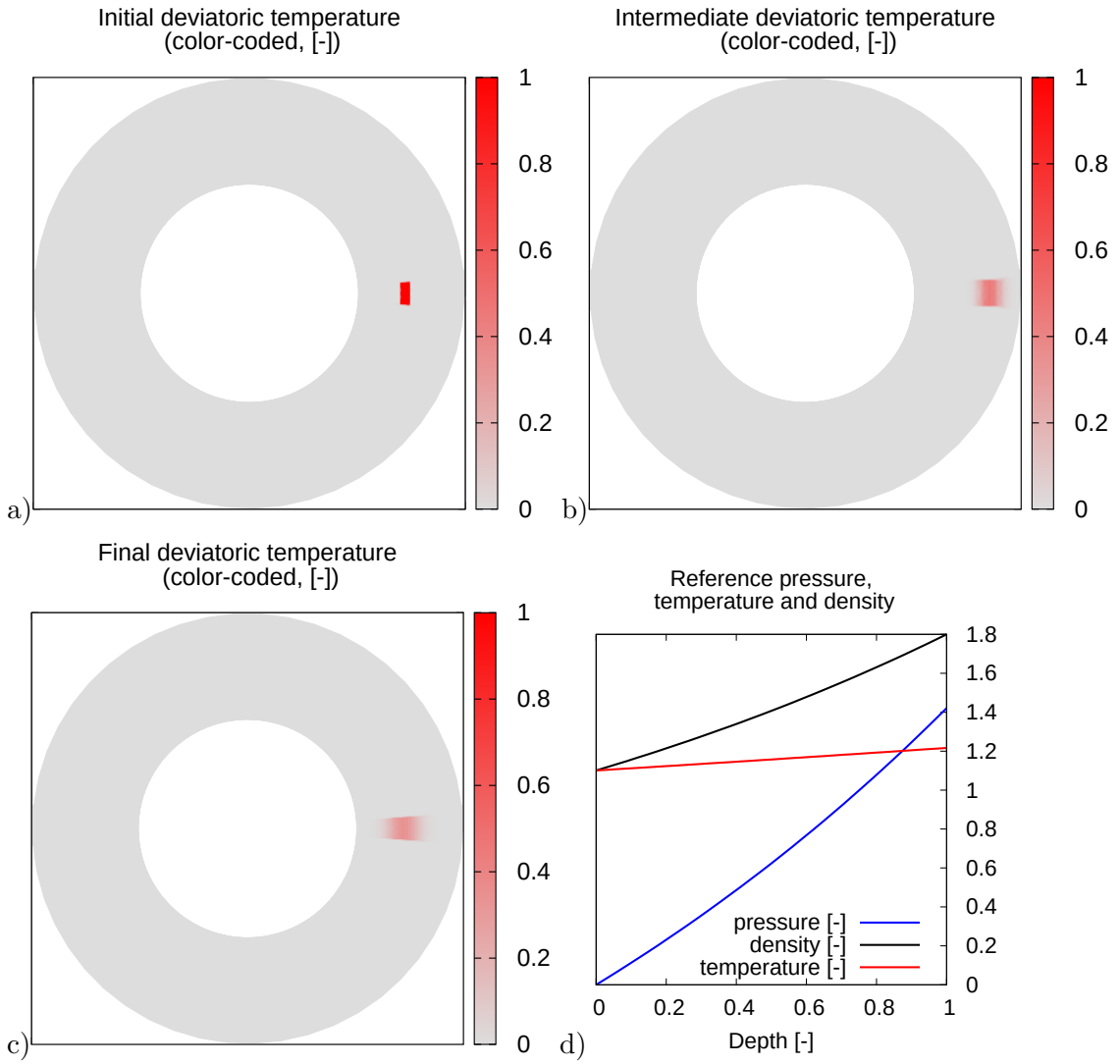

Figure 4.4: Advection of a temperature anomaly outwards and back for the TALA case with reference state and an explicit Euler time stepping scheme. Shown are the initial temperature distribution (a), the intermediate step at which the velocity is switched from outwards to inwards (b), and the final temperature distribution (c). Panel (d) shows the reference state.

Scenario	*Total energy initially in [-] the system [-]*	*Total energy defect [-]*	*Relative energy defect [percent]*	*Energy outside original angle range [percent]*
Boussinesq	$1.86 \cdot 10^{10}$	$3.31 \cdot 10^{7}$	0.178	$1.67 \cdot 10^{-15}$
TALA with ref. state	$7.23 \cdot 10^{12}$	$1.65 \cdot 10^{9}$	0.0228	0.00381

Table 4.2: Numerical errors from the advection of a temperature anomaly outwards and back inwards for different approximations and an explicit Euler time stepping scheme. All values would be zero for an ideal solver.

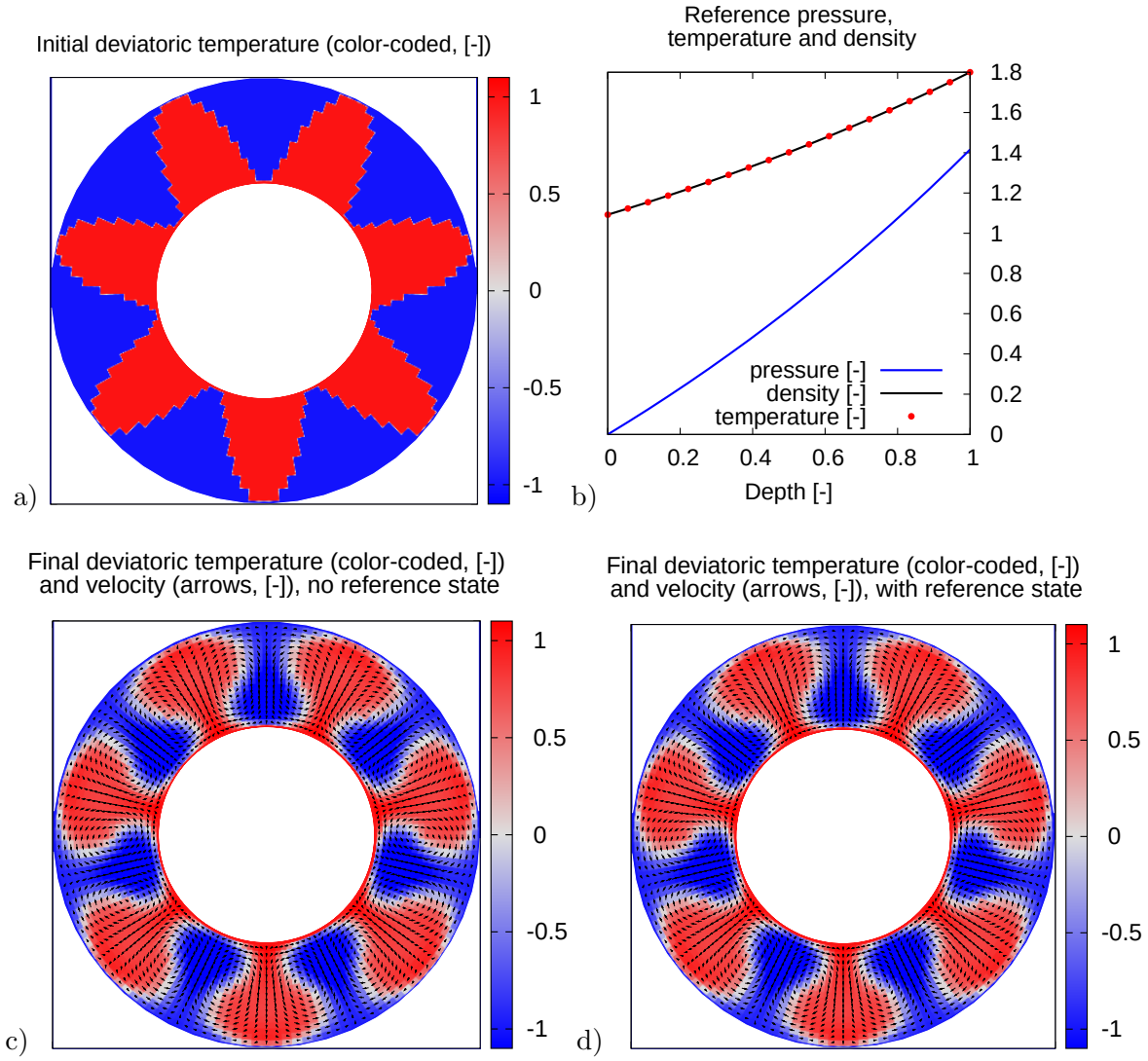

Figure 4.5: Comparison of a convection simulation in the TALA approximation with and without reference state with a first order Euler time stepping scheme. Panel (a) shows the initial temperature distribution, (c) and (d) the final state after 20 time steps in the formulations with and without a reference state. Panel (b) shows the reference state, which is also used to calculate the deviatoric temperature in the reference state-free scenario.

5 Mantle Convection Simulations in Two Dimensions

5.1 Introduction

The patterns that arise in mantle convection models are dominated by many factors, out of which we would like to mention here: rheology, initial and boundary conditions, heating mode and mineralogy (including phase changes and compressibility).

From stability analysis we know that the main parameter that steers mantle convection is the Rayleigh number (*Turcotte and Oxburgh*, 1967; *Houseman et al.*, 1981; *Davies*, 1999; *Ribe*, 2007), and many numerical simulations confirm this finding (*Jarvis and McKenzie*, 1980; *Baumgardner*, 1985; *Gurnis and Davies*, 1986; *Dubuffet et al.*, 2000). Convection starts at about $\mathrm{Ra}_{\mathrm{b}} = 1000$, and becomes more vigorous with higher Rayleigh numbers. Once they get sufficiently high, the influence of the initial conditions diminishes after a few times the convective timescale (*Ita and King*, 1994). The boundary conditions on the other hand influence the convective patterns at all times. Three types of boundaries are relevant for mantle convection: homogeneous Dirichlet conditions (no-slip), stress-free (free-slip) and inhomogeneous Dirichlet conditions (plate-like) (*Solomatov*, 1995; *Schubert et al.*, 2001). While no-slip and free-slip conditions at both boundaries imply purely buoyancy-driven flow, plate-like boundaries can provide an additional driving component.

Another very important factor for the convective planform is viscosity stratification. *Bunge et al.* (1996) have studied the effect of layered viscosities and found that high stratification shifts the planform towards low degree, long wavelength patterns. They also concluded that phase changes further increase the dominating convective wavelength. *Bunge et al.* (1997) then established that viscosity stratification is the dominating factor for the long wavelength patterns above phase transitions, which have less influence on the planform. *Tackley* (1996) found that also lateral viscosity variations are of less importance compared to viscosity layering.

The role of phase transitions in the mantle has also been explored by *Schubert et al.* (1975), who determined from boundary layer theory that material can penetrate the transition zone convectively, and that mineralogical effects in the transition zone can enhance mantle convection. Later, slabs have been found to pond near the 660 km phase transition, warming conductively before they continue downward into the lower mantle. The ponding slabs can also be seen in seismic images of the upper mantle. *Fukao et al.* (2009) give a review of the relevant studies.

The effects of hydrostatically compressible materials in three dimensional spherical models of the mantle have first been studied by *Baumgardner and Frederickson* (1985); *Glatzmaier* (1988), and are today routinely described by the TALA approximation (see section 2.4.1).

Steinbach et al. (1989) have found that compressibility induces an asymmetry between the structures in the upper and the lower boundary layer. *Zhang and Yuen* (1996) pointed out that compressibility mainly influences small-scale structures, while its effects on the global patterns are relatively small.

Concerning the heating mode, early estimates of the mantle's heat budget assume roughly ten percent bottom heating compared to internal radiogenic heat production (*Davies and Richards*, 1992). More recent studies correct this estimate to about 30 percent (*Lay et al.*, 2008). *Bunge et al.* (1997) confirmed that bottom heating is responsible for a lower boundary layer, where it adds modest thermal heterogeneity compared to purely internal heating.

Another topic that is strongly debated in the geodynamic community is that of the nature of large seismically slow anomalies in the lowermost mantle. It has been proposed that these large low shear velocity provinces consist of chemically distinct material (*Trampert et al.*, 2004), and it has been suggested that plumes originate from the borders of these so-called chemical piles (*Burke et al.*, 2008; *Steinberger and Torsvik*, 2012). Statistical analysis however suggests that this assumption may not be robust (*Davies et al.*, 2012; *Austermann et al.*, 2014), so the nature of these provinces stays debated.

In this chapter we will examine the effects of a depth-dependent viscosity in combination with compressibility and the heating mode, and then move on to examining the effect of realistic mineralogical parameters. We use Flatland with the Q1-P0 discretization, the PARDISO solver for the constant viscosity case, and the pressure correction/multigrid scheme for variable viscosity. For integration in time we use the second order explicit midpoint method scheme with adaptive time steps according to the CFL criterion in the simulations without phase transitions (section 5.2), and an implicit Euler time stepping for simulations that incorporate latent heat effects (section 5.3).

5.2 Planform of Mantle Convection

5.2.1 Boussinesq, Bottom Heated

We set up an incompressible, bottom-heated model with no-slip boundary conditions on an annulus with $\mathrm{Ra}_\mathrm{b} = 10^7$, which corresponds to the low end range of assumed Rayleigh numbers within the Earth. We run one simulation with a constant viscosity of $10^{22}\,\mathrm{Pa\,s}$, and another with a two-layered viscosity profile. This profile has an upper/lower mantle transition at 660 km, with a viscosity of $10^{22}\,\mathrm{Pa\,s}$ in the lower mantle and $10^{20}\,\mathrm{Pa\,s}$ in the upper mantle.

Figure 5.1 shows the resulting convective patterns after an initialization phase of several overturns. For the constant viscosity case, we notice up- and downwellings of equal strength, and an average deviatoric temperature that corresponds to the average between the upper and lower boundary temperature. We also note that stress and strain rate are diffused evenly across the domain, owing to the elliptic nature of the Stokes equations.

For the depth-dependent viscosity, the downwellings become more numerous, while the upwellings are more focused along well-defined vertical channels. Also, due to the lower viscosity in the upper layer, heat is removed from the system more efficiently at the upper boundary than heat is entered at the CMB. This leads to an effective cooling of the system, and thus a sub-adiabatic average temperature throughout the whole mantle after equilibration has been achieved.

The strain rate across the upper/lower mantle interface changes by a factor of about two, leading to correspondingly larger velocities in the upper mantle. The stress on the other hand

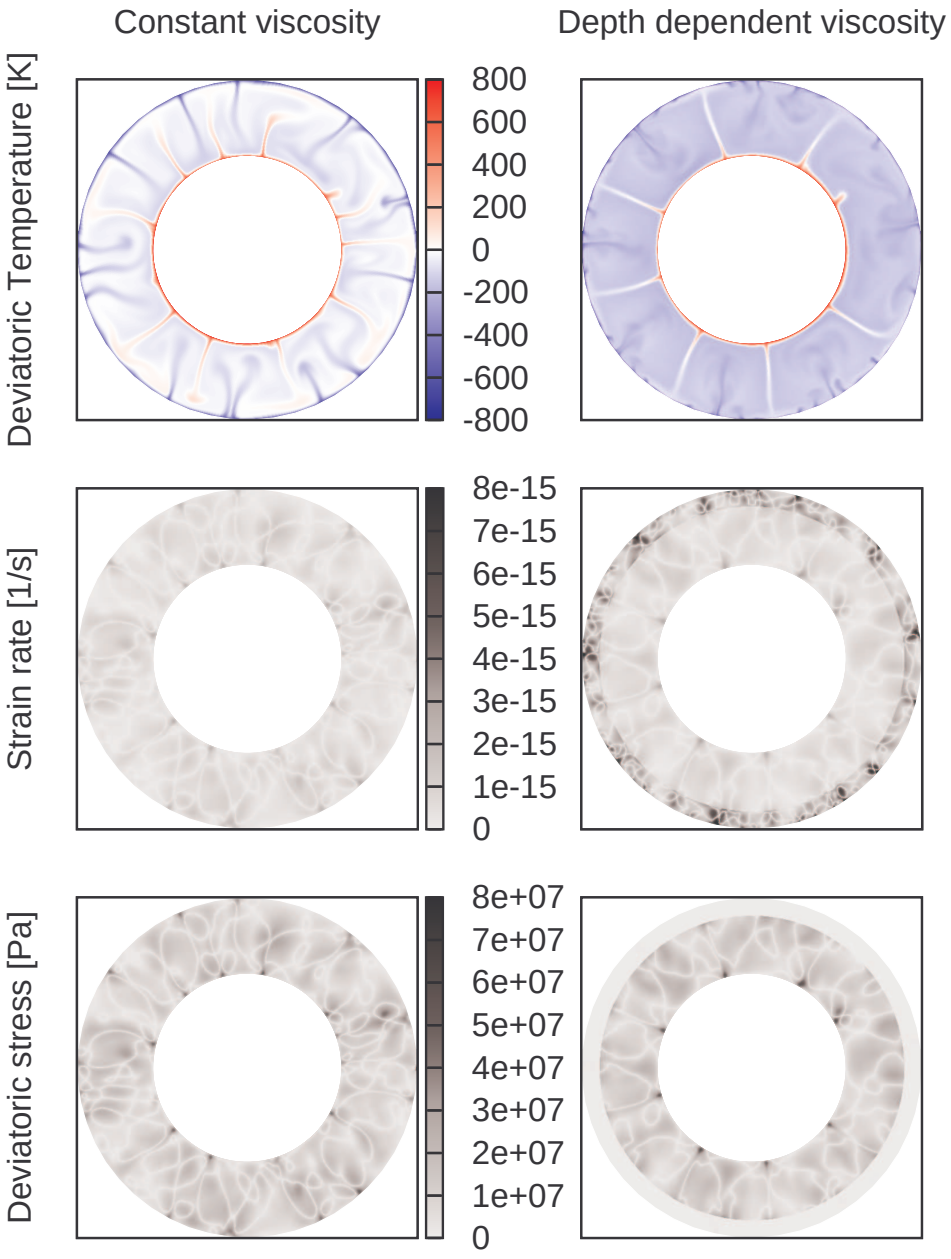

Figure 5.1: Convective patterns of two mantle convection simulations in the Boussinesq approximation and pure bottom heating several overturns after initialization. *Left:* Constant viscosity of 10^{22} Pa s, *right:* Layered viscosity model with 10^{22} Pa s below 660 km and 10^{20} Pa s above. *Top:* Deviatoric temperature, *center:* strain rate, *bottom:* deviatoric stress.

decreases by a factor of roughly 50.

5.2.2 Anelastic, Bottom Heated

To evaluate the effect of compressibility, we employ the same setup as before, but use the TALA approximation with a reference state as shown in figure 5.2. This represents a planet that is somewhat more compressible than the Earth. We neglect shear heating to strictly exclude all internal heating contributions.

The resulting convective patterns are shown in figure 5.3 (top). In contrast to the Boussinesq case, we observe an asymmetry between down- and upwellings, with amplified downwellings and weakened plumes. The impact of the viscosity on the convective planform is similar to the incompressible case. Since the Rayleigh number has been calculated using the surface density, we observe an overall slightly more vigorous convection than in the incompressible case, owing to the effectively higher density in the lower mantle. The low- viscosity zone in combination with compressibility leads to a stronger sub- adiabaticity in the lower than in the upper mantle, where cold material from the surface accumulates more effectively than in the Boussinesq case. In the constant viscosity case, the average temperature follows the adiabate.

5.2.3 Anelastic, Mixed Heating

Finally, we add internal heating to the previous setup, with a Rayleigh number for internal heating of $\mathrm{Ra_i} = 6 \cdot 10^8$, and bottom heating of $\mathrm{Ra_b} = 10^7$ as before. As shown in figure 5.3 (bottom), we observe a strongly super-adiabatic temperature for the constant viscosity case. The depth-dependent viscosity on the other hand allows for an effective cooling from the upper boundary and thus a sub-adiabatic average temperature very similar to the bottom-heated case. For both viscosity settings, the planform is similar to the bottom-heated scenario, with only slightly more vigorous convection patterns in the mixed-heating settings.

5.3 Mineralogy

Based on model parameters by *Stixrude and Lithgow-Bertelloni* (2011), *Chust et al.* (2015) have developed a model of a pyrolitic mantle, giving its physical properties in dependence of temperature and pressure. Using lookup-tables, this allows us to incorporate realistic values for density, entropy, thermal conductivity, bulk modulus, specific heat and thermal expansion coefficient into our 2d model of the mantle. Latent heat is incorporated into the specific heat and the thermal expansion coefficient, rendering them effective properties. Since we need to resolve the latent heat within our mantle convection model, we smooth all physical properties such that the latent heat is distributed over a pressure range which is represented by several radial grid nodes within the convection model. As an example, density, effective thermal expansion coefficient and effective heat capacity are shown in figures 5.4, 5.5 and 5.6.

We use the truncated anelastic liquid approximation (TALA) in the reference-state-free formulation, although we still calculate an adiabate in order to show deviatoric quantities. A resolution of $n_r = 100$ and $n_t = 601$ is chosen, and we use a constant viscosity of $10^{22}\,\mathrm{Pa\,s}$ in a purely bottom-heated mantle. We set up an adiabatic reference state with a reference temperature of 1600 K at the upper boundary, while we prescribe an actual temperature of 300 K at the surface, yielding a deviatoric temperature at the surface of $-1300\,\mathrm{K}$. The

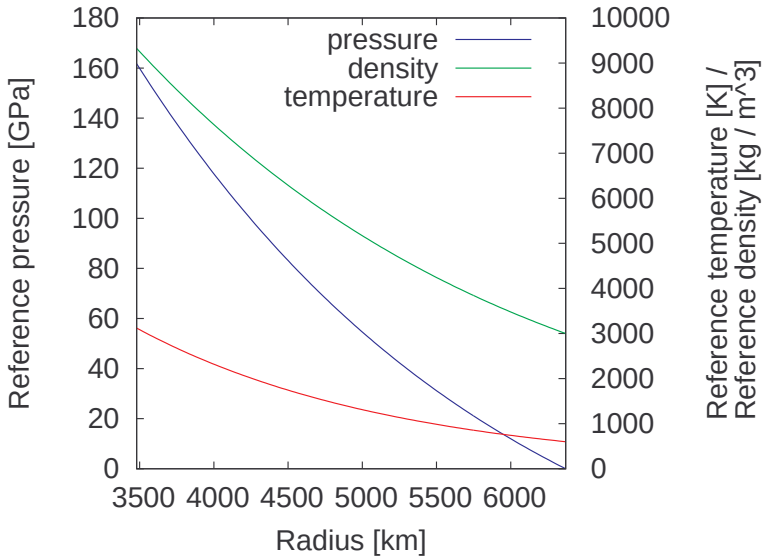

Figure 5.2: Reference state for the anelastic simulations over the radius within the planet. CMB is left, lithosphere right. *Left axis:* Reference pressure (blue curve), *right axis:* reference temperature (red curve) and reference density (green curve).

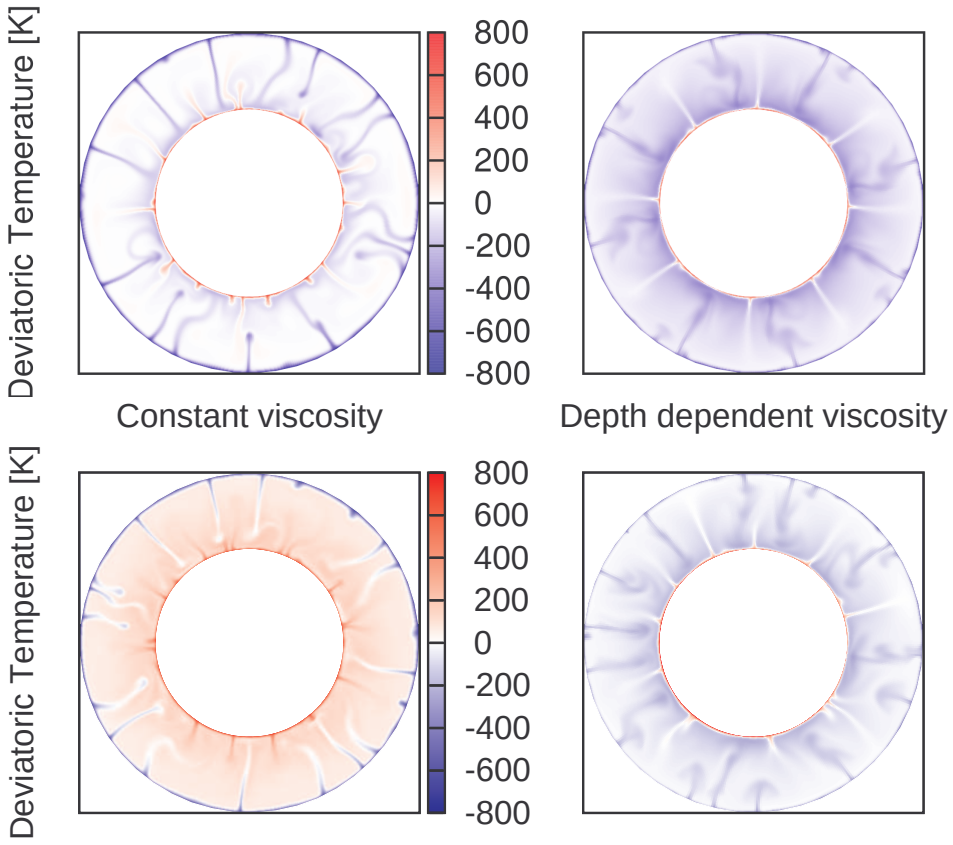

Figure 5.3: Convective patterns of two mantle convection simulations in the TALA approximation several overturns after initialization. *Top:* pure bottom heating, *bottom:* mixed heating. *Left:* Constant viscosity of 10^{22} Pa s, *right:* Layered viscosity model with 10^{22} Pa s below 660 km and 10^{20} Pa s above.

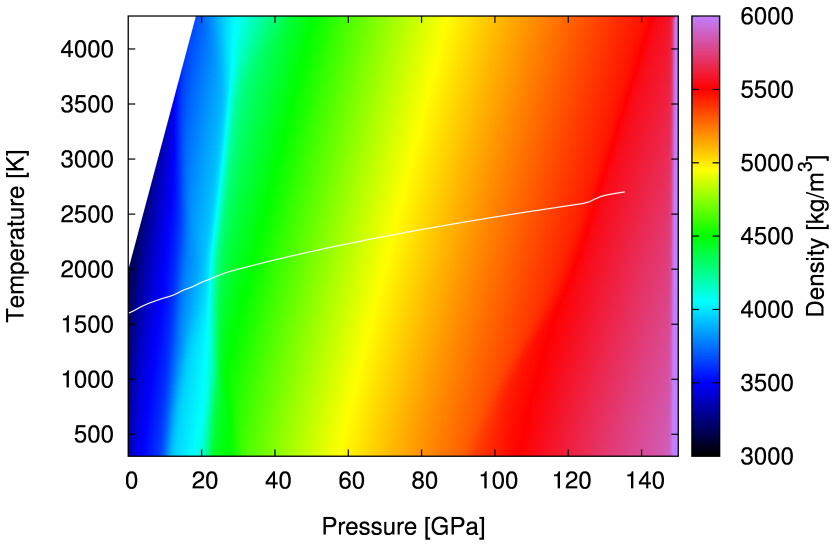

Figure 5.4: Smoothed phase diagram for a pyrolite. Melting occurs in the upper left, therefore no data is shown here. The white line represents the adiabatic reference state within the simulated mantle.

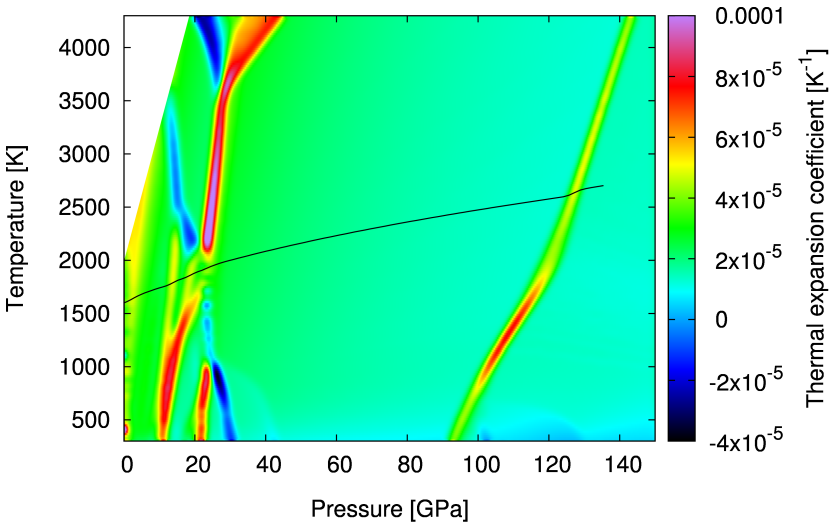

Figure 5.5: Smoothed effective thermal expansion coefficient for a pyrolite, including latent heat. Melting occurs in the upper left, therefore no data is shown here. The white line represents the adiabatic reference state within the simulated mantle.

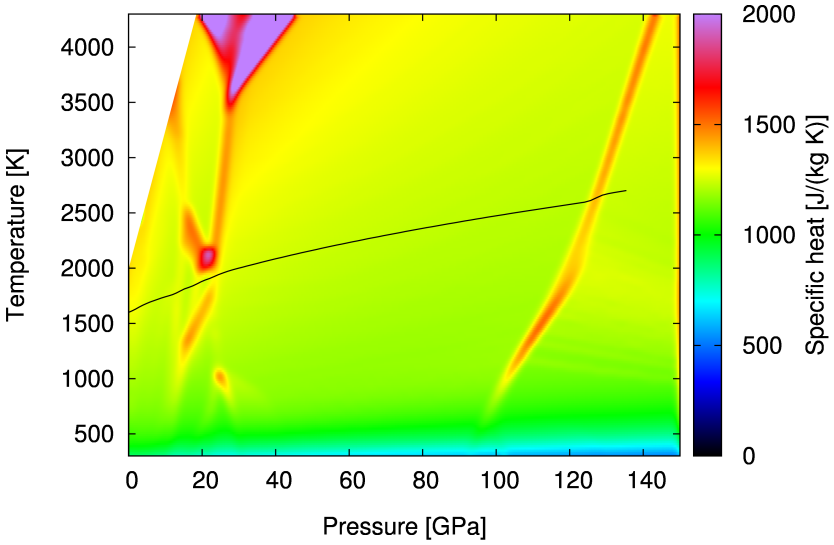

Figure 5.6: Smoothed effective heat capacity c_p for a pyrolite, including latent heat. Melting occurs in the upper left, therefore no data is shown here. The white line represents the adiabatic reference state within the simulated mantle.

resulting adiabate is shown in 5.7. The deviatoric temperature at the lower boundary is set to 1500 K, yielding a Rayleigh number of $\mathrm{Ra}_{\mathrm{b}} = 10^7$.

Figure 5.8 shows an example of a planform that arises after roughly three overturns with this setting. In contrast to all previous simulations with constant viscosity, the average temperature is hotter than the adiabate. The three major phase transitions at 410 km and 660 km depth and the perovskite/post-perovskite transition 200 km above the CMB are clearly visible in the density plot, since small perturbations in the depth of the transition zones show as strong density deviations from the discontinuous adiabatic background. Hot material accumulates below the perovskite/post-perovskite transition, offering a possible explanation for large low shear velocity provinces without requiring chemical heterogeneities. The effect of slab ponding near the 660 km transition can also be seen, although it is not very pronounced.

5.4 Discussion and Conclusions

In this chapter we have restricted our analysis to no-slip conditions. From stability analysis we know that free-slip conditions yield a slightly lower critical Rayleigh number than no-slip conditions. Analogously, it is often assumed that for higher Rayleigh numbers, the behavior of both setups is similar when slightly higher Rayleigh numbers are chosen in the no-slip case. For a more detailed overview on the topic see e.g. *Ribe* (2007); *Ricard* (2007).

The effect of viscosity stratification described by *Bunge et al.* (1996, 1997); *Tackley* (1996) can clearly be seen in our simulations, the lateral coherence length of the convection is consistently larger in the variable viscosity cases. In addition, we find that the low-viscosity layer in the upper mantle leads to an effective cooling of the mantle. This is a result of the low viscosity near the upper boundary, which increases the effective Rayleigh number and thus removes heat from the boundary more efficiently. A brittle lithosphere will most likely change this behavior, since it serves as a thermal insulator and will greatly modify the properties of the boundary layer. We thus conclude that for realistic, transient models of the Earth and other planets, the inclusion of a realistic upper boundary is indispensable *Höink and Lenardic* (2008, 2010); *Höink et al.* (2011, 2012).

As shown in figure 5.1, the strain rate across the upper/lower mantle interface changes by a factor of about two, while the stress decreases by a factor of roughly 50. Effectively, this means that the Stokes equations mainly diffuse the strain rate, not the stress. This can be attributed to the layered structure of the viscosity, in combination with the radial driving force from buoyancy: If we imagine a thin pipe placed vertically within the mantle, the mass flux (and thus the velocity in an incompressible setting) is constant everywhere within the pipe, regardless of the viscosity. The force (and thus the stress) required to move the material through the pipe on the other hand is highly dependent on the resistance (and thus on the viscosity). Since within the mantle, lateral outflow from the imagined pipe is possible, the velocity within the upper mantle can (and does) increase, while the stress does not reduce as much as it would in our vertical pipe. The vertical nature of buoyancy combined with the horizontally layered viscosity does however lead to the observed effect that the interface in the viscosity mainly induces a change in stress, not in strain rate.

In the compressible case, we find a broken symmetry between the up- and downwellings. For downwellings as an example this effect can be understood by looking at two adiabates: One that starts at the upper boundary at reference temperature, and one at the same place but at a lower temperature. Due to the exponential nature of the adiabates, the difference

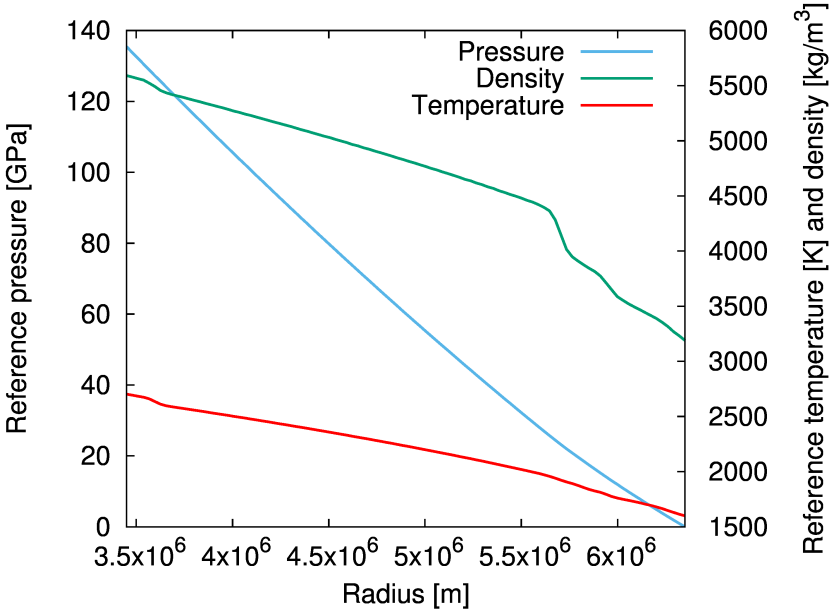

Figure 5.7: Adiabatic reference state for the simulated mantle with pyrolitic composition. The phase transitions at 410 km, 660 km and the perovskite/post-perovskite transition near the CMB are smoothed out, but still well visible in the density.

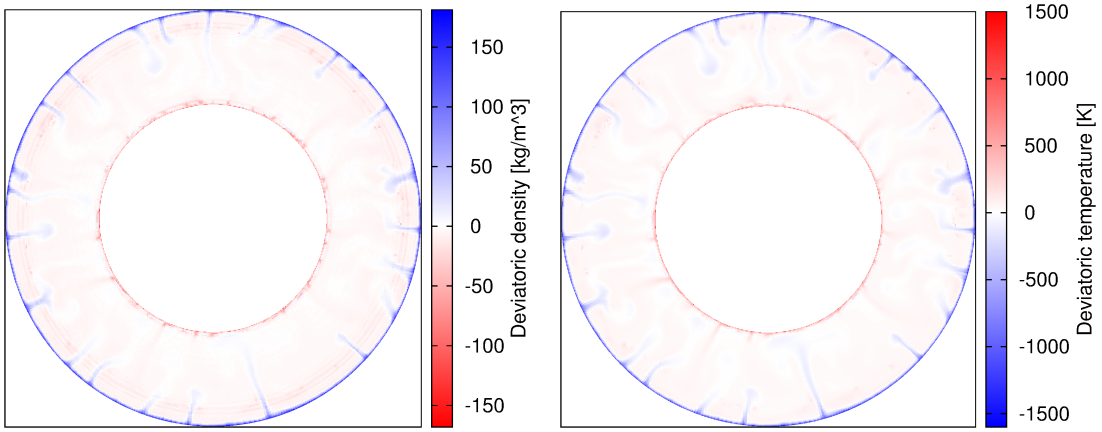

Figure 5.8: Deviatoric density (*left*) and temperature (*right*) of a mantle convection simulations in the TALA approximation with material properties from mineral physics and bottom heating.

in the temperatures increases when the material moves deeper down into the mantle. Since each package of material follows its own adiabate, the temperature difference between a downwelling and its surrounding increases, effectively cooling a downwelling relative to its surrounding on its way down. The same effect in reverse holds for the plumes, effectively weakening them in the process. This is in agreement with findings by *Steinbach et al.* (1989), who reported a break of symmetry between the upper and lower boundary.

In accordance to *Davies et al.* (2012) and *Austermann et al.* (2014), our preliminary simulations with mineralogical models suggest that the low shear velocity provinces as reported by *Trampert et al.* (2004) could possibly be caused by thermal effects alone, without invoking chemically distinct material. However, a more detailed analysis would be required to make a reliable statement on the matter.

Finally, a word of caution is necessary for the models with realistic mineralogical properties, since there is a great source of uncertainty within the numerics, in a sense that that running a model with an explicit solver yields numerical instabilities near the phase transitions. To explain this phenomenon, let us assume that we have a warm anomaly below an endothermic phase transition. Buoyancy moves this anomaly upwards across the interface, cooling it down in the process. Since we are using a finite time step, the package is potentially moved far beyond the phase transition, and thus cooled far more than it would be in nature. In the next time step, the newly acquired very low buoyancy will move the package back down, heating it up again. This numerical artifact leads to strong and well observable small-scale oscillations along the transition zone, completely dominating the convective patterns within the simulation. We can counter this effect to some degree with an implicit time stepping scheme, but sometimes the implicit Euler scheme does not converge to a solution. Resetting the implicit solver with new parameters and a new initial search window can help, but sometimes we still run into unresolvable non-converging time-steps.

A different way to deal with phase transitions is described i.e. in *Nakagawa et al.* (2009), where compositional fields are tracked and latent heat is added and subtracted as material changes its phase. However, to the best of my knowledge, there exists no discussion on potential numerical instabilities in this procedure. The unpredictable behavior in our scheme thus calls for caution with the above interpretation of the models that incorporate a realistic mineralogy.

6 Hierarchical Hybrid Grids (HHG)

The Hierarchical Hybrid Grids software (HHG) is a massively parallel software package originally designed to solve the Poisson equation on very large domains (*Hülsemann et al.*, 2003; *Bergen and Hülsemann*, 2004; *Bergen*, 2005). It is based on block-structured tetrahedral grids, combining an unstructured coarse grid with structured refinements of the tetrahedral blocks as proposed by *Bey* (1995). HHG has been able to solve a linear system of equations with 10^{10} unknowns (*Bergen et al.*, 2005) on an SGI Altix supercomputer in 2005, scaled to 10^{11} unknowns on the HLRB-II in 2008 (*Gradl and Rüde*, 2008; *Gradl et al.*, 2008) and to 10^{12} unknowns on the Juqueen supercomputer in 2013 (*Gmeiner et al.*, 2014a).

In 2013, HHG became the prototype for a new mantle convection code called Terraneo. In this context, significant development has taken place: The first extension of HHG to a solver for the Stokes system with the Schur-complement/pressure correction/multigrid algorithm (as introduced in section 3.5) has been presented by *Gmeiner* (2013) and *Gmeiner et al.* (2014a). They have scaled the code to the full Juqueen supercomputer and shown that the code is bound by computations, not memory bandwidth, which is an unusual property for a high performance code like HHG. Variable viscosity has also been introduced to HHG by *Gmeiner* (2013) in a way that preserves the low memory-bandwidth requirements of the code. The transport problem for the heat equation has been addressed by *Gmeiner et al.* (2014b) and *Waluga et al.* (2015), presenting a way to enforce local mass conservation for the Q1-Q1 finite element pair. New solvers for the Stokes system and free slip boundary conditions have been implemented by *Huber et al.* (2015), enabling a shorter time to solution and physically more realistic boundaries.

6.1 The HHG Parallelization Concept

HHG is designed for first order finite elements, and so the unknowns can be mapped to the points of a mesh. To minimize communication, HHG uses a concept that subdivides the grid into four so called primitives: vertices, edges, faces and volumes. The coarse grid nodes are stored in a vertex data structure, and all fine grid points along the edges that connect the vertices are stored on an edge data structure. The points on the faces between the tetrahedra are stored in a triangular face data structure, and all remaining points within the coarse grid tetrahedron have a tetrahedral volume data structure. All computations (smoothings, residual calculations and operator applications, in the following called lattice updates) are done locally on each primitive. Since they sometimes require values from outside their primitive, each primitive gets so called ghost layers, which shadow all points from the neighboring primitives that are needed for the computations. The resulting data layout is visualized in figure 6.1.

For the parallelization, the primitives are distributed among the processes. Ghost layers

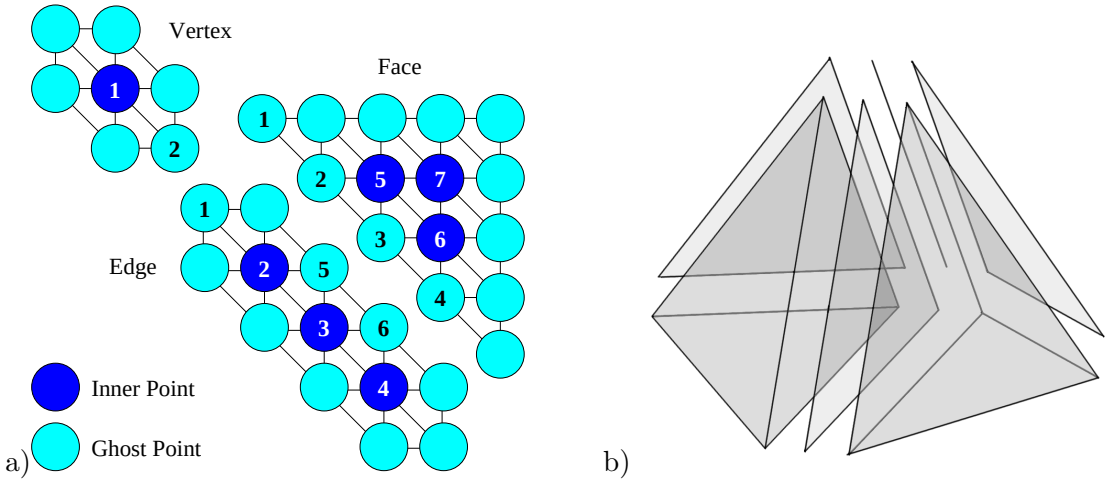

Figure 6.1: HHG data layout. **a)** 2d simplification of vertex, edge and face primitives after two refinements. The inner points are numbered arbitrarily, with their ghost copies labeled correspondingly. **b)** Two macro tetrahedra with some of the neighboring faces and edges.

are exchanged via local copy operations for primitives that share common memory, and by message passing for primitives on remote compute nodes. For the smoother, this means that HHG implements a pseudo-Gauss Seidel algorithm, since it exchanges the ghost layers only after every smoothing sweep, using values from before the smoothing sweep on the ghost nodes.

Much of the high performance of HHG stems from the fact that each primitive element is regularly refined, such that the stencils are identical for all fine grid points within a primitive, and can be assembled in the setup phase (*Bergen*, 2005; *Gmeiner et al.*, 2014a). This way, the stencil does not have to be loaded from memory, and each lattice update requires only 29 floating point operations.

For variable viscosity, the stencil varies depending on the viscosity. *Gmeiner* (2013) has implemented a scheme in which instead of the 15 stencil values, the 96 entries of the local element matrix get stored, a quantity that is once again the same for each point within a coarse grid tetrahedron. For each node, the stencil can then be assembled on-the-fly from the local element matrix and the viscosity, now requiring 227 floating point operations per lattice update, but preserving the low-memory bandwidth requirements of HHG (*Gmeiner et al.*, 2014a).

6.2 Projected Coordinates

6.2.1 Motivation

To simulate the Earth's mantle in HHG, an icosahedral mesh as described in section 3.3 is used as the coarse grid. This mesh consists of $20 \cdot (n_r - 1) \cdot (n_t - 1)^2$ prismoids, each divided into three coarse grid tetrahedra. Each coarse grid tetrahedron is then n_{level} times regularly refined, leading to a total of $60 \cdot (n_r - 1) \cdot (n_t - 1)^2 \cdot n_{\text{level}}^3$ fine grid tetrahedra. The problem with this procedure is that the additional nodes that we get from the refinement process are not on the original spherical surface. Assuming $n_t = 65$ on the coarse grid (as in the HPC run by *Gmeiner et al.*, 2014a), we get a distance of $\sim 100\,\text{km}$ between the coarse grid nodes.

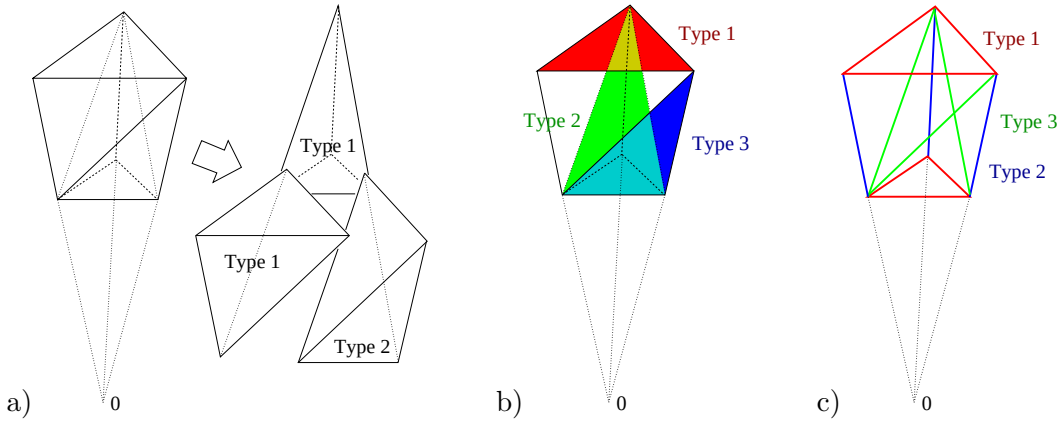

Figure 6.2: Examples for the different possible types of primitives for the projection. **a)** Two types of tetrahedra: Type 1 with three nodes on one spherical surface, type 2 with two nodes on two distinct spherical surfaces each. **b)** Three types of faces: Type 1 with three tangential edges, type 2 with one tangential and two diagonal edges, and type 3 with one tangential edge, one radial edge and one diagonal edge. **c)** Three types of edges: Type 1 tangential, type 2 radial, and type 3 diagonal.

This means that the direct line between two of these nodes from the regular refinement is up to $\sim 200\,\mathrm{m}$ below the spherical surface. For a typical run on a mid-size cluster we could assume $n_t = 17$, corresponding to a distance of $\sim 400\,\mathrm{km}$ between the coarse grid nodes. With this we would obtain points that are up to $\sim 3.1\,\mathrm{km}$ below the spherical surface.

We would like to note two issues that arise from this setup: First, the buoyancy term near the surface and the core mantle boundary (and, to a lesser degree, near the phase transitions) is distorted due to the additional mass that is contained within the bulges which arise from the non-spherical surface. Assuming a typical lateral density variation of 2% (*Ritsema et al.*, 2011; *Stixrude and Lithgow-Bertelloni*, 2005), this bulge would correspond to a density anomaly of 10 km thickness for the HPC scenario, and 155 km for the mid-size cluster scenario. This effect can be seen in some of the preliminary simulations with HHG, where up- and downwellings form preferentially along the coarse grid nodes. Second, we are interested in geoid and dynamic topography signals from the simulations. These can be derived from independent observations of the Earth and can be calculated in the simulations from the stress state near the surface. Typical geoid signals are on the order of 100 m (e.g. *Hager and Richards*, 1989; *Brockmann et al.*, 2014), while dynamic topography signals are typically on the order of 1 km (e.g. *Winterbourne et al.*, 2009). Distortions in the surface geometry of a few hundred meters to several kilometers would render these tests against independent observations impossible.

6.2.2 Projection

We can avoid these problems if we adjust the local element matrices and the stencils such that they represent a geometry in which the fine grid nodes are projected onto the sphere. Since HHG does not store geometric information on the fine grid, the first challenge is to determine the coordinates of all fine grid nodes from the coordinates of the nodes of the coarse grid primitive. For this, we assume that the center of the spherical mesh is at the origin of the coordinate system and distinguish nine different types of primitives as shown in figure 6.2:

- *Tetrahedron type 1:* Three of the four coarse grid nodes are on one spherical surface,

while the fourth is at a different radius. As shown in figure 6.3, we call the node at a separate radius d, the one radially above or below r, and the remaining two nodes ea and eb. We know the fine grid coordinates of a given point p on the unprojected mesh. To obtain the coordinates of the projected point p' we first construct the plane f parallel to the plane $(r,\ ea,\ eb)$ with the normal vector n and the point of support p. Intersecting f with the line $(r,\ d)$ gives us the point q, which has the same distance from the origin as p'. We can now obtain p' by projecting p to the radius given by q.

- *Tetrahedron type 2:* Two of the coarse grid nodes are on one spherical surface, and two on another. One of these nodes each are radially above each other. Of the other two we pick one at random and project it to the other spherical plane. Now we know the three nodes from one of the bases of the prism, and we can proceed as for tetrahedra of type 1.

- *Face type 1:* In this case, all three coarse grid nodes are on the same spherical surface, and we simply have to project all inner nodes to the same radius as any of the coarse grid nodes.

- *Face type 2:* Here, two coarse grid nodes are on the same spherical surface, and the third node is on a different surface and not radially above or below one of the other nodes. In this case, we call the two nodes on the same surface ea and eb, the third node d, and obtain a node r by projecting d to the same spherical surface as ea and eb. From here we can then continue as for tetrahedra of type 1.

- *Face type 3:* Here we have two coarse grid nodes radially above each other which we call r and d, and one on the same spherical surface as r, called e. Instead of a plane we now set up a straight line parallel to the line $(r,\ e)$ that passes through p. The intersection of this line with the line $(r,\ d)$ gives us the point q, which then tells us to what distance from the origin we have to project p.

- *Edge type 1:* Both points are on the same spherical surface. As for the face of type 1, we can simply project all nodes on the edge to the distance from the origin of either of the coarse grid nodes.

- *Edge type 2:* Both coarse grid points are radially above each other. In this case, all fine grid nodes are at the correct distance from the origin, and we do not have to project.

- *Edge type 3:* Both nodes are on different spherical surfaces and not radially above each other. In this case, we exploit the fact that the distance of the fine grid nodes from the origin scales linearly between the two coarse grid nodes. We can then scale the radial distances from the origin linearly with the distance of the unprojected points to the coarse grid nodes.

- *Vertex:* All coarse grid vertices are on the correct spherical surface, so no projection is necessary.

For the basic geometric operations, special care has to be taken with nodes along the main coordinate axes to avoid divisions by zero. Extra if-clauses are introduced to catch these cases and treat them correctly.

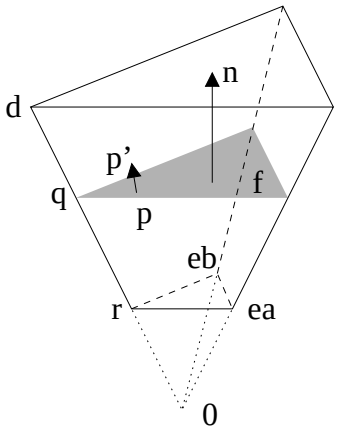

Figure 6.3: Projection of a point p within a prismoid to the point p' on a spherical surface.

Once we know the coordinates of the fine grid nodes, we can evaluate the local element matrices and compute the stencils from them and from the viscosity. An exemplary implementation of a Gauss-Seidel smoother on a tetrahedron with projected coordinates and constant viscosity is shown in appendix A. Here, we calculate the local element matrix for each of the 24 tetrahedra that are adjacent to the current node and add the corresponding entries to obtain the stencil. For variable viscosity, we would then scale the element matrices with the viscosity in the elements. Finally, we can apply the stencil as in the unprojected smoother.

6.2.3 Analysis

For the unprojected, constant coefficient case, every lattice update requires 29 floating point operations: 15 multiplications to multiply the stencil weight with the corresponding neighbor and 14 additions to add the contributions from the neighbors together. For variable viscosity as implemented by *Gmeiner* (2013) we need an additional 16 operations to average the node-based viscosities to the elements, 181 operations to assemble the stencil from the precomputed element matrices and the averaged viscosity, and one division to obtain the inverse of the central stencil entry, giving a total of 227 operations. For the projected coordinates, we now also have to calculate the local element matrices, at the cost of one function call and 53 operations per element matrix for the Laplace operator. Since we have to calculate 24 element matrices, we get a total of 1272 additional operations for these cases. Table 6.1 lists an overview of the number of operations for the different cases, and gives runtime measurements for an example run. In it we solve the scalar Poisson problem with $n_r = 3$ and $n_t = 5$ on the

Projected coordinates?	**Variable viscosity?**	**FLOPs / lattice update**	**runtime $\mathbf{n_{level} = 3}$**	**runtime $\mathbf{n_{level} = 5}$**	**runtime $\mathbf{n_{level} = 7}$**
no	no	29	0.17 s	0.85 s	28 s
no	yes	227	0.90 s	2.6 s	63 s
yes	no	1483	0.40 s	9.1 s	610 s
yes	yes	1499	1.4 s	14 s	700 s

Table 6.1: Number of floating point operations for the smoothing of a single grid point with a Laplacian in four different scenarios. Also shown is the corresponding runtime for one V-cycle for the Poisson problem on a BorgCube machine (see text).

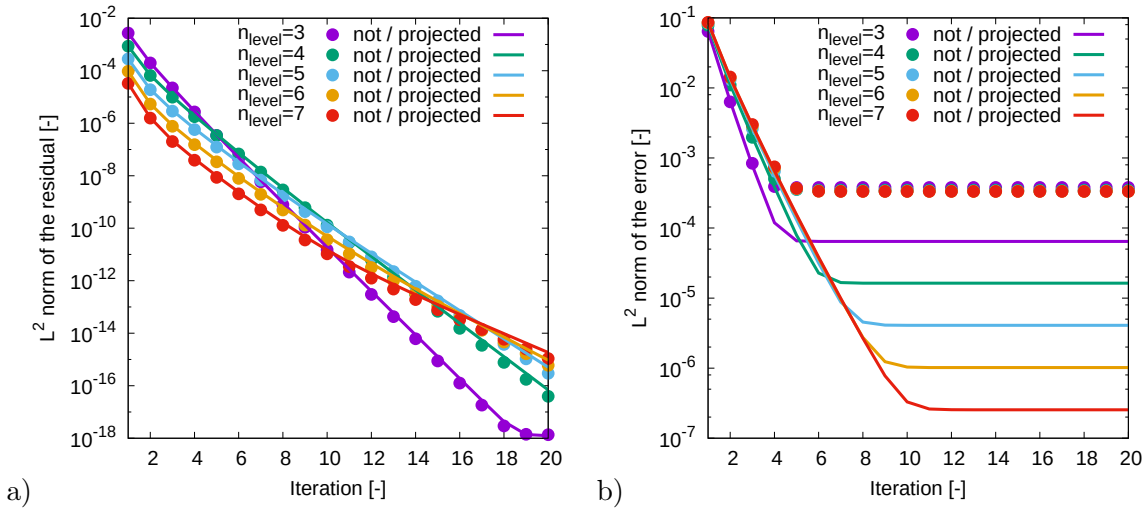

Figure 6.4: Residual (a) and error against an analytical solution (b) of the HHG multigrid solver for the Poisson problem.

coarse grid, which we refine n_{level} times for the fine grid. We use a V-cycle with five pre- and post-smoothing steps and a Gauss-Seidel coarse grid solver with 100 iterations. The problem has been run on a BorgCube ccNUMA machine with 32 virtual cores at 2.7 GHz. For a high number of refinements n_{level}, the wall clock time increases by about a factor of ten for the projected coordinates compared to the unprojected, variable viscosity case, owing to an increase in the required number of computations of about seven and the additional function calls for the computation of the local element matrices. A more detailed analysis and performance optimizations for the projected coordinate case are in preparation (*Bauer et al.*, 2016).

To assess the quality of the implementation, we use the above setup to solve the scalar Poisson problem

$$\Delta v(x,y,z) = -3\sin(x)\sin(y)\sin(z) \quad \forall\,(x,y,x) \in \Omega \tag{6.1}$$

$$v(x,y,z) = \sin(x)\sin(y)\sin(z) \quad \forall\,(x,y,x) \in \partial\Omega. \tag{6.2}$$

The analytical solution to this problem is $v(x,y,z) = \sin(x)\sin(y)\sin(z)\ \ \forall\,(x,y,x) \in \Omega$. We start with an initial solution of $v(x,y,z) = 0\ \ \forall\,(x,y,x) \in \Omega$, and show the residual and the error between numerical and analytical solution in the L^2 norm (figure 6.4).

The residual plot represents the convergence of the multigrid algorithm, showing very similar behavior for the projected and the unprojected case. The residual between iteration 5 and 15 reduces by a factor of 6.9 for $n_{\text{level}} = 3$, and by 2.9 for $n_{\text{level}} = 7$, so multigrid convergence is better for a coarser problem and for a shallower multigrid hierarchy. To assess the cause in more detail, one would have to analyze the eigenvalues of the iteration matrix (i.e. *Baumann and Mohr*, 2015).

The error in the unprojected case is dominated by the geometric error from the edgy domain. A finer resolution does not significantly increase the quality of the solution, since the coarse grid and thus the geometry stays the same for all refinement settings. In the projected case, the error reduces by an average factor of 8.2 for $n_{\text{level}} = 3$, and by 4.3 for $n_{\text{level}} = 7$ between the first few iterations, before it converges to the discretization error. The discretization error decreases by an average factor of 3.989 with every additional refinement, representing

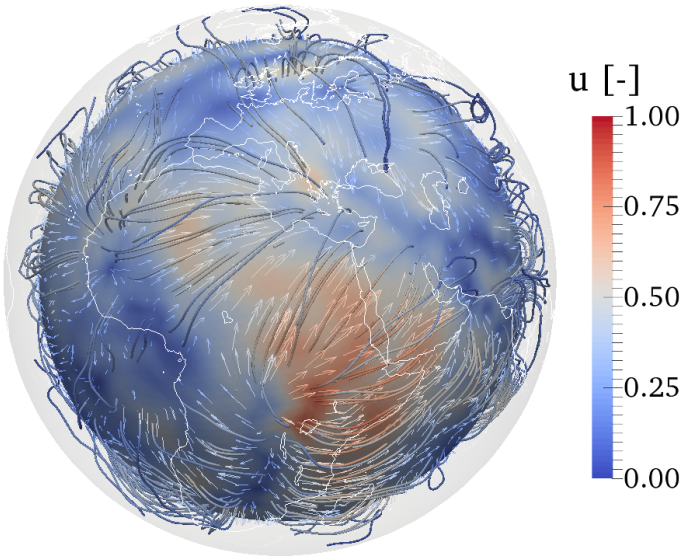

Figure 6.5: Visualization of the asthenosphere in a static mantle convection simulation with a buoyancy from tomography (see text). Color-coded is the normalized absolute of the velocity, not the seismic tomography. Well-visible is the fast upwelling region under the African superswell.

quadratic convergence, as would be expected from the theory of linear finite elements (see e.g. *Gunzburger*, 1989; *Braess*, 2007).

6.3 Application to the Earth

As an example for an application of HHG to the Earth, we calculate the velocity field that arises from a tomography-derived buoyancy term in a static simulation. We set up a spherical shell with nondimensionalized radii that correspond to the Earth mantle, $r_{\text{lower}} = 0.545$ and $r_{\text{upper}} = 1$. We use a two-layered viscosity profile with an upper-to-lower mantle boundary at $r = 0.843$ (corresponding to a depth of 1000 km) and a viscosity that is ten times smaller in the upper than in the lower mantle, in agreement with the Haskell-constraint (*Haskell*, 1935; *Mitrovica*, 1996). Both boundaries are no-slip, and the buoyancy is derived from the tomographic model by *Grand et al.* (1997), converted to densities with the mineralogic model by *Stixrude and Lithgow-Bertelloni* (2005). We use a resolution of $n_r = n_t = 129$, resulting in a total of approximately 21 million unknowns. The system is solved with a pressure correction Schur CG solver and a four-level V-cycle multigrid on a BorgCube machine with 32 virtual cores.

In the nondimensional setting, absolute velocities scale linearly with the Rayleigh number, which is a poorly constrained parameter for the Earth. We therefore show a normalized velocity field in figure 6.5. Features such as the fast upwelling speeds below the African superswell, old subduction under the Eastern Mediterranean and the upwelling around the Icelandic plume can be identified. For a more realistic setup, future simulations will incorporate free-slip boundaries and a more sophisticated upper boundary (i.e. from plate reconstructions) as well as a more shallow asthenosphere. Time-dependent simulations, ideally inverse and backwards in time, together with realistic, nonlinear rheologic models and mineralogical tables are the long-term goal.

7 Fast Asthenosphere Motion in High-Resolution Global Mantle Flow Models

A variety of geologic observations point to fast upper mantle flow that may exceed plate tectonic velocities by an order of magnitude. At the same time there is mounting evidence from seismology for flow like structures in the upper $100 - 200$ km of the mantle. In this chapter we present a set of simulations done with HHG to link these observations. In a synthetic setting, we include asthenospheric channels of varying thickness, with an extreme case of 100 km, and a significant viscosity contrast of up to four orders of magnitude relative to the deeper mantle. We obtain an increase in velocity by a factor of ten between a 1000 km thick and the very thin channel, translating into velocities of ~ 20 cm/a within the narrow asthenosphere. We further present and verify a simple Poiseuille flow model, predicting that the upper mantle velocity scales with the inverse of the asthenosphere thickness.

The work in this chapter constitutes a collaborative effort between myself, Björn Gmeiner, Siavash Ghelichkhan, Markus Huber, Lorenz John, Barbara Wohlmuth, Ulrich Rüde and Hans-Peter Bunge. The text has previously been published as *Weismüller et al.* (2015).

7.1 Introduction

The asthenosphere is a mechanically weak layer beneath the lithosphere that was advocated early on as a means to promote isostatic movement (*Barrell*, 1914) and lubricate plate motion (*Chase*, 1979). Robust evidence for a weak layer in the uppermost mantle beneath the lithosphere comes from a variety of observations. They include studies of the geoid (e.g. *Hager and Richards*, 1989), glacial rebound (e.g. *Mitrovica*, 1996), oceanic intraplate seismicity (*Wiens and Stein*, 1985), ocean ridge bathymetry (*Buck et al.*, 2009), seismic anisotropy (e.g. *Debayle et al.*, 2005), and electromagnetic sounding (e.g. *Jones*, 1982). A cause for low mechanical strength in the asthenosphere is seen in weakening effects related to partial melting (e.g. *Anderson and Sammis*, 1970) and/or water (e.g. *Karato and Jung*, 1998), or a combination of pressure and temperature (e.g. *Stixrude and Lithgow-Bertelloni*, 2005). A consequence is the concentration of upper mantle flow into a thin channel of greatly enhanced material mobility. Fluid dynamic studies based on numerical and analytic modeling techniques (e.g. *Bunge et al.*, 1997; *Busse et al.*, 2006; *Tackley*, 1996) agree that high material mobility in the asthenosphere is essential to promote the long-wavelength character of mantle flow observed on Earth.

Interest in the asthenosphere has been renewed lately. New upper mantle tomographic models based on computationally demanding full waveform modeling techniques suggest a narrow width of the asthenosphere. The seismic studies have imaged a thin horizontal layer of $\sim 100 - 200$ km depth extent beneath the lithosphere, with heterogeneity structures indicative

of channeled flow (e.g. *Fichtner et al.*, 2009; *French et al.*, 2013; *Colli et al.*, 2013). At the same time, a variety of geological observations suggest that fast flow velocities in the asthenosphere may exceed plate tectonic speeds by an order of magnitude in some cases. These observations include V-shaped ridges of thickened oceanic crust near Iceland, which have been attributed to the fast propagation of plume related material (*Vogt*, 1971; *Parnell-Turner et al.*, 2014), rapid uplift events along passive margins (*Japsen et al.*, 2012a) together with strong gradients in dynamic topography (*Winterbourne et al.*, 2009), as well as transient landscapes in the North Atlantic sea floor, now buried beneath marine sediments. The latter were used to infer the horizontal passage of buoyant sublithospheric material at velocities in excess of 20 cm/a (*Hartley et al.*, 2011).

A series of papers by *Höink and Lenardic* (2008, 2010) and *Höink et al.* (2011, 2012) has explored the geodynamic implications of a thin low viscosity layer for mantle convection models in the context of Poiseuille/Couette flow, i.e. pressure driven vs. plate driven flow. The pressure driven Poiseuille flow type can be tied to geologic observables, as it relates plate motion changes explicitly to non-isostatic vertical motion, i.e., changes in dynamic topography (*Iaffaldano and Bunge*, 2015). Poiseuille flow, therefore, provides a plausible link between rapid spreading changes and coeval episodes of epeirogenic motion (*Colli et al.*, 2014) that is documented in the South Atlantic region (e.g. *Japsen et al.*, 2012b).

The amount of viscosity reduction in the asthenosphere relative to the deeper mantle is not well known. A consensus exists that the average viscosity in the upper part of the mantle is smaller than at greater depth. But the magnitude of the viscosity increase and its depth remain debated. *Haskell* (1935) calculated the viscosity of the upper part of the mantle to be $\sim 10^{21}$ Pa s, a value known as the Haskell constraint. While *Mitrovica* (1996) showed that the Haskell constraint applies effectively to mantle depths of $1000 - 1200$ km, *Paulson and Richards* (2009) and *Schaber et al.* (2009) emphasized the inherent trade-off between radial viscosity contrast and layer thickness: models favoring a coarse subdivision of the mantle into two layers separated at the 660 km phase transition naturally obtain modest viscosity contrasts, while fitting the data as well as models with a thin layer and strong viscosity reduction. The trade-off arises because the decay time τ for a given loading wave number varies linearly with the viscosity μ and inversely with h^3 (e.g. *Cathles*, 1975), where h is the layer thickness. These results, when combined with the seismic evidence for a narrow channel width, argue for a substantial viscosity reduction in the asthenosphere.

It is now feasible to simulate global mantle flow at very high numerical resolution on massively parallel architectures using more than 100 000 processor cores (*Burstedde et al.*, 2009, 2013; *Kronbichler et al.*, 2012). In this article, we employ the TERRA-NEO prototype HHG to explore the mantle flow field in a three dimensional spherical geodynamic model. Specifically, we focus on the velocity contrast that develops between the high viscosity deeper mantle and the asthenosphere in the presence of a strongly localized low viscosity channel. We find fast asthenosphere motion in our models, and report that the velocity increases by more than a factor of ten when decreasing the asthenosphere channel thickness by a factor of ten. We use our results to derive a simple scaling relation for the asthenosphere/deeper mantle velocity contrast based on Poiseuille flow.

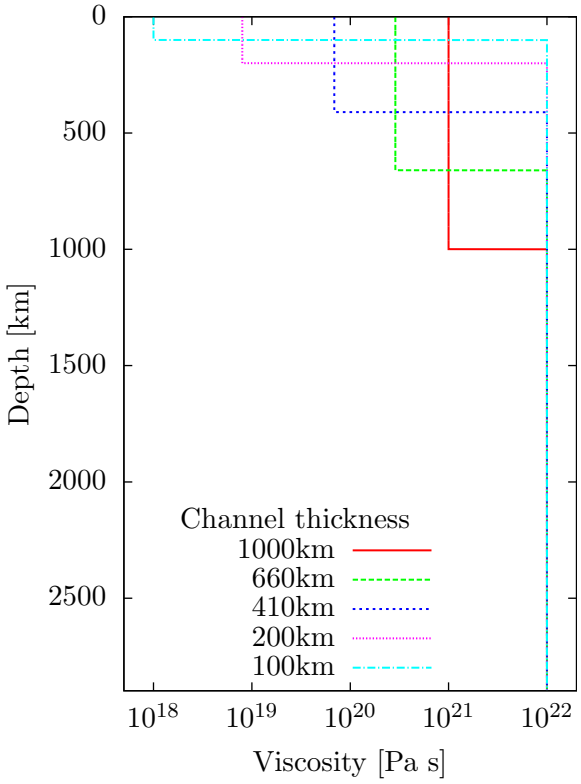

Figure 7.1: Viscosity profiles of the five simulation scenarios explored in this study. The lower mantle viscosity is assumed as 10^{22} Pa s, while the upper mantle viscosity of the 1000 km reference scenario (red) is set to the Haskell value (see text). Two cases with a channel thickness corresponding to either the 660 or 410 km mantle phase transitions (green, blue), or a thin channel scenario with a 200 or 100 km layer depth (magenta, light blue) are shown as well.

7.2 Methods and Results

Our code is based on the hierarchical hybrid grids (HHG) multigrid software framework (*Bergen et al.*, 2006; *Gmeiner et al.*, 2014a), which is optimized for high performance computing and scales to some of the fastest systems currently in use (*Gmeiner and Rüde*, 2014). This allows our global mantle flow solutions to treat different types of boundary conditions and to vary the key controlling parameters, i.e., viscosity reduction and channel width, systematically over a wide range in accord with the Haskell constraint. We perform our computations on the JUQUEEN Supercomputer at Jülich Supercomputing Center using up to 65 536 processes for a thin-layer asthenosphere scenario with a channel width of 100 km. For this scenario, we use a total of $2.3 \cdot 10^{10}$ degrees of freedom, resolving the asthenospheric channel with 32 radial layers. We assume the Boussinesq approximation for the stationary Stokes equations. Nondimensionally they are of the form

$$-\nabla \cdot \left(\frac{\mu}{\mu_0} \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^t \right) \right) + \nabla p = -\mathrm{Ra}\, T\, \hat{\mathbf{e}}_\mathbf{r} \tag{7.1}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{7.2}$$

and describe the momentum balance for creeping flow, i.e. flow that ignores the effects of inertia. Here, $\mathbf{u}$ [-] denotes the velocity, μ [Pa s] the dynamic viscosity, $\mu_0 = 10^{22}$ Pa s a normalization factor for the viscosity, p [-] the pressure, Ra [-] the Rayleigh number, T [-] the buoyancy term (e.g. temperature) and $\hat{\mathbf{e}}_\mathbf{r}$ [-] the unit vector in radial direction. We note that the momentum balance is linear and yields a direct scaling with the Rayleigh number between driving buoyancy forces and resulting flow velocities. This means that the absolute value for the upper and lower mantle velocities will depend linearly on the choice for the

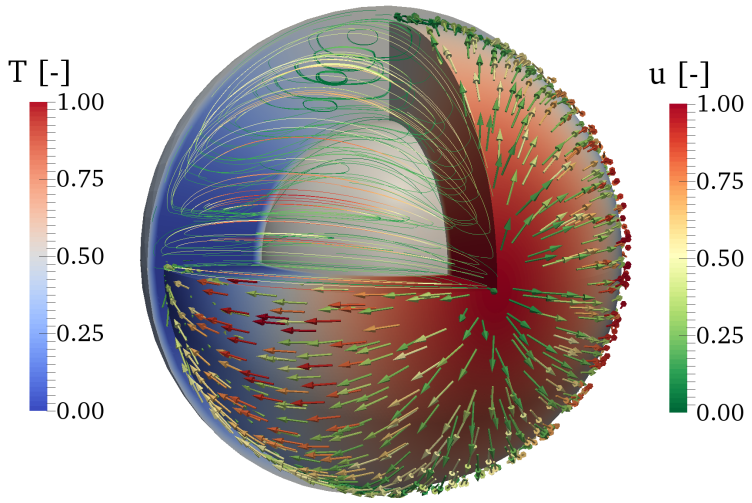

Figure 7.2: Visualization of the simulation setup and the resulting velocity field for the scenario with 410 km channel depth and a stagnant lid. The left color bar represents the nondimensional temperature (and thus the buoyancy field Y_2^2, red = hot, blue = cold), which is cut off above 410 km depth in the front half of the sphere. In the asthenosphere, flow vectors from the associated nondimensional velocity field are superimposed (right color bar, red=fast, green=slow). On the top left part of the figure, the shell is cut open all the way to the core mantle boundary, and flow is visualized by flow lines instead of arrows. Here it becomes apparent that high velocities are concentrated in the asthenosphere.

amplitude of the mantle buoyancy field. We will thus use the average channel velocity of the 1000 km reference scenario as our reference velocity and for the thinner channel cases report the average velocity relative to this reference value.

Heterogeneity in the Earth's mantle is dominated by large scale structure, concentrated at the lowest spherical harmonics degree two (*Ritsema et al.*, 2011; *Grand et al.*, 1997). Hence we choose a long wavelength heterogeneity field for our simulations, and set the buoyancy term to the spherical harmonics term $Y_{l=2}^{m=2}$, normalized in the range $[0, 1]$ and multiplied by a linear decrease to zero radially within 225 km from the upper and lower boundary. (*Solomatov*, 1995), field of Channelized flow in the asthenosphere is approximated by modeling mantle flow under the two end member cases of a stagnant and a mobile lid regime (*Solomatov*, 1995). For the stagnant lid, we impose a no-slip condition ($u = 0$) on the surface velocity field of the shell, while the mobile lid is approximated with a free-slip condition (no outflow, no tangential stress). We vary the thickness h_u of the low viscosity layer systematically between values of 1000 km and 100 km. This setup yields a simplified two-layer radial viscosity profile, where the lower mantle viscosity assumes a fixed value of $\mu_l = 10^{22}$ Pa s, while the weak layer viscosity μ_u is specified in accordance with the Haskell constraint (*Haskell*, 1935; *Mitrovica*, 1996). In other words, μ_u scales with the third power of the channel thickness (e.g., *Cathles* (1975)), as suggested by *Paulson and Richards* (2009) and *Schaber et al.* (2009), such that

$$\mu_u = 10^{21} \times \left(\frac{h}{1000 \text{ km}} \right)^3 \text{ [Pa s]}. \tag{7.3}$$

Fig. 7.1 reports the resulting viscosity profiles for five scenarios. Our reference case has a 1000 km channel thickness, i.e. the Haskell depth, with $\mu_u = \mu_0 = 10^{21}$ Pa s. In two further cases we reduce the channel thickness to 660 km and 410 km, i.e., depth levels associated

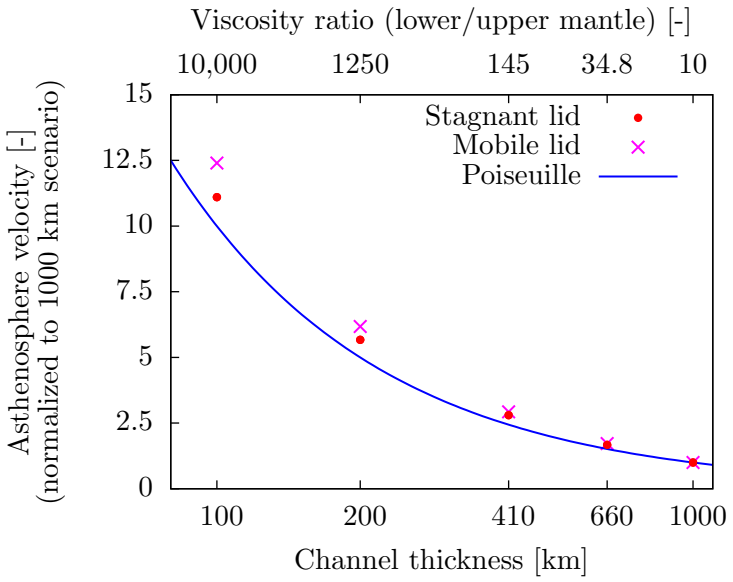

Figure 7.3: Average flow velocities in a low viscosity asthenosphere channel relative to a reference scenario with a channel thickness of 1000 km. High flow velocities (left axis) arise for thin channel widths (lower axis). The blue curve represents theoretical predictions from Poiseuille flow, red dots and purple crosses are simulation results from the no-slip and free-slip upper boundary condition, respectively. The upper axis gives the asthenosphere viscosity corresponding to the channel thickness according to the h^3 law (equation 7.3).

with the major mantle phase transitions. Additionally, two extreme cases are modeled. Here we assume a narrow depth extent for the asthenosphere of 200 km and 100 km, respectively, as suggested by the seismic observations. Table 7.1 gives an overview of the key model parameters (viscosity and channel thickness) for the different mantle flow simulations. In Fig. 7.2 we show the buoyancy and flow fields for the 410 km scenario as an example of our simulations. In this, as in all other cases, we obtain two large scale upwelling and downwelling regions in the lower mantle. They are induced by the degree two buoyancy term we have imposed. In the asthenosphere we find the flow field to be dominated by lateral motion. This is evident from the horizontal orientation of flow lines in this region, as seen in Fig. 7.2. The flow velocities increase systematically with decreasing channel depth and viscosity, with a slightly stronger increase for the mobile lid cases. The scenario with a low viscosity layer of 660 km thickness yields an increase in the average velocity of about 1.7 compared to the 1000 km reference scenario for both stagnant and mobile lid regime. Our end member cases, where we assume a narrow 100 km channel thickness, yield an increase in the average upper mantle velocity of 11.1 for the stagnant lid and 12.4 for the mobile lid case. These values are listed in Table 7.1 and plotted in Fig. 7.3.

channel thickness	asthenosphere viscosity (μ_u)	asthenosphere velocity (stagnant lid)	asthenosphere velocity (mobile lid)
1000 km	10^{21} Pa s	1.0	1.0
660 km	$2.87 \cdot 10^{20}$ Pa s	1.67	1.72
410 km	$6.89 \cdot 10^{19}$ Pa s	2.80	2.93
200 km	$8 \cdot 10^{18}$ Pa s	5.67	6.18
100 km	10^{18} Pa s	11.1	12.4

Table 7.1: Parameter values of the simulations showing thickness and viscosity of the low viscosity channel, where the lower mantle viscosity is 10^{22} Pa s in all cases. Also shown are the resulting upper mantle average velocities for the stagnant and mobile lid cases, normalized to the 1000 km reference scenario (see text).

7.3 Discussion

Recent geological observations are indicative of sublithospheric flow speeds on the order of 20 cm/a: they include ridges of thickened oceanic crust near Iceland (*Vogt*, 1971; *Parnell-Turner et al.*, 2014), rapid uplift events along passive margins (*Japsen et al.*, 2012a) together with strong variations in dynamic topography (*Winterbourne et al.*, 2009) and transient landscapes in the North Atlantic sea floor (*Hartley et al.*, 2011). Our results, reported as velocities relative to the reference scenario with a channel thickness of 1000 km, are consistent with these observations. They suggest that high asthenosphere velocities arise self-consistently in global, buoyancy driven flow. If we assume an average deep mantle flow velocity of 1.5 cm/a (*Butterworth et al.*, 2014), the thin channel simulations with 200 km and 100 km depth extent yield asthenosphere velocities that average to 8.15 cm/a and 15.7 cm/a for the stagnant lid case, respectively, and 9.78 cm/a and 18.6 cm/a for the mobile lid regime, an order of magnitude higher than average plate velocities (*DeMets et al.*, 2010).

To explain our results, we introduce a simple scaling based on Poiseuille flow, similar to *Höink and Lenardic* (2010): We approximate the asthenosphere as a two dimensional, plane channel, in which the flow is driven by a pressure gradient between the two ends. Applying this model to the reference case as well as to the case we are investigating, the velocity in the channel normalized to the reference velocity is given by

$$\frac{u}{u_{\mathrm{ref}}} = \frac{\mu_{\mathrm{ref}}\, h^2}{\mu\, h_{\mathrm{ref}}^2}. \tag{7.4}$$

Together with the h^3 scaling derived from the Haskell constraint (7.3), this model results in a scaling of the upper mantle velocity of

$$\frac{u}{u_{\mathrm{ref}}} = \frac{h_{\mathrm{ref}}}{h}. \tag{7.5}$$

We note that this model is free of any fitting parameters, and depends on nothing but the channel thickness of the current scenario and that of the reference case. It is plotted in Fig. 7.3, showing a good prediction with errors of consistently less than 15% for the stagnant lid case, and an underprediction of up to 25% for the mobile lid case.

We must qualify our results and point to important limitations in this study. For instance, both models restrict asthenospheric flow to the Poiseuille regime: For the stagnant lid case, the channel is bounded by the surface and the high-viscosity lower mantle, and flow is driven by pressure alone. An alternative view on asthenospheric flow is Couette flow (*Chase*, 1979): Here, the idea is that flow is driven solely by the moving plates above the asthenosphere, in which case asthenosphere velocities can never exceed plate speeds. *Höink and Lenardic* (2010) have shown that both flow regimes may be present in the Earth, and suggested that on small scales (short wavelengths), Poiseuille flow is the dominant component, whereas on larger scales (long wavelengths), flow is dominated by the moving plates, i.e. the Couette regime.

Our stagnant lid simulations show that with a very thin channel, the Poiseuille component alone can reach significant magnitudes also for long wavelengths. Simulations by *Höink and Lenardic* (2010) among others show that adding a Couette component (i.e. a moving plate) can potentially further increase these velocities. While closer examination of this case would be of high interest, our mobile lid scenarios show that channel speeds somewhat above the predictions by the simple Poiseuille model are well within the realm of possibilities.

As another simplification we take the viscosity as constant and Newtonian, even though there is evidence for nonlinear, stress dependent behavior at least within the upper mantle (e.g. *Karato and Wu*, 1993). While it is more difficult to anticipate the effects of a nonlinear rheology on upper mantle flow, it seems reasonable to assume it would yield larger flow velocities. This would render the flow velocity estimates from our study in the upper mantle as conservative. Another point is the fact that models of lithosphere thickness show significant regional variations (e.g. *Artemieva*, 2006). These variations should have a strong influence on asthenosphere flow velocities for the relatively narrow channel thickness considered here. More realistic models including deep cratonic roots thus would be important to consider in future studies.

7.4 Conclusions

A variety of evidence points towards the existence of a weak asthenosphere layer in the upper mantle, although neither its depth extent nor its viscosity are well known. Using a new global high resolution mantle convection model, we explore a parameter space of global mantle flow with a very strong viscosity reduction in a very thin asthenospheric channel. While traditionally the asthenosphere is assumed to extend to the depth of the large seismic discontinuities at 660 or 410 km, observational constraints allow for much thinner and weaker channels. In our simulations we have used channel thicknesses of down to 100 km, and shown that velocities on the order of 20 cm/a arise naturally in this setting.

Our setup is consistent with the Haskell constraint, which suggests that the viscosity of the upper $1000 - 1200$ km is on the order of 10^{21} Pa s. We have further argued that the simulations are in agreement with recent seismological studies that find a thin horizontal layer in a depth of $100 - 200$ km as well as seismic heterogeneities in this part of the mantle. The results also agree with recent geological observations of rapid uplift, short wavelength dynamic topography signals and V-shaped ridges of oceanic crust, indicative of high sublithospheric velocities.

8 Conclusions

The equations that govern mantle convection are a Stokes-type approximation of the Navier-Stokes equations. The laws of momentum and mass conservation constitute a time-independent elliptic problem, immediatley propagating any change in the boundary conditions or the buoyancy term through the whole domain. Time dependence is incorporated via the energy equation, which is dominated by the advective term throughout most of the domain. Conduction is mainly important near the boundaries, where no radial advective transport is allowed. Complementing the conservation laws, additional material laws are needed to describe the physical behavior of the mantle, most notably the equation of state (coupling density, temperature und pressure) and the rheology (governing the viscosity).

The equations are solved numerically with either first order or mixed first/zeroth order finite elements on an annulus in two dimension and on an icosahedron-based mesh in three dimensions. The resulting system is then solved with a multigrid based iterative solver, coupled with a first or second order explicit or implicit time stepping scheme.

We have presented the new 2d mantle convection code Flatland together with a set of tests and benchmarks. This software has been used to assess the effects of viscosity stratification and mineral physics on global convection patterns. We have shown that a radial jump in the viscosity mainly reduces the stress across the interface, and not the strain rate, and explained that this is due to the radial nature of the buoyancy term. We elaborated on the challenges of including mineral models into convection simulations, and speculated that the D" phase transition could be responsible for the large low shear velocity provinces in the mantle without requiring a chemically distinct material.

We then introduced the high performance software framework HHG as a prototype for a new mantle convection code Terra-neo. HHG is a highly parallel, multigrid based software package that uses an unstructured coarse grid, which is then refined structurally. This approach combines high scalability with high node-level performance, but has some limitations on the geometry of the overall grid. Most importantly for our application, it cannot resolve the curvature between coarse grid nodes, leading to artifacts in the solution. To resolve this issue, we have presented a method to project the fine grid nodes to the sphere while maintaining the block-structured nature of the grid. Tests show that this resolves the geometry artifacts, at the cost however of a reduced node-level performance. Current but unfinished work is aiming to reduce the computational costs of the approach with the help of a polynomial interpolation of the curvature, and a mesh that resolves curvature only near the boundary.

Finally, we have presented an application of the HHG software package to the question of the depth and viscosity of the asthenosphere. We have found that fast velocities of up to 20 cm/a in an asthenospheric channel of 100 km thickness are dynamically feasible and consistent with a wide set of geological direct and indirect observations.

Bibliography

Alisic, L., M. Gurnis, G. Stadler, C. Burstedde, and O. Ghattas (2012), Multi-scale dynamics and rheology of mantle flow with plates, *J. Geophys. Res.*, *117*, B10402.

Anderson, D. L., and C. Sammis (1970), Partial melting in the upper mantle, *Phys. Earth. Planet. Inter.*, *3*, 41–50.

Artemieva, I. M. (2006), Global $1^\circ \times 1^\circ$ thermal model TC1 for the continental lithosphere: Implications for lithosphere secular evolution, *Tectonophysics*, *416*, 245–277.

Atanga, J., and D. Silvester (1992), Iterative methods for stabilized mixed velocity-pressure finite elements, *Int. J. Numer. Methods Fluids*, *14*, 71–81.

Austermann, J., B. T. Kaye, J. X. Mitrovica, and P. Huybers (2014), A statistical analysis of the correlation between large igneous provinces and lower mantle seismic structure, *Geophys. J. Int.*, ggt500.

Barrell, J. (1914), The strength of the earths crust. part vi. relations of isostatic movements to a sphere of weakness - the asthenosphere, *J. Geol.*, *22*, 655–683.

Bauer, S., B. Gmeiner, J. Weismüller, M. Mohr, B. Wohlmuth, and U. Rüde (2016), Efficient on-the-fly stencil assembly for high-performance stencil codes (preliminary authors and title), *in preparation*.

Baumann, S., and M. Mohr (2015), Aspects of multigrid for mantle convection, *Numer. linear Algebr.*, special issue for the 17th Copper Mountain Conference on Multigrid Methods, 2015, submitted.

Baumgardner, J. (1985), Three-dimensional treatment of convective flow in the Earth's mantle, *J. Statist. Phys.*, *39*, 501–511.

Baumgardner, J. R. (1983), A three-dimensional finite element model for mantle convection, Ph.D. thesis, University of California, Los Angeles.

Baumgardner, J. R., and P. O. Frederickson (1985), Icosahedral discretization of the two-sphere, *SIAM J. Numer. Anal.*, *22*, 1107–1115.

Bercovici, D. (2003), The generation of plate tectonics from mantle convection, *Earth Planet. Sci. Lett.*, *205*, 107–121.

Bergen, B., and F. Hülsemann (2004), Hierarchical hybrid grids: data structures and core algorithms for multigrid, *Numer. linear Algebr.*, *11*, 279–291.

Bergen, B., F. Hülsemann, and U. Rüde (2005), Is 1.7×10^{10} unknowns the largest finite element system that can be solved today?, in *Proceedings of the 2005 ACM/IEEE conference on Supercomputing*, IEEE Computer Society Washington, DC, USA, p. 5.

Bergen, B., T. Gradl, F. Hülsemann, and U. Rüde (2006), A massively parallel multigrid method for finite elements, *Comp. Sci. Eng.*, *8*, 56–62.

Bergen, B. K. (2005), Hierarchical hybrid grids: Data structures and core algorithms for efficient finite element simulations on supercomputers, Ph.D. thesis, Technische Fakultät der Universität Erlangen-Nürnberg.

Bey, J. (1995), Tetrahedral grid refinement, *Computing*, *55*, 355–378.

Birch, F. (1947), Finite elastic strain of cubic crystals, *Phys. Rev.*, *71*, 809–824.

Blankenbach, B., F. Busse, U. Christensen, L. Cserepes, D. Gunkel, U. Hansen, H. Harder, G. Jarvis, M. Koch, G. Marquart, D. Moore, P. Olson, H. Schmeling, and T. Schnaubelt (1998), A benchmark comparison for mantle convection codes, *Geophys. J. Int.*, *98*, 23–38.

Braess, D. (2007), *Finite Elemente*, 4 ed., Springer Berlin Heidelberg New York.

Briggs, W., V. Henson, and S. McCormick (2000), *A multigrid tutorial*, 2 ed., Society for Industrial and Applied Mathematics (SIAM), Philadelphia, USA.

Brockmann, J., N. Zehentner, E. Höck, R. Pail, I. Loth, T. Mayer-Gürr, and W. Schuh (2014), EGM TIM RL05: An independent geoid with centimeter accuracy purely based on the GOCE mission, *Geophys. Res. Lett.*, *41*, 8089–8099.

Buck, W. R., C. Small, and W. B. F. Ryan (2009), Constraints on asthenospheric ow from the depths of oceanic spreading centers: The east pacic rise and the australian-antarctic discordance, *Geochem. Geophy. Geosy.*, *10*, Q09007.

Bunge, H. P., and J. Baumgardner (1995), Mantle convection modeling on parallel virtual machines, *Comput. Phys.*, *9*, 207–215.

Bunge, H. P., M. A. Richards, and J. R. Baumgardner (1996), Effect of depth-dependent viscosity on the planform of mantle convection, *Nature*, *379*, 436–438.

Bunge, H. P., M. A. Richards, and J. R. Baumgardner (1997), A sensitivity study of three-dimensional spherical mantle convection at 10^8 Rayleigh number: Effects of depth-dependent viscosity, heating mode, and an endothermic phase change, *J. Geophys. Res.*, *102*, 11,991–12,007.

Burke, K., B. Steinberger, T. H. Torsvik, and M. A. Smethurst (2008), Plume generation zones at the margins of large low shear velocity provinces on the coremantle boundary, *Earth Planet. Sci. Lett.*, *265*, 49–60.

Burstedde, C., O. Ghattas, G. Stadler, T. Tu, and L. C. Wilcox (2009), Parallel scalable adjoint-based adaptive solution of variable-viscosity stokes flow problems, *Comput. Methods Appl. Mech. Engrg.*, *198*, 1691–1700.

Burstedde, C., G. Stadler, L. Alisic, L. C. Wilcox, E. Tan, M. Gurnis, and O. Ghattas (2013), Large-scale adaptive mantle convection simulation, *Geophys. J. Int.*, ggs070.

Busse, F. H., M. A. Richards, and A. Lenardic (2006), A simple model of high prandtl and high rayleigh number convection bounded by thin low-viscosity layers, *Geophys. J. Int.*, *164*, 160–167.

Butcher, J. C. (2008), *Numerical Methods for Ordinary Differential Equations*, John Wiley & Sons New York.

Butterworth, N. P., A. S. Talsma, R. D. Müller, M. Seton, H. P. Bunge, B. S. A. Schuberth, G. E. Shephard, and C. Heine (2014), Geological, tomographic, kinematic and geodynamic constraints on the dynamics of sinking slabs, *J. Geodyn.*, *73*, 1–13.

Cathles, L. M. (1975), *The Viscosity of the Earth's Mantle*, Princeton University Press.

Chase, C. G. (1979), Asthenospheric counterflow: a kinematic model, *Geophys. J. Roy. Astr. S.*, *56*, 1–18.

Chust, T., G. Steinle-Neumann, and H. P. Bunge (2015), EoS - A computational framework for mineralogical thermodynamics (preliminary authors and title), *J. Geophys. Res.*, *in preparation*.

Clark, D. (2015), Intel rechisels the tablet on moores law: The last two technology transitions have signaled that our cadence today is closer to two and a half years than two, *Wall Street Journal Digits Tech News and Analysis*.

Colli, L., A. Fichtner, and H. P. Bunge (2013), Full waveform tomography of the upper mantle in the south atlantic region: Imaging a westward fluxing shallow asthenosphere?, *Tectonophysics*, *604*(0), 26–40.

Colli, L., I. Stotz, H. P. Bunge, M. Smethurst, S. Clark, G. Iaffaldano, A. Tassara, F. Guillocheau, and M. C. Bianchi (2014), Rapid south atlantic spreading changes and coeval vertical motion in surrounding continents: Evidence for temporal changes of pressure-driven upper mantle flow, *Tectonics*, *33*, 1304–1321.

Cordier, P., J. Amodeo, and P. Carrez (2012), Modelling the rheology of MgO under Earth's mantle pressure, temperature and strain-rates, *Nature*, *481*, 177–180.

Davies, D. R. (2008), Applying multi-resolution numerical methods to geodynamics, Ph.D. thesis, Cardiff university, school of Earth, ocean and planetary science.

Davies, D. R., S. Goes, J. Davies, B. Schuberth, H.-P. Bunge, and J. Ritsema (2012), Reconciling dynamic and seismic models of Earth's lower mantle: The dominant role of thermal heterogeneity, *Earth and Planetary Science Letters*, *353-354*, 253–269.

Davies, G. F. (1999), *Dynamic Earth: Plates, Plumes and Mantle Convection*, Cambridge University Press.

Davies, G. F., and M. A. Richards (1992), Mantle convection, *J. Geol.*, *100*, 151–206.

Debayle, E., B. Kennett, and K. Priestley (2005), Global azimuthal seismic anisotropy and the unique plate-motion deformation of australia, *Nature*, *433*, 509–512.

DeMets, C., R. G. Gordon, and D. F. Argus (2010), Geologically current plate motions, *Geophys. J. Int.*, *181*, 1–80.

Dohrmann, C. R., and P. B. Bochev (2004), A stabilized finite element method for the stokes problem based on polynomial pressure projections, *Int. J. Numer. Meth. Fluids*, *46*, 183–201.

Dubuffet, F., M. Rabinowicz, and M. Monnereau (2000), Multiple scales in mantle convection, *Earth Planet. Sci. Lett.*, *178*, 351–366.

Dziewonski, A. M., and D. L. Anderson (1981), Preliminary reference Earth model, *Phys. Earth Planet. Inter.*, *25*, 297–356.

Fichtner, A., B. L. N. Kennett, H. Igel, and H. P. Bunge (2009), Full seismic waveform tomography for upper mantle structure in the australasian region using adjoint methods, *Geophys. J. Int.*, *179*, 1703–1725.

Forsyth, D., and S. Uyeda (1975), On the relative importance of the driving forces of plate motion, *Geophys. J. Int.*, *43*, 163–200.

Fowler, C. M. R. (2005), *The Solid Earth: An Introduction to Global Geophysics*, Cambridge University Press.

French, S., V. Lekic, and B. Romanowicz (2013), Waveform tomography reveals channeled flow at the base of the oceanic asthenosphere, *Science*, *342*, 227–230.

Fuchs, K., E. A. Kozlovsky, A. I. Krivtsov, and M. D. Zoback (Eds.) (1990), *Super-Deep Continental Drilling and Deep Geophysical Sounding*, Springer Berlin.

Fukao, Y., M. Obayashi, and T. Nakakuki (2009), Stagnant slab: A review, in *Annu. Rev. Earth Planet. Sci.*, vol. 37, pp. 19–46.

Gerya, T. V., and D. A. Yuen (2007), Robust characteristics method for modelling multiphase visco-elasto-plastic thermo-mechanical problems, *Phys. Earth Planet. Inter.*, *163*, 83–105.

Glatzmaier, G. A. (1988), Numerical simulation of mantle convection: time-dependent, three-dimensional, compressible, spherical shell, *Geophys. Astrophys. Fluid Dyn.*, *43*, 223–264.

Gmeiner, B. (2013), Design and analysis of hierarchical hybrid multigrid methods for peta-scale systems and beyond, Ph.D. thesis, Department of Computer Science, Friedrich-Alexander Universität Erlangen.

Gmeiner, B., and U. Rüde (2014), Peta-scale hierarchical hybrid multigrid using hybrid parallelization, in *Large-Scale Scientific Computing*, *Lecture Notes in Computer Science*, vol. 8353, pp. 439–447, Springer.

Gmeiner, B., U. Rüde, H. Stengel, C. Waluga, and B. Wohlmuth (2014a), Performance and scalability of hierarchical hybrid multigrid solvers for Stokes systems, *SIAM J. Sci. Comput.*, *37*, C143–C168.

Gmeiner, B., C. Waluga, and B. Wohlmuth. (2014b), Local mass-corrections for continuous pressure approximations of incompressible flow, *SIAM J. Numer. Anal.*, *52*, 2931–2956.

Gmeiner, B., U. Rüde, H. Stengel, C. Waluga, and B. Wohlmuth (2015), Towards textbook efficiency for parallel multigrid, *Numer. Math. Theory Me.*, *8*, 22–46.

GRACE (2002), Static field geopotential coefficients estimated from satellite data only, information system and data center, *Tech. rep.*, GFZ German Research Centre for Geosciences, Germany.

Gradl, T., and U. Rüde (2008), High performance multigrid in current large scale parallel computers, in *9th Workshop on Parallel Systems and Algorithms (PASA)*, vol. 124, pp. 37–45.

Gradl, T., C. Freundl, H. Köstler, and U. Rüde (2008), Scalable multigrid, in *High Performance Computing in Science and Engineering*, edited by S. Wagner, M. Steinmetz, A. Bode, and M. Brehm, pp. 475–483, LRZ, KON-WIHR, Springer Berlin Heidelberg New York.

Grand, S. P., R. D. van der Hilst, and S. Widiyantoro (1997), Global seismic tomography: A snapshot of convection in the Earth, *GSA Today*, *7*, 1–7.

Griffiths, D., and D. Silvester (2011), Unstable modes of the q1-p0 element, *Tech. rep.*, University of Manchester.

Gunzburger, M. D. (1989), *Finite element methods for viscous incompressible flows*, Academic Press, INC.

Gurnis, M., and G. F. Davies (1986), Numerical study of high Rayleigh number convection in a medium with depth-dependent viscosity, *Geophys. J. Roy. Astr. S.*, *186*, 523–541.

Hager, B. H., and R. J. O'Connell (1981), A simple global model of plate dynamics and mantle convection, *J. Geophys. Res.*, *86*, 4843–4867.

Hager, B. H., and M. A. Richards (1989), Long-wavelength variations in earths geoid: physical models and dynamical implications, *Philos. T. Roy. Soc. A*, *328*, 309–327.

Haq, B. U., J. Hardenbol, and P. R. Vail (1998), Mesozoic and cenozoic chronostratigraphy and cycles of sea-level change.

Hartley, R. A., G. G. Roberts, N. White, and C. Richardson (2011), Transient convective uplift of an ancient buried landscape, *Nature Geosci.*, *4*, 562–565.

Haskell, N. A. (1935), The motion of a fluid under a surface load, *Physics*, *6*, 265–269.

Hirth, G., and D. Kohlstedt (2003), *Rheology of the Upper Mantle and the Mantle Wedge: A View from the Experimentalists. In: Inside the Subduction Factory*, pp. 83–105, American Geophysical Union.

Höink, T., and A. Lenardic (2008), Three-dimensional mantle convection simulations with a low-viscosity asthenosphere and the relationship between heat flow and the horizontal length scale of convection, *Geophys. Res. Lett.*, *35*, L10304.

Höink, T., and A. Lenardic (2010), Long wavelength convection, poiseuillecouette ow in the low-viscosity asthenosphere and the strength of plate margins, *Geophys. J. Int.*, *180*, 23–33.

Höink, T., A. M. Jellinek, and A. Lenardic (2011), Viscous coupling at the lithosphereasthenosphere boundary, *Geochem. Geophy. Geosy.*, *12*, Q0AK02.

Höink, T., A. Lenardic, and M. Richards (2012), Depth-dependent viscosity and mantle stress amplification: implications for the role of the asthenosphere in maintaining plate tectonics, *Geophys. J. Int.*, *191*, 30–41.

Horbach, A., H. P. Bunge, and J. Oeser (2014), The adjoint method in geodynamics: derivation from a general operator formulation and application to the initial condition problem in a high resolution mantle circulation model, *GEM Int. J. Geomath.*, *5*, 163–194.

Houseman, G. A., D. P. McKenzie, and P. Molnar (1981), Convective instability of a thickened boundary layer and its relevance for the thermal evolution of continental convergent belts, *J. Geophys. Res. Sol. Ea.*, *86*, 6115–6132.

Huber, M., L. John, B. Gmeiner, U. Rüde, and B. Wohlmuth (2015), Massively parallel hybrid multigrid solvers for the Stokes system, *In preparation.*

Hughes, T. J. R., L. P. Franca, and M. Balestra (1986), A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuska-Brezzi condition: a stable Petrov-Galerkin formulation of the stokes problem accommodating equal-order interpolations, *Comput. Method. Appl. M.*, *59*, 85–99.

Hülsemann, F., B. Bergen, and Rüde (2003), Hierarchical hybrid grids as basis for parallel numerical solution of PDE, in *Euro-Par 2003 Parallel Processing*, pp. 840–843, Springer.

Iaffaldano, G., and H.-P. Bunge (2015), Rapid plate motion variations through geological time: Observations serving geodynamic interpretation, *Annu. Rev. Earth Planet. Sci.*, *43*, 19.1–19.22.

Irifune, T., and T. Tsuchiya (2007), Mineralogy of the Earth - phase transitions and mineralogy of the lower mantle.

ISO1539 (1991), Fortran 90, *Tech. Rep. ISO/IEC 1539 : 1991 (E)*, International Organization for Standardization, Geneva, Switzerland.

Ita, J., and S. D. King (1994), Sensitivity of convection with an endothermic phase change to the form of governing equations, initial conditions, boundary conditions, and equation of state, *J. Geophys. Res.*, *99*, 15,919–15,938.

Japsen, P., J. A. Chalmers, P. F. Green, and J. M. Bonow (2012a), Elevated, passive continental margins: Not rift shoulders, but expressions of episodic, post-rift burial and exhumation, *Global Planet. Change*, *90–91*, 73–86.

Japsen, P., J. M. Bonow, P. F. Green, P. R. Cobbold, D. Chiossi, R. Lilletveit, L. P. Magnavita, and A. Pedreira (2012b), Episodic burial and exhumation in NE Brazil after opening of the South Atlantic, *Geol. Soc. Am. Bull.*, *124*, 800–816.

Jarvis, G. T., and D. P. McKenzie (1980), Convection in a compressible fluid with infinite Prandtl number, *J. Fluid Mech.*, *96*, 515–583.

Jones, A. G. (1982), Observations of the electrical asthenosphere beneath scandinavia, *Tectonophysics*, *90*, 37–55.

Karato, S., and H. Jung (1998), Water, partial melting and the origin of seismic low velocity and high attenuation zone in the upper mantle, *Earth Planet. Sci. Lett.*, *157*, 193–207.

Karato, S. I., and P. Wu (1993), Rheology of the upper mantle: A synthesis, *Science*, *260*, 771–778.

King, S. D., C. Lee, P. E. van Keken, W. Leng, S. Zhong, E. Tan, N. Tosi, and M. C. Kameyama (2010), A community benchmark for 2-d cartesian compressible convection in the Earths mantle, *Geophys. J. Int.*, *180*, 73–87.

Korenaga, J. (2011), Earth's heat budget: Clairvoyant geoneutrinos, *Nat. Geosci.*, *4*, 581–582.

Kronbichler, M., T. Heister, and W. Bangerth (2012), High accuracy mantle convection simulation through modern numerical methods, *Geophys. J. Int.*, *191*, 12–29.

Kuzmin, A., M. Luisier, and O. Schenk (2013), Fast methods for computing selected elements of the green's function in massively parallel nanoelectronic device simulations, in *Euro-Par 2013, LNCS 8097*, edited by F. Wolf, B. Mohr, and D. an Ney, pp. 533–544, Springer-Verlag Berlin Heidelberg.

Landau, L. D., and E. M. Lifshitz (1959), *Fluid Mechanics, Course of Theoretical Physics*, vol. 6, 2nd ed., Butterworth-Heinemann.

Larson, K. M., J. T. Freymueller, and S. Philipsen (1997), Global plate velocities from the global positioning system, *J. Geophys. Res. Sol. Ea.*, *102*, 9961–9981.

Lay, T., J. Hernlund, and B. A. Buffett (2008), Core-mantle boundary heat flow, *Nat. Geosci.*, *1*, 25–32.

Lenardic, A., M. A. Richards, and F. H. Busse (2006), Depth-dependent rheology and the horizontal length scale of mantle convection, *J. Geophys. Res.*, *111*, B07404.

Leng, W., and S. Zhong (2008), Viscous heating, adiabatic heating and energetic consistency in compressible mantle convection, *Geophys. J. Int.*, *173*, 693–702.

Lithgow-Bertelloni, C., and P. G. Silver (1998), Dynamic topography, plate driving forces and the African superswell, *Nature*, *395*, 269–272.

Liu, L., and M. Gurnis (2008), Simultaneous inversion of mantle properties and initial conditions using an adjoint of mantle convection, *J. Geophys. Res.*, *113*, B08405.

McElhinny, M. W., and J. Lock (1996), IAGA paleomagnetic databases with access, *Surv. Geophys*, *17*, 575–591.

McKenzie, D. P., J. M. Roberts, and N. O. Weiss (1974), Convection in the Earth's mantle: towards a numerical simulation, *J. Fluid Mech.*, *62*, 465–538.

Mitrovica, J. X. (1996), Haskell [1935] revisited, *J. Geophys. Res.*, *101*, 555–569.

Müller, R. D., M. Sdrolias, C. Gaina, and W. R. Roest (2008), Age, spreading rates, and spreading asymmetry of the world's ocean crust, *Geochem. Geophy. Geosy.*, *9*, 1525–2027.

Murnaghan, F. D. (1944), The compressibility of media under extreme pressures, *Proc. N. A. S.*, *30*, 244–247.

Nakagawa, T., P. J. Tackley, and J. A. D. Connolly (2009), Incorporating self-consistently calculated mineral physics into thermochemical mantle convection simulations in a 3-d spherical shell and its influence on seismic anomalies in earths mantle, *Geochem. Geophy. Geosy.*, *10*, 1525–2027.

netlib.org (2015), LAPACK – Linear Algebra PACKage, http://www.netlib.org/lapack/, accessed February 23, 2015.

Parnell-Turner, R., N. White, T. Henstock, B. Murton, J. Maclennan, and S. M. Jones (2014), A continuous 55 million year record of transient mantle plume activity beneath Iceland, *Nature Geosci.*, *7*, 914–919.

Paulson, A., and M. A. Richards (2009), On the resolution of radial viscosity structure in modelling long-wavelength postglacial rebound data, *Geophys. J. Int.*, *179*, 1516–1526.

Piazzoni, A., G. Steinle-Neumann, H. P. Bunge, and D. Dolejš (2007), A mineralogical model for density and elasticity of the Earth's mantle, *Geochem. Geophy. Geosy.*, *8*, 1–23.

Popov, A. A., and S. Sobolev (2008), Slim3d: A tool for three-dimensional thermomechanical modeling of lithospheric deformation with elasto-visco-plastic rheology, *Phys. Earth Planet. Inter.*, *171*, 55–75.

Ramage, A., and A. J. Wathen (1994), Iterative solution techniques for the Stokes and Navier-Stokes equations, *Int. J. Numer. Methods Fluids*, *19*, 67–83.

Ribe, N. M. (2007), Analytical approaches to mantle dynamics, in *Treatise on Geophysics*, vol. 7, pp. 167–226, Elsevier.

Ricard, Y. (2007), Physics of mantle convection, in *Treatise on Geophysics*, vol. 7, pp. 31–87, Elsevier.

Richardson, . M., and S. C. Solomon (1976), Intraplate stress as an indicator of plate tectonic driving forces, *J. Geophys. Res.*, *81*, 1847–1856.

Richter, F. M. (1973), Convection and the large-scale circulation of the mantle, *J. Geophys. Res.*, *78*, 8735–8745.

Ritsema, J., A. Deuss, H. J. van Heijst, and J. H. Woodhouse (2011), S40RTS: a degree-40 shear-velocity model for the mantle from new Rayleigh wave dispersion, teleseismic traveltime and normal-mode splitting function measurements, *Geophys. J. Int.*, *184*, 1223–1236.

Schaber, K., H. P. Bunge, B. S. A. Schuberth, R. Malservisi, and A. Horbach (2009), Stability of the rotation axis in high-resolution mantle circulation models: Weak polar wander despite strong core heating, *Geochem. Geophy. Geosy.*, *10*, Q11W04.

Schenk, O., and K. Gärtner (2014), *Parallel Sparse Direct And Multi-Recursive Iterative Linear Solvers. PARDISO. User Guide Version 5.0.0*, http://www.pardiso-project.org/manual/manual.pdf.

Schubert, G., D. A. Yuen, and D. L. Turcotte (1975), Role of phase transitions in a dynamic mantle, *Geophys. J. R. astr. Soc.*, *42*, 705–735.

Schubert, G., D. L. Turcott, and P. Olson (2001), *Mantle Convection in the Earth and Planets*, Cambridge University Press, New York.

Schuberth, B. S. A., H. P. Bunge, G. Steinle-Neumann, C. Moder, and J. Oeser (2009), Thermal versus elastic heterogeneity in high-resolution mantle circulation models with pyrolite composition: High plume excess temperatures in the lowermost mantle, *Geochem. Geophy. Geosy.*, *10*, Q01W01.

Solomatov, V. S. (1995), Scaling of temperature and stress dependent viscosity convection, *Phys. Fluids*, *7*, 266–274.

Stegman, D. R., M. A. Richards, and J. R. Baumgardner (2002), Effects of depth-dependent viscosity and plate motions on maintaining a relatively uniform mid-ocean ridge basalt reservoir in whole mantle flow, *J. Geophys. Res.*, *107*, ETG-5.

Steinbach, V., U. Hansen, and A. Ebel (1989), Compressible convection in the earths mantle: a comparison of different approaches, *Geophys. Res. Lett.*, *16*, 633–636.

Steinberger, B., and T. H. Torsvik (2012), A geodynamic model of plumes from the margins of large low shear velocity provinces, *Geochem. Geophy. Geosy.*, *13*, 1525–2027.

Stixrude, L., and C. Lithgow-Bertelloni (2005), Thermodynamics of mantle minerals – I. physical properties, *Geophys. J. Int.*, *162*, 610–632.

Stixrude, L., and C. Lithgow-Bertelloni (2011), Thermodynamics of mantle minerals – II. phase equilibria, *Geophys. J. Int.*, *184*, 1180–1213.

Strohmaier, E., J. Dongarra, H. Simon, and M. Meuer (2015), Top 500 list, http://www.top500.org/list/2015/06/.

Stuhne, G. R., and W. R. Peltier (1996), Vortex erosion and amalgamation in a new model of large scale flow on the sphere, *J. Comput. Phys.*, *128*, 58–81.

Tackley, P. J. (1996), Effects of strongly variable viscosity on three-dimensional compressible convection in planetary mantles, *J. Geophys. Res.*, *101*, 3311–3332.

Tackley, P. J. (2000), Self-consistent generation of tectonic plates in time-dependent, three-dimensional mantle convection simulations, *Geochem. Geophy. Geosy.*, *1*, 1021.

Tackley, P. J. (2012), Dynamics and evolution of the deep mantle resulting from thermal, chemical, phase and melting effects, *Earth-Sci. Rev.*, *110*, 1–25.

Tackley, P. J., D. J. Stevenson, G. A. Glatzmaier, and G. Schubert (1993), Effects of endothermic phase transitions at 670 km depth in a spherical model of convection in the earth's mantle, *Nature*, *361*, 699–704.

Trampert, J., F. Deschamps, J. Resovsky, and D. Yuen (2004), Probabilistic tomography maps chemical heterogeneities throughout the lower mantle, *Science*, *306*, 853–856.

Turcotte, D. L., and E. R. Oxburgh (1967), Finite amplitude convective cells and continental drift, *J. Fluid Mech.*, *28*, 29–42.

Verfürth, R. (1984), A combined conjugate gradient-multigrid algorithm for the numerical solution of the Stokes problem, *IMA J. Numer. Anal.*, *4*, 441–455.

Vogt, P. R. (1971), Asthenosphere motion recorded by the ocean floor south of iceland, *Earth Planet. Sci. Lett.*, *13*, 153–160.

Vynnytska, L., and H. P. Bunge (2014), Restoring past mantle convection structure through fluid dynamic inverse theory: regularisation through surface velocity boundary conditions, *GEM Int. J. Geomath.*, *6*, 83–100.

Waluga, C., B. Wohlmuth, and U. Rüde (2015), Mass-corrections for the conservative coupling of flow and transport on collocated meshes, *Tech. rep.*, Technische Universität München.

Weismüller, J., B. Gmeiner, S. Ghelichkhan, M. Huber, L. John, B. Wohlmuth, U. Rüde, and H. P. Bunge (2015), Fast asthenosphere motion in high-resolution global mantle flow models, *Geophys. Res. Lett.*, 2015GL063727.

Wiens, D. A., and S. Stein (1985), Implications of oceanic intraplate seismicity for plate stresses, driving forces and rheology, *Tectonophysics*, *116*, 143–162.

Winterbourne, J., A. Crosby, and N. White (2009), Depth, age and dynamic topography of oceanic lithosphere beneath heavily sedimented atlantic margins, *Earth Planet. Sci. Lett.*, *287*, 137–151.

Workman, R. K., and S. R. Hart (2005), Major and trace element composition of the depleted MORB mantle, *Earth Planet. Sci. Lett.*, *231*, 53–72.

Yang, W. S. (1997), Variable viscosity thermal convection at infinite Prandtl number in a thick spherical shell, Ph.D. thesis, University of Illinois.

Zhang, S., and D. A. Yuen (1996), Various influences on plumes and dynamics in time-dependent, compressible mantle convection in 3-d spherical shell, *Phys. Earth Planet. Inter.*, *94*, 241–267.

Zhong, S., N. Zhang, Z. X. Li, and J. H. Roberts (2007a), Supercontinent cycles, true polar wander, and very long-wavelength mantle convection, *Earth Planet. Sci. Lett.*, *261*, 551–564.

Zhong, S. J., D. A. Yuen, and L. N. Moresi (2007b), Numerical methods for mantle convection, in *Treatise on Geophysics*, vol. 7, pp. 227–252, Elsevier.

Zienkiewicz, O. C., R. L. Taylor, and J. Z. Zhu (2005), *The Finite Element Method: Its Basis and Fundamentals*, Butterworth-Heinemann.

Acknowledgements

First of all I would like to thank my supervisor, Hans-Peter Bunge for giving me the opportunity to write this thesis, his constant encouragement and many valuable discussions. Many thanks also to Marcus Mohr for co-supervising me, especially for his constant help with the mathematical and computational aspects of my work. A lot of thanks also to my committe, to Ulrich Rüde, Anke Friedrich, Heiner Igel, Ralf Ludwig and Claudia Trepmann for their dedication.

Many thanks to the geodynamics group, to Simon Bauer, Hans-Peter Bunge, Thomas Chust, Lorenzo Colli, Sia Ghelichkhan, Andre Horbach, Christoph Moder, Marcus Mohr, Rainer Nerlich and Bernhard Schuberth, for many fruitful discussions and their support throughout my time at the geophysics intitute.

A lot of thanks also to Björn Gmeiner, Lorenz John, Markus Huber, Ulrich Rüde, Christian Waluga, Barbara Wohlmuth and the whole Terra-neo team for the good collaboration and their support with the HHG-related parts of my work.

Thanks to Yang Li and Stephan Eder for the discussions and the nice atmosphere at the office during the first year of my thesis.

Finally I would like to thank Petra Effenberger and my family for their support and help during this time.

This work was supported in parts by the German Research Foundation (DFG) within the priority programs SPP 1648-SPPEXA and SPP 1375-SAMPLE. Computing resources were provided by the Gauss Center for Supercomputing (GCS) on the supercomputers JUQUEEN and Super-MUC, funded by the German Federal Ministry of Education and Research (BMBF) and the German State Ministries for Research of Baden-Württemberg (MWK), Bayern (StMWFK) and Nordrhein-Westfalen (MIWF).

Appendix

A Code Example for a Projected Coordinate Smoother in HHG

Listing A.1: On-the-fly stencil assembly and application for projected coordinates and constant coefficients.

```
// Gauss-Seidel smoother for constant coefficients and projected coordinates on a tetrahedron

void tet_gs_coords(std::vector<double*> &u, std::vector<double*> &f, std::vector<double*> &X,
                   std::vector<hhgTetrahedronOperator<double>*> oprTypes, int tsize) {

   const int tet_mcq =  0,
   const int tet_me  =  1;
   const int tet_mnw =  2;
   const int tet_mn  =  3;
   const int tet_ts  =  4;
   const int tet_tse =  5;
   const int tet_tw  =  6;
   const int tet_tc  =  7;
   const int tet_bc  =  8;
   const int tet_be  =  9;
   const int tet_bnw = 10;
   const int tet_bn  = 11;
   const int tet_ms  = 12;
   const int tet_mse = 13;
   const int tet_mw  = 14;
   const int tet_mc  = 15;

   int mp, tp, bp;
   int mp_mr, mp_tr, mp_br;
   int tp_mr, tp_br;
   int bp_mr, bp_tr;

   int p0, p1, p2, p3;

   bp = 0;
   mp = PLANESIZE(tsize);
   tp = mp + PLANESIZE(tsize - 1);

   // The 15-point stencils for all operators and the local element matrix
   std::vector<double*> stencil;
   double c[16];

    // Coordinates of the four corners of the fine grid tetrahedron
   double coords[4*3];
```

```
// We store 16 stencil entries, one of them is the inverse of the central element
for (int op = 0; op < oprTypes.size(); ++op) {
  stencil.push_back(new double[16]);
}

// Main loops over all inner points within the coarse grid tetrahedron
for (int k = 1; k < (tsize - 3); ++k) {

   bp_mr = bp + tsize - k + 1;
   bp_tr = bp + 2 * (tsize - k + 1) - 1;

   mp_br = mp;
   mp_mr = mp + tsize - k;
   mp_tr = mp + 2 * (tsize - k) - 1;

   tp_br = tp;
   tp_mr = tp + tsize - k - 1;

   for (int j = 1; j < (tsize - k - 2); ++j) {

      for (int i = 1; i < (tsize - j - k - 1); i = i + 1) {

         // Loop over all operators we want to apply
         for (int op = 0; op < oprTypes.size(); ++op) {

           // Object that provides the method to calculate the local stiffness matrix
           hhgTetrahedronOperator<double>& localMatrix = *oprTypes[op];

           ////////////////////////////////////////
           // group zero element
           ////////////////////////////////////////

           // position zero
           p0 = TET_MC; p1 = TET_ME; p2 = TET_MN; p3 = TET_TC;
           coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
           coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
           coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
           coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
           // Calculate the local stiffness matrix and add the elements to the stencil
           localMatrix.matrix(c, coords);
           stencil[op][tet_mc] = c[0 + 0*4];
           stencil[op][tet_me] = c[0 + 1*4];
           stencil[op][tet_mn] = c[0 + 2*4];
           stencil[op][tet_tc] = c[0 + 3*4];

           // position one
           p0 = TET_MW; p1 = TET_MC; p2 = TET_MNW; p3 = TET_TW;
           coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
           coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
           coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
           coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
           // Calculate the local stiffness matrix and add the elements to the stencil
           localMatrix.matrix(c, coords);
           stencil[op][tet_mw] = c[1 + 0*4];
           stencil[op][tet_mc] += c[1 + 1*4];
           stencil[op][tet_mnw] = c[1 + 2*4];
           stencil[op][tet_tw] = c[1 + 3*4];
```

```
// position two
p0 = TET_MS; p1 = TET_MSE; p2 = TET_MC; p3 = TET_TS;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_ms] = c[2 + 0*4];
stencil[op][tet_mse] = c[2 + 1*4];
stencil[op][tet_mc] += c[2 + 2*4];
stencil[op][tet_ts] = c[2 + 3*4];

// position three
p0 = TET_BC; p1 = TET_BE; p2 = TET_BN; p3 = TET_MC;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_bc] = c[3 + 0*4];
stencil[op][tet_be] = c[3 + 1*4];
stencil[op][tet_bn] = c[3 + 2*4];
stencil[op][tet_mc] += c[3 + 3*4];

////////////////////////////////////////
// group one element
////////////////////////////////////////

// position zero
p0 = TET_MC; p1 = TET_TS; p2 = TET_TW; p3 = TET_TC;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_mc] += c[0 + 0*4];
stencil[op][tet_ts] += c[0 + 1*4];
stencil[op][tet_tw] += c[0 + 2*4];
stencil[op][tet_tc] += c[0 + 3*4];

// position one
p0 = TET_BN; p1 = TET_MC; p2 = TET_MNW; p3 = TET_MN;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_bn] += c[1 + 0*4];
stencil[op][tet_mc] += c[1 + 1*4];
stencil[op][tet_mnw] += c[1 + 2*4];
stencil[op][tet_mn] += c[1 + 3*4];
```

```
// position two
p0 = TET_BE; p1 = TET_MSE; p2 = TET_MC; p3 = TET_ME;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_be] += c[2 + 0*4];
stencil[op][tet_mse] += c[2 + 1*4];
stencil[op][tet_mc] += c[2 + 2*4];
stencil[op][tet_me] += c[2 + 3*4];

// position three
p0 = TET_BC; p1 = TET_MS; p2 = TET_MW; p3 = TET_MC;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_bc] += c[3 + 0*4];
stencil[op][tet_ms] += c[3 + 1*4];
stencil[op][tet_mw] += c[3 + 2*4];
stencil[op][tet_mc] += c[3 + 3*4];

////////////////////////////////////////
// group two element
////////////////////////////////////////

// position zero
p0 = TET_MC; p1 = TET_MNW; p2 = TET_TW; p3 = TET_TC;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_mc] += c[0 + 0*4];
stencil[op][tet_mnw] += c[0 + 1*4];
stencil[op][tet_tw] += c[0 + 2*4];
stencil[op][tet_tc] += c[0 + 3*4];

// position one
p0 = TET_MSE; p1 = TET_MC; p2 = TET_TS; p3 = TET_TSE;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_mse] += c[1 + 0*4];
stencil[op][tet_mc] += c[1 + 1*4];
stencil[op][tet_ts] += c[1 + 2*4];
stencil[op][tet_tse] = c[1 + 3*4];

// position two
```

```
p0 = TET_BE; p1 = TET_BN; p2 = TET_MC; p3 = TET_ME;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_be] += c[2 + 0*4];
stencil[op][tet_bn] += c[2 + 1*4];
stencil[op][tet_mc] += c[2 + 2*4];
stencil[op][tet_me] += c[2 + 3*4];

// position three
p0 = TET_BC; p1 = TET_BNW; p2 = TET_MW; p3 = TET_MC;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_bc] += c[3 + 0*4];
stencil[op][tet_bnw] = c[3 + 1*4];
stencil[op][tet_mw] += c[3 + 2*4];
stencil[op][tet_mc] += c[3 + 3*4];

////////////////////////////////////////
// group three element
////////////////////////////////////////

// position zero
p0 = TET_MC; p1 = TET_ME; p2 = TET_TSE; p3 = TET_TC;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_mc] += c[0 + 0*4];
stencil[op][tet_me] += c[0 + 1*4];
stencil[op][tet_tse] += c[0 + 2*4];
stencil[op][tet_tc] += c[0 + 3*4];

// position one
p0 = TET_MW; p1 = TET_MC; p2 = TET_TS; p3 = TET_TW;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_mw] += c[1 + 0*4];
stencil[op][tet_mc] += c[1 + 1*4];
stencil[op][tet_ts] += c[1 + 2*4];
stencil[op][tet_tw] += c[1 + 3*4];

// position two
p0 = TET_BNW; p1 = TET_BN; p2 = TET_MC; p3 = TET_MNW;
```

```
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_bnw] += c[2 + 0*4];
stencil[op][tet_bn] += c[2 + 1*4];
stencil[op][tet_mc] += c[2 + 2*4];
stencil[op][tet_mnw] += c[2 + 3*4];

// position three
p0 = TET_BC; p1 = TET_BE; p2 = TET_MSE; p3 = TET_MC;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_bc] += c[3 + 0*4];
stencil[op][tet_be] += c[3 + 1*4];
stencil[op][tet_mse] += c[3 + 2*4];
stencil[op][tet_mc] += c[3 + 3*4];

////////////////////////////////////////
// group four element
////////////////////////////////////////

// position zero
p0 = TET_MC; p1 = TET_TS; p2 = TET_TSE; p3 = TET_TC;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_mc] += c[0 + 0*4];
stencil[op][tet_ts] += c[0 + 1*4];
stencil[op][tet_tse] += c[0 + 2*4];
stencil[op][tet_tc] += c[0 + 3*4];

// position one
p0 = TET_BN; p1 = TET_MC; p2 = TET_ME; p3 = TET_MN;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_bn] += c[1 + 0*4];
stencil[op][tet_mc] += c[1 + 1*4];
stencil[op][tet_me] += c[1 + 2*4];
stencil[op][tet_mn] += c[1 + 3*4];

// position two
p0 = TET_BNW; p1 = TET_MW; p2 = TET_MC; p3 = TET_MNW;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
```

```
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_bnw] += c[2 + 0*4];
stencil[op][tet_mw] += c[2 + 1*4];
stencil[op][tet_mc] += c[2 + 2*4];
stencil[op][tet_mnw] += c[2 + 3*4];

// position three
p0 = TET_BC; p1 = TET_MS; p2 = TET_MSE; p3 = TET_MC;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_bc] += c[3 + 0*4];
stencil[op][tet_ms] += c[3 + 1*4];
stencil[op][tet_mse] += c[3 + 2*4];
stencil[op][tet_mc] += c[3 + 3*4];

////////////////////////////////////////
// group five element
////////////////////////////////////////

// position zero
p0 = TET_MC; p1 = TET_MNW; p2 = TET_MN; p3 = TET_TC;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_mc] += c[0 + 0*4];
stencil[op][tet_mnw] += c[0 + 1*4];
stencil[op][tet_mn] += c[0 + 2*4];
stencil[op][tet_tc] += c[0 + 3*4];

// position one
p0 = TET_MSE; p1 = TET_MC; p2 = TET_ME; p3 = TET_TSE;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
// Calculate the local stiffness matrix and add the elements to the stencil
localMatrix.matrix(c, coords);
stencil[op][tet_mse] += c[1 + 0*4];
stencil[op][tet_mc] += c[1 + 1*4];
stencil[op][tet_me] += c[1 + 2*4];
stencil[op][tet_tse] += c[1 + 3*4];

// position two
p0 = TET_MS; p1 = TET_MW; p2 = TET_MC; p3 = TET_TS;
coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
```

```
  coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
  coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
  // Calculate the local stiffness matrix and add the elements to the stencil
  localMatrix.matrix(c, coords);
  stencil[op][tet_ms] += c[2 + 0*4];
  stencil[op][tet_mw] += c[2 + 1*4];
  stencil[op][tet_mc] += c[2 + 2*4];
  stencil[op][tet_ts] += c[2 + 3*4];

  // position three
  p0 = TET_BC; p1 = TET_BNW; p2 = TET_BN; p3 = TET_MC;
  coords[0] = X[0][p0]; coords[ 1] = X[1][p0]; coords[ 2] = X[2][p0];
  coords[3] = X[0][p1]; coords[ 4] = X[1][p1]; coords[ 5] = X[2][p1];
  coords[6] = X[0][p2]; coords[ 7] = X[1][p2]; coords[ 8] = X[2][p2];
  coords[9] = X[0][p3]; coords[10] = X[1][p3]; coords[11] = X[2][p3];
  // Calculate the local stiffness matrix and add the elements to the stencil
  localMatrix.matrix(c, coords);
  stencil[op][tet_bc] += c[3 + 0*4];
  stencil[op][tet_bnw] += c[3 + 1*4];
  stencil[op][tet_bn] += c[3 + 2*4];
  stencil[op][tet_mc] += c[3 + 3*4];
}

// Do we have a vector-valued problem?
if (f.size() > 1) {
  double tf[3] = { f[0][mp_mr + i], f[1][mp_mr + i], f[2][mp_mr + i] };
  int opm[3][3] = { { 0, 1, 2 }, { 3, 4, 5 }, { 6, 7, 8 } };

  // We have three components on our unknown vector and on our right hand side.
  for (int opi = 0; opi < 3; ++opi) {
    for (int com = 0; com < 3; ++com) {

    int op = opm[com][opi];

    tf[com] = tf[com] - stencil[op][1] * u[opi][mp_mr + i + 1]
                      - stencil[op][2] * u[opi][mp_tr + i - 1]
                      - stencil[op][3] * u[opi][mp_tr + i]
                      - stencil[op][4] * u[opi][tp_br + i]
                      - stencil[op][5] * u[opi][tp_br + i + 1]
                      - stencil[op][6] * u[opi][tp_mr + i - 1]
                      - stencil[op][7] * u[opi][tp_mr + i]
                      - stencil[op][8] * u[opi][bp_mr + i]
                      - stencil[op][9] * u[opi][bp_mr + i + 1]
                      - stencil[op][10] * u[opi][bp_tr + i - 1]
                      - stencil[op][11] * u[opi][bp_tr + i]
                      - stencil[op][12] * u[opi][mp_br + i]
                      - stencil[op][13] * u[opi][mp_br + i + 1]
                      - stencil[op][14] * u[opi][mp_mr + i - 1];
    }
  }

  double invdet = 1.0 / (stencil[0][15] * stencil[4][15] * stencil[8][15]
                       - stencil[0][15] * stencil[5][15] * stencil[5][15]
                       - stencil[1][15] * stencil[1][15] * stencil[8][15]
                  + 2 * stencil[1][15] * stencil[2][15] * stencil[5][15]
                       - stencil[2][15] * stencil[2][15] * stencil[4][15]);
```

```
        double b11 = ( stencil[4][15] * stencil[8][15]
                     - stencil[5][15] * stencil[5][15]) * invdet;
        double b12 = (-stencil[1][15] * stencil[8][15]
                     + stencil[2][15] * stencil[5][15]) * invdet;
        double b13 = ( stencil[1][15] * stencil[5][15]
                     - stencil[2][15] * stencil[4][15]) * invdet;
        double b22 = ( stencil[0][15] * stencil[8][15]
                     - stencil[2][15] * stencil[2][15]) * invdet;
        double b23 = (-stencil[0][15] * stencil[5][15]
                     + stencil[1][15] * stencil[2][15]) * invdet;
        double b33 = ( stencil[0][15] * stencil[4][15]
                     - stencil[1][15] * stencil[1][15]) * invdet;

        u[0][mp_mr + i] = (b11 * tf[0] + b12 * tf[1] + b13 * tf[2]);
        u[1][mp_mr + i] = (b12 * tf[0] + b22 * tf[1] + b23 * tf[2]);
        u[2][mp_mr + i] = (b13 * tf[0] + b23 * tf[1] + b33 * tf[2]);

      } else {

        // We have a scalar problem

        // compute center weight quotient
        stencil[0][tet_mcq] = 1.0 / stencil[0][tet_mc];

        int comp = 0;
        int op = 0;
        u[comp][mp_mr + i] = stencil[op][0] * (f[comp][mp_mr + i]
                           - stencil[op][1] * u[comp][mp_mr + i + 1]
                           - stencil[op][2] * u[comp][mp_tr + i - 1]
                           - stencil[op][3] * u[comp][mp_tr + i]
                           - stencil[op][4] * u[comp][tp_br + i]
                           - stencil[op][5] * u[comp][tp_br + i + 1]
                           - stencil[op][6] * u[comp][tp_mr + i - 1]
                           - stencil[op][7] * u[comp][tp_mr + i]
                           - stencil[op][8] * u[comp][bp_mr + i]
                           - stencil[op][9] * u[comp][bp_mr + i + 1]
                           - stencil[op][10] * u[comp][bp_tr + i - 1]
                           - stencil[op][11] * u[comp][bp_tr + i]
                           - stencil[op][12] * u[comp][mp_br + i]
                           - stencil[op][13] * u[comp][mp_br + i + 1]
                           - stencil[op][14] * u[comp][mp_mr + i - 1]);
      }
    }

    bp_mr = bp_tr;
    bp_tr = bp_tr + tsize - j - k;

    mp_br = mp_mr;
    mp_mr = mp_tr;
    mp_tr = mp_tr + tsize - j - k - 1;

    tp_br = tp_mr;
    tp_mr = tp_mr + tsize - j - k - 1;
  }

  bp = mp;
  mp = tp;
```

```
      tp = tp + PLANESIZE(tsize-k-1);
   }

}
```