The Dissertation Committee for Johann Rudi
certifies that this is the approved version of the following dissertation:

Global Convection in Earth's Mantle: Advanced Numerical Methods and Extreme-Scale Simulations

Committee:

Omar Ghattas, Supervisor

Georg Stadler, Co-Supervisor

Michael Gurnis

Kui Ren

George Biros

Marc Hesse

Global Convection in Earth's Mantle: Advanced Numerical Methods and Extreme-Scale Simulations

by

Johann Rudi

DISSERTATION

Presented to the Faculty of the Graduate School of
The University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

The University of Texas at Austin

August 2018

Dedicated to my dear parents, Eugenia and Peter Rudi.

Acknowledgments

Any researcher is, of course, involved in expanding individual knowledge, but all of us contribute to *the construction of the same house*—building a culture of scientific understanding. I could not have created this dissertation without the body of research before me and also not without the people that surrounded me. Therefore I am foremost grateful to my outstanding advisors and mentors, Omar Ghattas and Georg Stadler, and wonderful collaborators and colleagues who I met at the Institute for Computational Engineering and Sciences, especially Carsten Burstedde, Tobin Isaac, George Biros, Hari Sundar, Noemi Petra, Andreas Mang, and Umberto Villa. I thank Ivo Babuška for engaging discussions on multigrid. The motivation for my mathematical and computational research is flow in Earth's mantle and was driven by my collaborators and exceptional geophysicists from Caltech, Mike Gurnis, Vishagan Ratnaswamy, and Xi Liu. One of my research highlights was during the time leading up to the 2015 Gordon Bell Prize submission. A time marked by intensity and struggle that was rewarded with a milestone achievement, which was reached thanks to the dedication and commitment of Cristiano Malossi from IBM Research, Zurich together with Peter Staar, Yves Ineichen, Costas Bekas, and Alessandro Curioni. In addition, I thank Scott Futral (Lawrence Livermore National Laboratory) and Roy Musselman (IBM) for enabling access to the Sequoia 1.6M cores supercomputer. I also like to thank the people at the Texas Advanced Computing Center for providing excellent computational resources, education, and support.

It is my great pleasure to offer deep thanks to the remarkable staff members, Susan Rodriguez, who was of great help in countless situations, and Stephanie Rodriguez, simply the best coordinator a graduate program can wish for. Many thanks to my fellow students for their friendship and support, to wonderful, lovely friends in the Austin area outside of graduate school, and to a fantastic community of runners from Gilbert's Gazelles and Austin Fit, who make life fun and beautiful. Finally, thank you so very much to my family—Mom, Dad, grandparents, aunts, uncles, and cousins who believed in me through it all.

Abstract

Global Convection in Earth's Mantle: Advanced Numerical Methods and Extreme-Scale Simulations

Johann Rudi, Ph.D.
The University of Texas at Austin, 2018
Supervisors: Omar Ghattas and Georg Stadler

The thermal convection of rock in Earth's mantle and associated plate tectonics are modeled by nonlinear incompressible Stokes and energy equations. This dissertation focuses on the development of advanced, scalable linear and nonlinear solvers for numerical simulations of realistic instantaneous mantle flow, where we must overcome several computational challenges. The most notable challenges are the severe nonlinearity, heterogeneity, and anisotropy due to the mantle's rheology as well as a wide range of spatial scales and highly localized features. Resolving the crucial small scale features efficiently necessitates adaptive methods, while computational results greatly benefit from a high accuracy per degree of freedom and local mass conservation. Consequently, the discretization of Earth's mantle is carried out by high-order finite elements on aggressively adaptively refined hexahedral meshes with a continuous, nodal velocity approximation and a discontinuous, modal pressure approximation. These velocity–pressure pairings yield optimal asymptotic convergence rates of the finite element approximation to the infinite-dimensional solution with decreasing mesh element size, are inf-sup stable on general, non-conforming hexahedral meshes with "hanging nodes," and have the advantage of preserving mass locally at the element level due to the discontinuous pressure. However, because of the difficulties cited above and the desired accuracy, the large implicit systems to be solved are extremely poorly conditioned and sophisticated linear and nonlinear solvers including powerful preconditioning techniques are required.

The nonlinear Stokes system is solved using a grid continuation, inexact Newton–Krylov method. We measure the residual of the momentum equation in the H^{-1}-norm for backtracking line search to avoid overly conservative update steps that are significantly reduced from one. The Newton

linearization is augmented by a perturbation of a highly nonlinear term in mantle's rheology, resulting in dramatically improved nonlinear convergence.

We present a new Schur complement-based Stokes preconditioner, *weighted BFBT*, that exhibits robust fast convergence for Stokes problems with smooth but highly varying (up to 10 orders of magnitude) viscosities, optimal algorithmic scalability with respect to mesh refinement, and only a mild dependence on the polynomial order of high-order finite element discretizations. In addition, we derive theoretical eigenvalue bounds to prove spectral equivalence of our inverse Schur complement approximation.

Finally, we present a parallel *hybrid spectral–geometric–algebraic multigrid (HMG)* to approximate the inverses of the Stokes system's viscous block and variable-coefficient pressure Poisson operators within weighted BFBT. Building on the parallel scalability of HMG, our Stokes solver demonstrates excellent parallel scalability to 1.6 million CPU cores without sacrificing algorithmic optimality.

Contents

List of Figures

List of Tables

1

Introduction and Key Ideas

1.1 Motivation and challenges

Geophysical fluid flows constitute an important class of creeping flows of non-Newtonian fluids [54, 92]. The incompressible Stokes equations with power-law rheology have become a prototypical continuum mechanical description for creeping flows occurring in such applications as mantle convection [116], magma dynamics [86], and ice flow [68]. This dissertation in particular targets Earth's mantle convection coupled with plate tectonics at global scale [110]. We employ realistic flow models that exhibit extreme mathematical and computational challenges due to severe nonlinearities in the constitutive law and a highly heterogeneous viscosity stemming from its dependence on temperature and strain rate and from incorporating sharp gradients in narrow regions modeling tectonic plate boundaries.

Earth is a dynamic system in which mantle convection drives plate tectonics and continental drift and, in turn, controls much activity ranging from the occurrence of earthquakes and volcanoes to mountain building and long-term sea level change. Despite the central role of mantle convection in solid Earth dynamics, first-order knowledge gaps remain, with questions that are as basic as what are the principal driving and resisting forces on plate tectonics to what is the energy balance of the planet as a whole. Indeed, understanding mantle convection has been designated one of the *"10 Grand Research Questions in Earth Sciences"* in a National Academies report [37]. We seek to address such fundamental questions as: *(i)* What are the main drivers of plate motion—negative buoyancy forces or convective shear traction? *(ii)* What are key processes governing the occurrence of great earthquakes—the material properties between the plates or the tectonic stress? *(iii)* What role do subducting slab geometries play? *(iv)* How well can the mantle's rheology be extrapolated from laboratory experiments?

Addressing these questions requires, on the one hand, global models of Earth's mantle convection and associated plate tectonics, where interactions between localized phenomena and longer ranging flow play out on the whole sphere. On the other hand, high resolutions down to faulted plate boundaries are crucial. Historically, modeling at this scale and resolution simultaneously has been infeasible due to

the enormous computational complexity associated with numerical solution of the underlying mantle flow equations. However, with the advent of multi-petaflops supercomputers as well as significant advances in seismic tomography and space geodesy placing key observational constraints on mantle convection, we now have the opportunity to address these fundamental questions.

Solid rock in the mantle flows like a viscous fluid on time scales of millions of years [104]; therefore we model instantaneous mantle flow by the nonlinear incompressible Stokes equations:

$$-\nabla \cdot \left[\mu(\boldsymbol{u})\left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\mathsf{T}}\right)\right] + \nabla p = \boldsymbol{f}, \tag{1.1a}$$

$$-\nabla \cdot \boldsymbol{u} = 0, \tag{1.1b}$$

where $\boldsymbol{u}$ and p are the velocity and pressure fields, respectively; the right-hand side force, $\boldsymbol{f}$, is derived from the Boussinesq approximation; and the temperature- and strain rate-dependent effective viscosity, μ, is characterized by the constitutive law

$$\mu(\boldsymbol{u}) = \mu(T, \dot{\boldsymbol{\varepsilon}}(\boldsymbol{u})) = \mu_{\min} + \min\left(\frac{\tau_{\text{yield}}}{2\dot{\varepsilon}_{\text{II}}}, w\left(\boldsymbol{x}\right) \min\left(\mu_{\max}, a\left(T\right)\, \dot{\varepsilon}_{\text{II}}^{\frac{1-n}{n}}\right)\right). \tag{1.2}$$

The effective viscosity depends (nonlinearly) on a power $(1-n)/n$, $n \geq 1$, of the square root of the second invariant of the strain rate tensor, $\dot{\varepsilon}_{\text{II}} \coloneqq (\frac{1}{2}\,\dot{\boldsymbol{\varepsilon}} : \dot{\boldsymbol{\varepsilon}})^{1/2}$, where ":" represents the inner product of second-order tensors and $\dot{\boldsymbol{\varepsilon}} \coloneqq \frac{1}{2}(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\mathsf{T}})$. The viscosity decays exponentially with temperature via the Arrhenius relationship, symbolized by $a\left(T\right)$. The constitutive relation incorporates (nonlinear) plastic yielding with yield strength τ_{yield}, lower/upper bounds on viscosity $\mu_{\min}$ and $\mu_{\max}$, and a decoupling factor $w\left(\boldsymbol{x}\right)$ to model plate boundaries. For whole Earth mantle flow models, (1.1) are augmented with free-slip conditions (i.e., no tangential traction, no normal flow) at the core–mantle and top surface boundaries.

Solution of the nonlinear Stokes equations (1.1) entails solution of a sequence of linearized Stokes problems. The linearization alters the coefficient (1.2) and right-hand sides in (1.1). Discretization of velocity and pressure fields and the partial differential equations (PDE's) is carried out by high-order finite elements on (aggressively adaptively refined) hexahedral meshes with velocity–pressure pairings $\mathbb{Q}_k \times \mathbb{P}_{k-1}^{\text{disc}}$ of polynomial order $k \geq 2$ with a continuous, nodal velocity approximation $\mathbb{Q}_k$ and a discontinuous, modal pressure approximation $\mathbb{P}_{k-1}^{\text{disc}}$. These pairings yield optimal asymptotic convergence rates of the finite element approximation to the infinite-dimensional solution with decreasing mesh element size, are inf-sup stable on general, non-conforming hexahedral meshes with "hanging nodes," and have the advantage of preserving mass locally at the element level due to the discontinuous pressure [46, 62, 111]. While these properties have been recognized to be important for geophysics applications (e.g., see [83, 84]), the high-order discretization, adaptivity, and discontinuous pressure approximation present significant additional difficulties for iterative solvers (relative to low order, uniform grid, continuous discretizations). Our locally adaptively refined hexahedral meshes generate extreme local refinement—critical to resolve plate boundaries down to a few hundred meters—while simultaneously permitting significantly coarser meshes away from these regions to efficiently capture global-scale behavior. Parallel adaptive forest-of-octrees algorithms, implemented in the p4est

parallel adaptive mesh refinement library, are used for efficient parallel refinement, coarsening, mesh balancing, and repartitioning [26, 30, 70].

Successful solution of realistic mantle flow problems must overcome a number of computational challenges due to the severe nonlinearity and heterogeneity of Earth's rheology; moreover, the large algebraic systems exhibit anisotropies upon linearization with Newton's method. Nonlinear behavior at narrow plate boundary regions influences the motion of whole plates at continental scales, resulting in a wide range of spatial scales. Crucial features are highly localized with respect to Earth's radius ($\sim$6371 km), including plate thickness of order $\sim$50 km and shear zones at plate boundaries of order $\sim$5 km. Desired resolution at plate boundaries is below $\sim$1 km. However, a mesh of Earth's mantle with uniform resolution of 0.5 km would result in $\mathcal{O}(10^{13})$ degrees of freedom (DOF), which would be prohibitive for our mantle models with the complexity cited above. Thus adaptive methods are essential. Six orders of magnitude viscosity contrast is characteristic of the shear zones at plate boundaries, yielding sharp viscosity gradients and leading to severely poorly conditioned algebraic systems. Furthermore, the viscosity's dependence on a power of the second invariant of the strain rate tensor and plastic yielding phenomena lead to strongly nonlinear behavior.

This work builds directly on major advances in adaptivity [26], where global models exhibited 20 km wide plate boundaries, nonlinear viscosity, and yielding, and in turn demonstrated an unanticipated level of coupling between plate motion and the deep mantle [5], bounds on energy dissipation within plates [110], and rapid motion of small tectonic plates adjacent to large ones [6]. Nevertheless, such models did not close the gap between the fine-scale ($\sim$1 to 10 km) patterns of earthquakes, stress, and topography along plate boundaries with plate motions. Narrowing the local-to-global divide is essential for extracting the key observations allowing one to reach a new understanding of the physics of solid Earth processes.

The central computational challenge to reach these next steps in scientific discovery is to design implicit solvers and implementations for high-resolution global mantle flow models that can handle the extreme degrees of nonlinearity and poor conditioning that arise, the wide ranges of length scales and material properties, and the highly adapted meshes and necessary advanced discretizations. Such nonlinear and linear iterative solvers need robust and effective preconditioners with optimal (or nearly optimal) algorithmic scalability, while also scaling in parallel to the $\mathcal{O}(10^6)$ cores characteristic of leadership class supercomputers. While the conventional view has been that these goals are too difficult or even impossible to achieve, we aim to make them possible through a careful redesign of discretization, algorithms, solvers, and implementation.

1.2 Contributions

The first key contribution of this dissertation is *weighted BFBT (w-BFBT)*, a viscosity variation-robust preconditioner for the Schur complement of the linearized and discretized Stokes system. The second major contribution are, *hybrid spectral–geometric–algebraic multigrid (HMG)* methods, which

constitute a core component of the Stokes solver and are essential for preconditioning efficacy and algorithmic and parallel scalability. Third, the w-BFBT preconditioner and HMG linear solver are complemented by *inexact Newton–Krylov* methods to solve the highly nonlinear mantle convection problems; these nonlinear solves constitute the third major contribution.

We outline the key ideas of these contributions in the order they are used in solving (1.1).

Inexact Newton–Krylov methods

We employ an inexact Newton–Krylov method [42, 89] for the nonlinear Stokes equations (1.1), i.e., we use a sequence of linearizations of (1.1) and inexactly solve the resulting linearized and discretized systems using a preconditioned Krylov method. The rheology described by the constitutive law (1.2) is modified such that it incorporates viscosity upper and lower bounds in a differentiable manner, permitting the use of Newton's method. To compute a Newton update $(\hat{\boldsymbol{u}}, \hat{p})$, we find the (inexact) solution of the linearized Stokes system,

$$-\nabla \cdot \left[\mu'(\boldsymbol{u})\left(\nabla\hat{\boldsymbol{u}} + \nabla\hat{\boldsymbol{u}}^{\mathsf{T}}\right)\right] + \nabla\hat{p} = -\boldsymbol{r}_{\text{mom}}, \tag{1.3a}$$

$$-\nabla \cdot \hat{\boldsymbol{u}} = -r_{\text{mass}}, \tag{1.3b}$$

with coefficient

$$\mu'(\boldsymbol{u}) = \mu(\boldsymbol{u})\,\mathbf{I} + \dot{\varepsilon}_{\text{II}}\frac{\partial\mu}{\partial\dot{\varepsilon}_{\text{II}}}\frac{(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^{\mathsf{T}}) \otimes (\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^{\mathsf{T}})}{|\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^{\mathsf{T}}|_F^2}, \tag{1.4}$$

where the current velocity is $\boldsymbol{u}$ and the residuals of the momentum and mass equations appear on the right-hand side of (1.3). Note that what plays the role of viscosity in the Newton step is an anisotropic fourth-order tensor (1.4), where "$\otimes$" is the outer product of two second-order tensors and $|\cdot|_F$ denotes the Frobenius norm.

Additionally, during the initial Newton iterations, grid continuation is performed between Newton steps, where the mesh is adapted to variations in the viscosity (1.2) that arise from the nonlinear dependence on the velocity. The residual of the momentum equation $\boldsymbol{r}_{\text{mom}}$ is measured in the H^{-1}-norm for the purposes of backtracking line search. This avoids overly conservative update step sizes that are significantly reduced from unity, especially when the current velocity–pressure pair is far from the solution.

The standard Newton linearization as presented in Equation (1.3) exhibits prohibitively poor convergence in the presence of the plastic yielding term, $\tau_{\text{yield}}/(2\dot{\varepsilon}_{\text{II}})$, of the mantle's rheology (1.2). This is addressed by a linearization modification by means of introducing an additional perturbation equation along with system (1.3). The effective computational cost of solving the perturbed linearized system is reduced such that it is essentially the same as for the standard linearization, but Newton convergence improves dramatically. This together with the entire inexact Newton–Krylov method is presented in Chapter 5.

Weighted BFBT (w-BFBT) preconditioner for the Schur complement

Discretizing the linearized system (1.3) and preconditioning with an upper triangular block matrix as explained in Chapter 4 yields the following iterative scheme for the algebraic Stokes system:

$$\underbrace{\begin{bmatrix}\mathbf{A} & \mathbf{B}^\mathsf{T}\\ \mathbf{B} & \mathbf{0}\end{bmatrix}}_{\text{Stokes operator}}\underbrace{\begin{bmatrix}\tilde{\mathbf{A}} & \mathbf{B}^\mathsf{T}\\ \mathbf{0} & \tilde{\mathbf{S}}\end{bmatrix}^{-1}}_{\text{Preconditioner}}\begin{bmatrix}\tilde{\mathbf{u}}\\ \tilde{\mathbf{p}}\end{bmatrix} = \begin{bmatrix}-\mathbf{r}_{\text{mom}}\\ -\mathbf{r}_{\text{mass}}\end{bmatrix}, \tag{1.5}$$

where $\mathbf{A}$, $\mathbf{B}^\mathsf{T}$, and $\mathbf{B}$ denote the discretized viscous stress, gradient, and divergence operators, respectively. When solving (1.5) with an iterative Krylov method, an effective approximation of the Schur complement, $\mathbf{S} \coloneqq \mathbf{B}\mathbf{A}^{-1}\mathbf{B}^\mathsf{T}$, is an essential ingredient for attaining fast convergence of Schur complement-based iterative solvers. More precisely, a sufficiently good and fast-to-apply approximation of the inverse Schur complement $\tilde{\mathbf{S}}^{-1} \approx \mathbf{S}^{-1}$ is sought, together with an approximation of the inverse viscous block $\tilde{\mathbf{A}}^{-1} \approx \mathbf{A}^{-1}$.

BFBT or Least Squares Commutator methods [43, 44, 105] derive such an approximation $\tilde{\mathbf{S}}^{-1}$ by seeking a commutator matrix $\mathbf{X}$ such that the following commutator nearly vanishes,

$$\mathbf{A}\mathbf{D}^{-1}\mathbf{B}^\mathsf{T} - \mathbf{B}^\mathsf{T}\mathbf{X} \approx \mathbf{0}, \tag{1.6}$$

for a given diagonal matrix $\mathbf{D}^{-1}$. The motivation for seeking a near-commutator $\mathbf{X}$ is that (1.6) can be rearranged to obtain

$$\mathbf{S} \approx \big(\mathbf{B}\mathbf{D}^{-1}\mathbf{B}^\mathsf{T}\big)\mathbf{X}^{-1}, \tag{1.7}$$

where the closer the commutator is to zero, the more accurate the approximation [46]. The goal of finding a vanishing commutator can be recast as solving a least-squares minimization problem, whose solution gives us

$$\mathbf{X} = \big(\mathbf{B}\mathbf{C}^{-1}\mathbf{B}^\mathsf{T}\big)^{-1}\big(\mathbf{B}\mathbf{C}^{-1}\mathbf{A}\mathbf{D}^{-1}\mathbf{B}^\mathsf{T}\big) \tag{1.8}$$

for a given diagonal matrix $\mathbf{C}^{-1}$. Finally, substituting (1.8) into (1.7) and inverting yields the BFBT approximation of $\mathbf{S}^{-1}$.

The choice of diagonal weighting matrices $\mathbf{C}$, $\mathbf{D}$ is crucial for the efficacy of BFBT, especially in the presence of a heterogeneous viscosity. This is demonstrated in [85, 97], where the weighting matrices are based on $\text{diag}(\mathbf{A})$. However, drawbacks with such choices of weighting matrices include convergence issues for some combinations of viscosities and meshes, problems with higher discretization orders, and limited possibilities for analytical analysis due to the heuristic approach of using matrix entries of $\mathbf{A}$. To overcome these drawbacks, we propose the following weighted BFBT approximation for the inverse Schur complement:

$$\tilde{\mathbf{S}}^{-1}_{\text{w-BFBT}} \coloneqq \big(\mathbf{B}\mathbf{C}_{w_l}^{-1}\mathbf{B}^\mathsf{T}\big)^{-1}\big(\mathbf{B}\mathbf{C}_{w_l}^{-1}\mathbf{A}\mathbf{D}_{w_r}^{-1}\mathbf{B}^\mathsf{T}\big)\big(\mathbf{B}\mathbf{D}_{w_r}^{-1}\mathbf{B}^\mathsf{T}\big)^{-1}, \tag{1.9}$$

where $\mathbf{C}_{w_l} = \tilde{\mathbf{M}}_{\boldsymbol{u}}(w_l)$ and $\mathbf{D}_{w_r} = \tilde{\mathbf{M}}_{\boldsymbol{u}}(w_r)$ are lumped velocity space mass matrices that are weighted by the square root of the viscosity, i.e., usually $w_l(\boldsymbol{x}) = \sqrt{\mu(\boldsymbol{x})} = w_r(\boldsymbol{x})$, $\boldsymbol{x} \in \Omega$. w-BFBT results in a

highly robust implicit solver and represents a significant improvement over the previous state of the art Schur preconditioners in terms of both viscosity variation-robustness and algorithmic scalability.

Chapter 6 contains a detailed derivation of w-BFBT and a theoretical analysis that estimates its efficacy by means of providing bounds on the ratio of maximal to minimal eigenvalues of the preconditioned system $\tilde{\mathbf{S}}^{-1}_{\text{w-BFBT}}\mathbf{S}$. The theoretical discussions are supported by extensive numerical experiments ranging from eigenvalue calculations, robustness tests with challenging benchmark problems, modifications such that $w_l \neq w_r$, and algorithmic and parallel scalability (Chapter 8).

Hybrid spectral–geometric–algebraic multigrid (HMG)

Approximate inversion of the viscous block $\mathbf{A}$, an elliptic differential operator in the iterative scheme (1.5), is well suited for multigrid V-cycles, which we denote as $\tilde{\mathbf{A}}^{-1}$. To this end, we developed a hybrid spectral–geometric–algebraic multigrid (HMG) method, which exhibits extreme parallel scalability and retains nearly optimal algorithmic scalability. HMG initially reduces the discretization order (spectral multigrid); after arriving at order one, it continues by coarsening mesh elements (geometric multigrid); once the degrees of freedom fall below a threshold, algebraic multigrid (AMG) carries out further coarsening until a direct solve can be computed efficiently. Parallel forest-of-octrees algorithms of the p4est parallel adaptive mesh refinement (AMR) library are used for efficient, scalable mesh coarsening, mesh balancing, and repartitioning in the geometric HMG phase. During parallel geometric coarsening, the number of compute cores and the size of the MPI communicator is reduced successively to minimize communication. Re-discretization of the differential equations is performed on each coarser spectral and geometric level, where a coarse viscosity field is formed via L^2-projection. The transition to AMG is done at a sufficiently small core count and small MPI communicator.

Other elliptic differential operators that require approximate inversion are found in the w-BFBT approximation of the inverse Schur complement (1.9). The operators $\left(\mathbf{B}\mathbf{C}^{-1}_{w_l}\mathbf{B}^{\mathsf{T}}\right)$ and $\left(\mathbf{B}\mathbf{D}^{-1}_{w_r}\mathbf{B}^{\mathsf{T}}\right)$ can be viewed as discrete, variable-coefficient Poisson operators acting on the pressure in the discontinuous space $\mathbb{P}^{\text{disc}}_{k-1}$ with Neumann boundary conditions. Therefore, multigrid V-cycles can also be employed to approximate their inverses. However, it turned out to be problematic to apply multigrid coarsening directly due to the discontinuous, modal discretization of the pressure. We took a novel approach in [97] by considering the underlying infinite-dimensional, variable-coefficient Poisson operator, where the coefficient is derived from the diagonal weighting matrix (here, $\mathbf{C}^{-1}_{w_l}$ and $\mathbf{D}^{-1}_{w_r}$). Then we re-discretize with continuous, nodal high-order finite elements in $\mathbb{Q}_k$, borrowed from the discretization of the velocity. This continuous, nodal discretization of the pressure Poisson operator is then approximately inverted with an HMG V-cycle that is similar to the one described above for the inverse viscous block approximation $\tilde{\mathbf{A}}^{-1}$. Additional smoothing is applied in the discontinuous pressure space to account for high frequency modes that are introduced through projections between $\mathbb{Q}_k$ and $\mathbb{P}^{\text{disc}}_{k-1}$.

Our hybrid multigrid method combines high-order L^2-restriction and interpolation operators and employs Chebyshev-accelerated point-Jacobi smoothers (from PETSc [10]). This results in optimal

(or nearly optimal) algorithmic multigrid performance, i.e., iteration numbers are independent of mesh size (and only mildly dependent on discretization order), while maintaining robustness with respect to highly heterogeneous coefficients. In addition, the efficacy of the HMG preconditioner does not deteriorate with increasing core counts, because the spectral and geometric multigrid is by construction independent of the number of cores and AMG (from PETSc [10]) is invoked for prescribed small problem sizes on essentially fixed small core counts. All of these properties of HMG are crucial for overall Stokes solver performance and scalability.

Chapter 7 presents abstract derivations regarding multigrid methods and discusses in detail the design of our HMG algorithms, its parallel implementation, and optimizations. Parallel scalability involving benchmark problems and global mantle flow simulations with aggressively adapted meshes is highlighted in Chapter 8 on multiple supercomputers and hardware architectures.

1.3 Classifying research under CSEM Ph.D. program disciplinary areas

Area A – Applicable mathematics. All of the major contributions in this dissertation, Newton–Krylov nonlinear solver, w-BFBT preconditioner, and HMG linear solver, are introduced through rigorous mathematical analysis. For Newton's method, a perturbed linearization is derived systematically in an abstract form. This generalizes results from the literature on modified linearizations for image restoration problems and, additionally, introduces ideas from constrained optimization. Thanks to our general derivations, the results are then straightforwardly applied to multiple problems including image restoration and nonlinear mantle convection. The weighting matrices of BFBT preconditioners were originally based on heuristic arguments. This dissertation and [99] propose the first theoretical analysis of the spectrum of the w-BFBT-preconditioned Schur complement such that robustness optimality is established, which advances the understanding of BFBT-type methods in the context of Stokes problems. Finally, an abstract framework for multigrid methods and its mathematical analysis yields representations of error operators that drive the development of our HMG methods. By starting with this theoretical foundation, we achieve the high solver efficacy as well as algorithmic and parallel scalability that are crucial to addressing the challenges of mantle flow problems.

Area B – Numerical analysis and scientific computation. The new parallel HMG methods for high-order discretizations on locally adapted meshes feature algorithmic optimality that is combined with excellent parallel scalability. The efficacy of HMG does not depend on the number of processors by design, the setup cost is negligibly low compared to the solve time even at extreme scale, and the multigrid smoothers at the spectral and geometric multigrid levels are matrix-free and thus make efficient use of memory. Our careful parallel implementation of sophisticated algorithms targets distributed and shared memory architectures of leadership-class supercomputers and involves various code optimizations, e.g., overlapping of communication with computation and speeding up of low-level

kernels of matrix-free smoothers. Many numerical experiment throughout this dissertation test and document different aspects of our solvers in the form of convergence rates, setup and solver runtimes, and time-to-accuracy. Our w-BFBT method for approximating the inverse Schur complement is based on results from the literature on BFBT/Least Squares Commutator methods. Different from previous formulations, however, we propose an HMG-based approach that does not require matrix assembly but is based on scalable, efficient, matrix-free HMG V-cycles. These HMG and Schur complement preconditioners allow excellent parallel scalability of our Stokes solver to 1.6 million CPU cores as we demonstrate in this dissertation and [97].

Area C – Mathematical modeling and applications. The mathematical and computational advances in this dissertation are driven by the multifaceted challenges of our target problem, the global simulation of Earth's instantaneous mantle convection and associated plate tectonics with realistic parameters and high resolutions down to faulted plate boundaries. Careful attention is given to modeling mantle convection and inversion of model parameters. Numerical simulations are carried out using real data describing temperature distribution and plate boundaries. The output of these simulations is matched to observations, e.g., from plate velocity data sets, in order to validate simulation results.

2

Forward Problem: Modeling Earth's Mantle Convection

This chapter introduces the governing equations and constitutive relations for our mantle convection models. It is mainly based on the literature on geodynamics and Earth mantle physics [5, 104, 116], mathematical treatment of geosciences [51], and fundamentals of continuum mechanics [4, 56].

2.1 Conservation of mass and momentum

Given a velocity field $\boldsymbol{u}$ that takes on values in $\mathbb{R}^3$ and denoting time as t, we introduce the total (or convective) derivative for a physical quantity,

$$\frac{D}{Dt} \coloneqq \frac{\partial}{\partial t} + \boldsymbol{u} \cdot \nabla,$$

to formulate the mass conservation law:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}) = \frac{1}{\rho} \frac{D\rho}{Dt} + \nabla \cdot \boldsymbol{u} = 0, \tag{2.1}$$

where ρ is the density per unit volume. This is the differential form of mass conservation for an infinitesimal volume element. In case of an incompressible fluid,

$$\frac{D\rho}{Dt} \equiv 0, \tag{2.2}$$

and the conservation of mass (2.1) reduces to

$$\nabla \cdot \boldsymbol{u} = 0,$$

which is also known as the continuity equation.

The stress tensor is composed of a normal stress component and a viscous stress component:

$$\boldsymbol{\sigma} = -p\mathbf{I} + \boldsymbol{\tau}. \tag{2.3}$$

The pressure, $-p\mathbf{I}$, where $\mathbf{I}$ is the second-order identity tensor, is thermodynamic in origin and is maintained by molecular collisions. The viscous stress tensor, $\boldsymbol{\tau}$, depends on gradients of the velocity and is due to relative motion on the continuum scale. For an isotropic fluid, the viscous (or deviatoric) stress tensor is of the following form:

$$\boldsymbol{\tau} = \mu\big(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\mathsf{T}}\big) + \lambda\big(\nabla \cdot \boldsymbol{u}\big)\mathbf{I},$$

where μ is called the (shear) viscosity and λ the bulk viscosity. If a fluid is also incompressible, i.e., $\nabla \cdot \boldsymbol{u} = 0$, the viscous stress tensor simplifies to

$$\boldsymbol{\tau} = \mu\big(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\mathsf{T}}\big) = 2\mu\dot{\boldsymbol{\varepsilon}}, \tag{2.4}$$

where we defined the strain rate tensor as $\dot{\boldsymbol{\varepsilon}} \coloneqq \frac{1}{2}\big(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\mathsf{T}}\big)$. With these definitions and given a body force $\boldsymbol{f}$ that is a vector field in $\mathbb{R}^3$ acting on a unit volume, the law of momentum conservation requires the balance of forces:

$$\rho \frac{D\boldsymbol{u}}{Dt} = -\nabla p + \nabla \cdot \boldsymbol{\tau} + \boldsymbol{f}, \tag{2.5}$$

which, for an incompressible fluid, can be written as

$$\rho \frac{D\boldsymbol{u}}{Dt} = -\nabla p + \nabla \cdot \big(2\mu\dot{\boldsymbol{\varepsilon}}\big) + \boldsymbol{f}.$$

Summarizing Equations (2.1) and (2.5), we obtain a set of Navier–Stokes equations that govern the flow of an incompressible, isotropic fluid:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}) = 0 \qquad \text{(conservation of mass)} \tag{2.6a}$$

$$\rho \frac{D\boldsymbol{u}}{Dt} = -\nabla p + \nabla \cdot \big(2\mu\dot{\boldsymbol{\varepsilon}}\big) + \boldsymbol{f} \quad \text{(conservation of momentum)} \tag{2.6b}$$

If the viscosity μ is independent of the velocity $\boldsymbol{u}$ or its derivatives, then the relation between the viscous stress tensor $\boldsymbol{\tau}$ and the strain rate tensor $\dot{\boldsymbol{\varepsilon}}$ is linear and such fluids are called Newtonian. Otherwise fluids are referred to as non-Newtonian.

2.2 Conservation of energy

Specifying the thermodynamic state variables, pressure p, specific volume $v = 1/\rho$, and temperature T, we define the coefficient of thermal expansion (or thermal expansivity of a material)

$$\alpha \coloneqq \frac{1}{v}\frac{\partial v}{\partial T}\bigg|_p = -\frac{1}{\rho}\frac{\partial \rho}{\partial T}\bigg|_p,$$

where the subscript p means that this variable is held fixed. Let q denote the internal heat, then the specific heat at constant pressure is defined by

$$c_p \coloneqq \frac{\delta q}{\delta T}\bigg|_p,$$

i.e., it is the ratio of the increment of heat added to the material δq to the change in temperature δT. Given the thermal conductivity, k, and the rate of internal heat production per unit mass, H, the conservation of energy requires, for an infinitesimal volume element of a fluid, that

$$\rho c_p \frac{DT}{Dt} - \alpha T \frac{D\rho}{Dt} = \nabla \cdot \big(k \nabla T\big) + \boldsymbol{\tau} : \nabla \boldsymbol{u} + \rho H, \tag{2.7}$$

where ":" represents the inner product of second-order tensors. The three terms on the right-hand side of Equation (2.7) represent, respectively, thermal conduction, volumetric heat production due to viscous dissipation, and internal heat generation. For an incompressible fluid satisfying Equations (2.2) and (2.4) and due to the symmetry of $\boldsymbol{\tau}$, the thermal energy equation (2.7) takes the form

$$\rho c_p \frac{DT}{Dt} = \nabla \cdot \big(k \nabla T\big) + 2\mu \big(\dot{\boldsymbol{\varepsilon}} : \dot{\boldsymbol{\varepsilon}}\big) + \rho H.$$

2.3 Approximations for mantle convection

In this section, we simplify the general equations for conservation of mass and momentum (2.6) as well as energy (2.7) for their application to thermal convection in Earth's mantle. The density of the mantle is mainly a function of temperature and pressure. Moreover, most of the density change in the mantle is due to hydrostatic compression, whereas density changes associated with convection are small compared to the spherically averaged density of the mantle. Therefore, after separating the variables for density, temperature, and pressure into reference states and departures from it,

$$\rho = \bar{\rho} + \rho', \quad T = \bar{T} + T', \quad p = \bar{p} + p',$$

we assume that the mantle density, $\rho = \rho\,(T, p)$, is linearized in the convecting state and of the form:

$$\rho = \bar{\rho}\,\big(\bar{T}, \bar{p}\big) + \rho' = \bar{\rho}\,\big(\bar{T}, \bar{p}\big) - \bar{\rho}\bar{\alpha} T' + \bar{\rho}\bar{\chi}_T p', \tag{2.8}$$

where $\bar{\chi}_T$ is the reference state of the isothermal compressibility of a material at constant temperature, which is defined by

$$\chi_T \coloneqq -\frac{1}{v}\frac{\partial v}{\partial p}\bigg|_T = \frac{1}{\rho}\frac{\partial \rho}{\partial p}\bigg|_T.$$

Definition 2.3.1 (Nondimensionalization)**.** We define the following scaling factors used for nondimensionalization:

representative thermal expansivity:	α_r
representative isothermal compressibility:	χ_{T_r}
representative specific heat at constant pressure:	c_{p_r}
representative density:	ρ_r
characteristic temperature difference:	T'_r
representative thermal conductivity:	k_r

representative viscosity:	μ_r
representative gravitational acceleration:	g_r
depth of convecting region:	b

and let the thermal diffusivity be $\kappa_r \coloneqq k_r/\rho_r c_{p_r}$. The scaling factors between dimensional and dimensionless variables, which are denoted with an asterisk, are

$$\begin{aligned} \rho &= \rho_r \rho^*, \quad & T' &= T'_r T^*, \\ \alpha &= \alpha_r \alpha^*, \quad & \chi_T &= \chi_{T_r} \chi_T^*, \\ \boldsymbol{x} &= b\boldsymbol{x}^*, & t &= \frac{\rho_r c_{p_r} b^2}{k_r} t^* = \frac{b^2}{\kappa_r} t^* \end{aligned}$$

and

$$\boldsymbol{u} = \frac{k_r}{\rho_r c_{p_r} b} \boldsymbol{u}^* = \frac{\kappa_r}{b} \boldsymbol{u}^*, \quad p' = \frac{\mu_r k_r}{\rho_r c_{p_r} b^2} p^* = \frac{\mu_r \kappa_r}{b^2} p^*.$$

However, we will drop the asterisk in the equations that follow for simplicity. Note that the viscous stress has the same scaling factor as the pressure, which can be seen from Equation (2.3).

With these definitions, we rewrite Equation (2.8) in its dimensionless form (omitting the asterisk):

$$\frac{\rho}{\bar{\rho}} = 1 - \bar{\alpha} T' \varepsilon_T + \bar{\chi}_T p' \varepsilon_p,$$

where

$$\varepsilon_T \coloneqq \alpha_r T'_r, \quad \varepsilon_p \coloneqq \frac{k_r^2 \chi_{T_r}}{\rho_r c_{p_r}^2 b^2} Pr, \quad Pr \coloneqq \frac{\mu_r c_{p_r}}{k_r}.$$

The dimensionless parameter Pr is called the Prandtl number. For the mantle, we can estimate the factors ε_T and ε_p to be sufficiently small, that is, $\varepsilon_T \approx 3 \times 10^{-2}$ and $\varepsilon_p \approx 2 \times 10^{-10}$ [104]. In addition, the dimensionless quantities $\bar{\alpha} T'$ and $\bar{\chi}_T p'$ are of order unity or smaller. Thus, the density perturbations due to mantle convection are small compared to the reference hydrostatic density and we can assume that $\rho'/\bar{\rho} \ll 1$.

Dynamically, phenomena in seismology and mantle convection do not overlap, since they occur on vastly different time scales. The anelastic form of the mass conservation equation eliminates seismic waves from the equations governing mantle convection, i.e., $\frac{\partial \rho'}{\partial t} \equiv 0$. With the assumptions from above, $\varepsilon_T, \varepsilon_p \ll 1$, the conservation of mass from Equation (2.1) reduces to

$$\nabla \cdot (\bar{\rho} \boldsymbol{u}) = 0.$$

If we further assume that the effects of compressibility are not dominant in mantle convection [104], we can eliminate variations in density $\bar{\rho}$ across the mantle and adopt the incompressible continuity equation

$$\nabla \cdot \boldsymbol{u} = 0.$$

This incompressible form is called the Boussinesq approximation.

We consider gravity to be the only body force in the momentum conservation equation (2.5),

$$\boldsymbol{f}(\boldsymbol{x}) = -\rho g \boldsymbol{e}_r(\boldsymbol{x}), \tag{2.9}$$

where g denotes the gravitational acceleration and $\boldsymbol{e}_r \in \mathbb{R}^3$, $|\boldsymbol{e}_r| = 1$, is the radial direction. First, we assume the variations in gravitational acceleration due to convection to be negligible. Second, if the reference state pressure satisfies a hydrostatic equation, i.e., when $\boldsymbol{u} \equiv 0$,

$$0 = -\nabla \bar{p} - \bar{\rho} g \boldsymbol{e}_r,$$

then the hydrostatic pressure $\bar{p}$ can be eliminated from Equation (2.5),

$$\rho \frac{D\boldsymbol{u}}{Dt} = -\nabla p + \nabla \cdot \boldsymbol{\tau} + \boldsymbol{f} = -\nabla(\bar{p} + p') + \nabla \cdot \boldsymbol{\tau} - (\bar{\rho} + \rho') g \boldsymbol{e}_r = -\nabla p' + \nabla \cdot \boldsymbol{\tau} - \rho' g \boldsymbol{e}_r.$$

Third, we assume the Prandtl number Pr to be sufficiently large such that the inertial force term can be eliminated, i.e., $\rho \frac{D\boldsymbol{u}}{Dt} \approx 0$. Finally, we apply the Boussinesq approximation of incompressibility and obtain the following simplified dimensionless form of the momentum equation:

$$0 = -\nabla p' + \nabla \cdot \left(2\mu \dot{\boldsymbol{\varepsilon}}\right) + Ra\, \bar{\rho} \bar{\alpha} T' \boldsymbol{e}_r. \tag{2.10}$$

In this form, we incorporated the assumption that variations in temperature are the major driving force for thermal convection; hence, the deviation ρ' from the reference density $\bar{\rho}$ in Equation (2.8) reduces to $\rho' = -\bar{\rho}\bar{\alpha}T'\varepsilon_T$. To obtain Equation (2.10) above, we define the Rayleigh number,

$$Ra \coloneqq \frac{\alpha_r T_r' \rho_r^2 g_r b^3 c_{p_r}}{\mu_r k_r} = \frac{\alpha_r T_r' \rho_r g_r b^3}{\mu_r \kappa_r} \tag{2.11}$$

with the representative gravitational acceleration g_r. The Rayleigh number multiplies the buoyancy force term and controls the vigor of convection.

For the energy conservation, we take the anelastic limit and apply the Boussinesq approximation. Then the nondimensional form of the energy equation simplifies to

$$\bar{\rho}\bar{c}_p \frac{DT'}{Dt} = \nabla \cdot \left(\bar{k} \nabla T'\right) + \bar{\rho} H \left(\frac{b^2 H_r \rho_r}{k_r T_r'}\right),$$

where we introduced a representative internal heating rate H_r (see [104] for more details). We summarize the results in the following corollary and provide our set of representative parameters used for nondimensionalization in Table 2.1 as well as typical values from the literature in Table 2.2.

Corollary 2.3.2 (Mantle approximation)**.** *The thermal convection in Earth's mantle is modeled by the anelastic, Boussinesq, infinite Prandtl number approximations of the conservation of mass, momentum, and energy equations* (2.6) *and* (2.7)*. The non-hydrostatic form of the governing equations is a set of Stokes equations with an additional energy equation:*

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \textit{(conservation of mass / continuity)} \tag{2.12a}$$

$$-\nabla \cdot \left(2\mu \dot{\boldsymbol{\varepsilon}}\right) + \nabla p' = Ra\, \bar{\rho}\bar{\alpha} T' \boldsymbol{e}_r \qquad \textit{(conservation of momentum)} \tag{2.12b}$$

$$\bar{\rho}\bar{c}_p \frac{DT'}{Dt} - \nabla \cdot \left(\bar{k} \nabla T'\right) = \bar{\rho} H \left(\frac{b^2 H_r \rho_r}{k_r T_r'}\right) \qquad \textit{(conservation of energy)} \tag{2.12c}$$

Table 2.1: Mantle convection parameters for nondimensionalization and temperature-driven buoyancy.

Parameter	**Symbol**	**[Unit]**	**Value**
Thermal expansivity	α_r	$[\mathrm{K}^{-1}]$	2×10^{-5}
Density	ρ_r	$[\mathrm{kg\,m}^{-3}]$	3300
Temperature difference	T_r'	[K]	1400
Thermal diffusivity	κ_r	$[\mathrm{m}^2\,\mathrm{s}^{-1}]$	10^{-6}
Representative viscosity	μ_r	[Pa s]	10^{20}
Gravitational acceleration	g_r	$[\mathrm{m\,s}^{-2}]$	9.81
Depth of convecting region	b	[m]	6371×10^3
Rayleigh number	Ra	–	2.34×10^9

Table 2.2: Mantle convection parameters from literature.

Parameter	**Symbol**	**[Unit]**	Schubert et al. [104]	Alisic et al. [5]	Ratnaswamy et al. [93]
Thermal expansivity	α_r	$[\mathrm{K}^{-1}]$	3×10^{-5}	2×10^{-5}	2×10^{-5}
Isothermal compressibility	χ_{T_r}	$[\mathrm{Pa}^{-1}]$	3×10^{-12}	–	–
Specific heat	c_{p_r}	$[\mathrm{kJ\,kg}^{-1}\,\mathrm{K}^{-1}]$	1	–	–
Density	ρ_r	$[\mathrm{kg\,m}^{-3}]$	4000	3300	3300
Temperature difference	T_r'	[K]	1000	1400	1400
Thermal conductivity	k_r	$[\mathrm{W\,m}^{-1}\,\mathrm{K}^{-1}]$	4	–	–
Thermal diffusivity	κ_r	$[\mathrm{m}^2\,\mathrm{s}^{-1}]$	10^{-6}	10^{-6}	10^{-6}
Representative viscosity	μ_r	[Pa s]	10^{21}	10^{20}	10^{20}
Gravitational acceleration	g_r	$[\mathrm{m\,s}^{-2}]$	9.81	9.81	9.81
Depth of convecting region	b	[m]	3000×10^3	6371×10^3	1500×10^3
Rayleigh number	Ra	–	3.18×10^7	2.34×10^9	3.06×10^7

The conservation of momentum equation (2.12b) can also be expressed in a form without eliminating the hydrostatic relation, namely:

$$-\nabla \cdot \left(2\mu\dot{\boldsymbol{\varepsilon}}\right) + \nabla p = Ra\,\bar{\rho}\left(\bar{\alpha}T' - \frac{1}{\alpha_r T_r'}\right)\boldsymbol{e}_r.$$

Note that the variables $\bar{\rho}$, $\bar{\alpha}$, and $\bar{c}_p$ are constant (and typically $\bar{\rho} = \bar{\alpha} = 1$), but neither the anelastic approximation nor the Boussinesq approximation require the viscosity μ or the thermal conductivity $\bar{k}$ to be constant.

2.4 Temperature variations in the mantle

Earth's mantle encompasses the globe's upper ∼3000 km, and our computational domain is an ideal spherical shell denoted by $\Omega \subset \mathbb{R}^d$, $d = 3$, with a smooth boundary $\partial\Omega$. We denote the boundary at the surface by $\Gamma_{\text{surf}} \subset \partial\Omega$ and the core–mantle boundary at 2867 km depth by $\Gamma_{\text{core}} \subset \partial\Omega$.

The lithosphere is a fundamental feature of the plate tectonics theory and describes an outer shell in the mantle that remains rigid during long time scales $\sim 10^6$ yr. The lithosphere of oceanic plates has an average thickness of 100 km whereas the continental lithosphere is typically estimated to be about 200 km thick. The asthenosphere is the region of the mantle beneath the lithosphere with a

lower viscosity due to higher temperatures. The boundary between lithosphere and asthenosphere can be defined in terms of mechanical, thermal, or elastic properties of the lithosphere [104]. The mechanical lithosphere is defined by an upper layer of rock that deforms slowly over geological time scales. The definition of the thermal lithosphere, on the other hand, prescribes a thermal boundary layer that constitutes the lithosphere. The base of this thermal boundary layer is typically chosen to be the depth at which the temperature has increased by 90% toward the asthenosphere's temperature.

The half-space cooling model provides an adequate first-order model for the temperature of oceanic plates of ages less than about 80 Myr. It is discussed in the remainder of this section because it is useful to describe the hydrostatic component of the body force (2.9). The governing (2-dimensional) convection-conduction equation for the temperature inside the plate is

$$u_{\mathrm{pl}} \frac{\partial T}{\partial x} = \kappa \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right), \tag{2.13}$$

where u_{pl} is the velocity of seafloor spreading and κ is the thermal diffusivity [104]. Let L be a typical distance from the ocean ridge, then we can define the Peclét number by $Pe \coloneqq u_{\mathrm{pl}} L/\kappa$. Typical values for the oceanic lithosphere are

$$L \approx 1000\,\mathrm{km}, \quad u_{\mathrm{pl}} \approx 50\,\mathrm{mm\,yr^{-1}}, \quad \kappa \approx 10^{-6}\,\mathrm{m^2\,s^{-1}}$$

and we find the Peclét number to be $Pe \approx 1600$.

For a large Peclét number, the horizontal temperature is convection-dominated, thus it is appropriate to neglect horizontal heat conduction, i.e., $\kappa \frac{\partial^2 T}{\partial x^2} \approx 0$. The boundary layer approximation of Equation (2.13) then becomes

$$u_{\mathrm{pl}} \frac{\partial T}{\partial x} = \kappa \frac{\partial^2 T}{\partial y^2}$$

and, since $t = x/u_{\mathrm{pl}}$, we can write

$$\frac{\partial T}{\partial t} = \kappa \frac{\partial^2 T}{\partial y^2}. \tag{2.14}$$

The partial differential equation (2.14) is augmented with boundary conditions

$$T = T_0 \quad \text{at } y = 0 \qquad \text{(at the surface)}, \tag{2.15a}$$

$$T \to T_1 \quad \text{as } y \to \infty \quad \text{(at infinite depth)}, \tag{2.15b}$$

where $0 \leq T_0 < T_1$, and the initial condition at $t = 0$

$$T = \begin{cases} T_0 & 0 = y, \\ T_1 & 0 < y. \end{cases} \tag{2.16}$$

Lemma 2.4.1 (Half-space cooling model)**.** *The solution of the partial differential equation* (2.14) *with boundary conditions* (2.15) *and initial condition* (2.16) *is given by*

$$\frac{T_1 - T}{T_1 - T_0} = \operatorname{erfc}\left(\frac{y}{2\sqrt{\kappa t}} \right). \tag{2.17}$$

Proof. We introduce the nondimensional similarity variables

$$\theta \coloneqq \frac{T_1 - T}{T_1 - T_0}, \quad \eta \coloneqq \frac{y}{2\sqrt{\kappa t}}. \tag{2.18}$$

Partial derivatives of (2.14) with respect to variables t and y transform to ordinary derivatives with respect to η:

$$\frac{\partial \theta}{\partial t} = \frac{d\theta}{d\eta}\frac{\partial \eta}{\partial t} = \frac{d\theta}{d\eta}\left(-\frac{y}{2\sqrt{\kappa t}}\frac{1}{2t}\right) = \frac{d\theta}{d\eta}\left(-\frac{\eta}{2t}\right), \tag{2.19}$$

$$\frac{\partial \theta}{\partial y} = \frac{d\theta}{d\eta}\frac{\partial \eta}{\partial y} = \frac{d\theta}{d\eta}\frac{1}{2\sqrt{\kappa t}} \quad \text{thus} \quad \frac{\partial^2 \theta}{\partial y^2} = \frac{d^2\theta}{d\eta^2}\frac{1}{2\sqrt{\kappa t}}\frac{\partial \eta}{\partial y} = \frac{d^2\theta}{d\eta^2}\frac{1}{4\kappa t}. \tag{2.20}$$

Substituting (2.19) and (2.20) into Equation (2.14) gives the ordinary differential equation

$$-\eta\frac{d\theta}{d\eta} = \frac{1}{2}\frac{d^2\theta}{d\eta^2} \tag{2.21}$$

with the boundary conditions

$$\theta = 1 \text{ at } \eta = 0, \quad \theta = 0 \text{ as } \eta \to \infty.$$

Now, it remains to solve an ordinary differential equation. First, let

$$\phi \coloneqq \frac{d\theta}{d\eta}, \tag{2.22}$$

then Equation (2.21) becomes

$$-\eta\phi = \frac{1}{2}\frac{d\phi}{d\eta} \quad \Rightarrow \quad -\eta\, d\eta = \frac{1}{2}\frac{1}{\phi}d\phi.$$

Integration yields

$$-\eta^2 = \ln\phi - \ln c_1,$$

where we introduced the constant of integration $-\ln c_1$. It follows, using definition (2.22), that

$$\frac{d\theta}{d\eta} = c_1\,\mathrm{e}^{-\eta^2} \quad \Rightarrow \quad d\theta = c_1\,\mathrm{e}^{-\eta^2}\,d\eta$$

and upon integration we obtain

$$\theta(\eta) - 1 = c_1\int_0^\eta \mathrm{e}^{-\xi^2}\,\mathrm{d}\xi, \tag{2.23}$$

where we evaluated the constant of integration using the boundary condition $\theta(0) = 1$. Further, since the following definite integral satisfies

$$\int_0^\infty \mathrm{e}^{-\xi^2}\,\mathrm{d}\xi = \frac{\sqrt{\pi}}{2}$$

and exploiting the boundary condition $\theta(\eta \to \infty) = 0$, we find the constant $c_1 = -2/\sqrt{\pi}$. Substituting c_1 into Equation (2.23), we get the solution of Equation (2.21)

$$\theta(\eta) = 1 - \frac{2}{\sqrt{\pi}}\int_0^\eta \mathrm{e}^{-\xi^2}\,\mathrm{d}\xi = 1 - \mathrm{erf}\,(\eta) = \mathrm{erfc}\,(\eta)\,.$$

The solution (2.17) in terms of the original variables is obtained by replacing the similarity variables with their definitions in (2.18). □

Definition 2.4.2 (Thermal boundary layer)**.** Based on the notion of the thermal lithosphere and the result from Lemma 2.4.1, we define the thermal boundary layer constituting the oceanic lithosphere to have a thickness $y_{\mathrm{TBL}} \in (0, \infty)$ such that

$$\operatorname{erfc}\left(\frac{y_{\mathrm{TBL}}}{2\sqrt{\kappa t}}\right) = \theta_{\mathrm{TBL}} \coloneqq 0.1,$$

which means we set the increase in temperature to 90%. This, in turn, results in $\eta_{\mathrm{TBL}} = \operatorname{erfc}^{-1}(\theta_{\mathrm{TBL}})$ and the thickness of the thermal boundary layer depending on the plate's age t is

$$y_{\mathrm{TBL}}(t) = 2\,\eta_{\mathrm{TBL}}\sqrt{\kappa t} = 2\operatorname{erfc}^{-1}(\theta_{\mathrm{TBL}})\sqrt{\kappa t} \approx 2.32\sqrt{\kappa t}.$$

In the mantle approximation of the momentum equation (2.12b), the hydrostatic relation is eliminated and hence the temperature-driven right-hand side forcing excludes the adiabatic temperature that only changes with depth but not laterally. In practice, we eliminate the adiabatic temperature from the forcing by subtracting a background temperature on the right-hand side, which is derived from the half-space cooling model.

Definition 2.4.3 (Background temperature)**.** The background temperature $T_{t_{\mathrm{pl}}}$ corresponding to a plate of age t_{pl} is derived from the solution (2.17) of the half-space cooling model with $T_0 = 0$ and $T_1 = 1$ in Lemma 2.4.1. It is thus defined by:

$$T_{t_{\mathrm{pl}}}(r) \coloneqq \operatorname{erf}\left(\frac{R - r}{2\sqrt{\kappa t_{\mathrm{pl}}}}\right) \quad \text{for } 0 \le r \le R,$$

where $R = 6371\,\mathrm{km}$ denotes Earth's radius.

Definition 2.4.4 (Temperature dependent viscosity component)**.** The viscosity depends on the temperature via an Arrhenius relationship and this temperature dependent term is defined by:

$$a(T) \coloneqq c_a \exp\left(E_a\,(0.5 - T)\right) \quad \text{for } 0 \le T \le 1, \tag{2.24}$$

where the parameter E_a assumes the role of an activation energy and c_a is a constant scaling factor.

2.5 Decoupling of plates via weak zones

The surface of the earth is subdivided into plates. An example for a set of plates and their boundaries is shown in Figure 2.1. In order to model the decoupling of adjacent plates in mantle's rheology, we incorporate trenches, fracture zones, and ridges via thin low-viscosity layers. These layers are called weak zones and they are described geometrically by manifolds, i.e., two-dimensional surfaces, that extend from Earth's surface into its interior with a corresponding layer width.

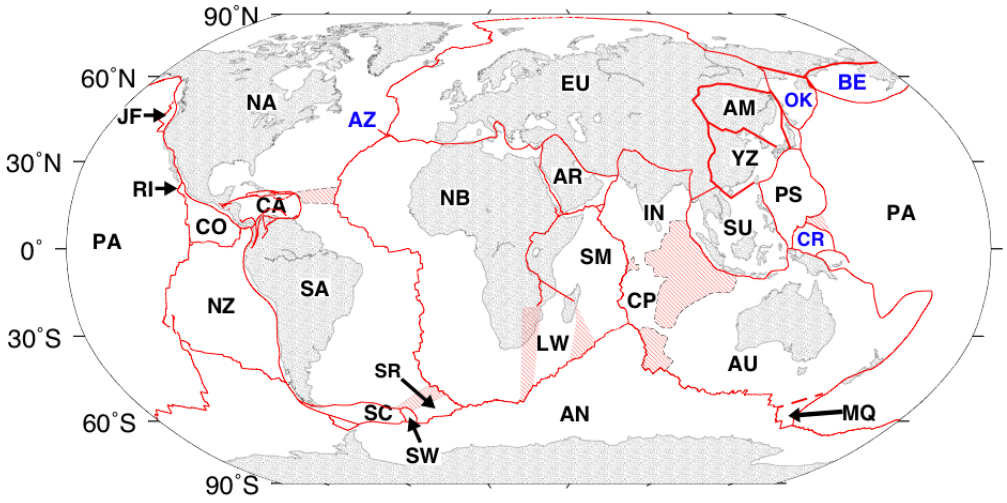


Figure 2.1: Plate boundaries *(red lines)* and plate geometries employed for MORVEL plate motion data set [36]. *Patterned red areas* show diffuse plate boundaries. Plate names are abbreviated by two letters. (Credit: DeMets, Gordon, Argus).

Definition 2.5.1 (Weak zone)**.** Given the geometric representation of weak zone manifolds and parameters

$$\begin{aligned} &\text{weak zone width:} && d_w, \\ &\text{plate boundary width:} && d_{\min}, \\ &\text{minimum weak zone factor:} && w_{\min} \in (0,1], \end{aligned}$$

the weak zone at any point $\boldsymbol{x}$ in the mantle domain Ω requires the distance $d(\boldsymbol{x})$ of $\boldsymbol{x}$ to the nearest manifold. Then the weakening factor is computed by

$$w(\boldsymbol{x}) \coloneqq 1 - (1 - w_{\min}) \exp\left(-\frac{\xi(\boldsymbol{x})^2}{2\sigma^2}\right) \in (0,1], \quad \boldsymbol{x} \in \Omega, \tag{2.25}$$

where

$$\xi(\boldsymbol{x}) \coloneqq \max(0, d(\boldsymbol{x}) - d_{\min}) \quad \text{and} \quad \sigma \coloneqq \frac{d_w - d_{\min}}{2}.$$

Note that Equation (2.25) incorporates a Gaussian-like mollifier that yields smooth weak zones. Furthermore, the minimum weak zone factor $w_{\min}$ is assumed for all points inside of the plate boundary layer of width $d_{\min}$. An example of a profile of a weak zone is shown in Figure 2.2. From a geodynamics perspective, areas of interesting activity around plate boundaries are about 40 km thick, within which we need to embed weak zones. It is desired that the weak zone is sufficiently thin such that we can observe dynamic behavior without imposing it through weak zones that are too wide. Weak zones at subduction zones need to be sufficiently thin such that the bending plate remains coherent. Therefore, acceptable choices for the weak zone width d_w are in the range 10–20 km. In case of weak zones that are too wide, unfortunately, the physical factors that govern the strength of weak zones (essentially

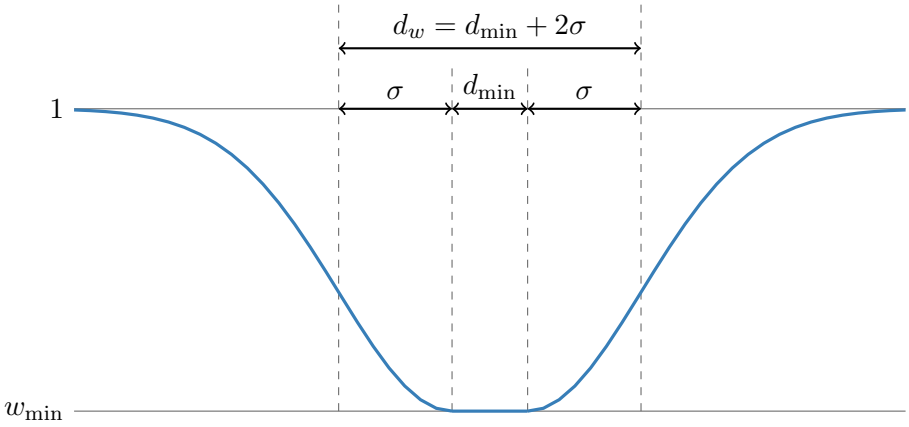


Figure 2.2: Weak zone profile with weak zone width $d_w = 20\,\mathrm{km}$, plate boundary width $d_{\min} = 5\,\mathrm{km}$, and weak zone factor $w_{\min} = 10^{-5}$.

the seismic megathrust) and the bending plate are not resolved. The megathrust and the bending plate are independent physical entities with a host of processes, which independently govern their strength.

2.6 A composite nonlinear viscosity model

The mantle's rheology can be modeled as a composite of linear viscosity (Newtonian rheology) and nonlinear viscosity (non-Newtonian rheology) [5, 110]. To describe the viscosities, first recall the definitions of the strain rate and viscous stress tensors, $\dot{\boldsymbol{\varepsilon}} = \frac{1}{2}\left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\mathsf{T}}\right)$ and $\boldsymbol{\tau} = 2\mu\dot{\boldsymbol{\varepsilon}}$, and next we define their second invariants.

Definition 2.6.1 (Second invariants)**.** The (square root of the) second invariant of the strain rate tensor is (using the Einstein summation convention)

$$\dot{\varepsilon}_{\mathrm{II}} = \left(\frac{1}{2}\left(\dot{\varepsilon}_{ij}\dot{\varepsilon}_{ij} - \dot{\varepsilon}_{kk}^2\right)\right)^{\frac{1}{2}} = \left(\dot{\varepsilon}_{12}^2 + \dot{\varepsilon}_{13}^2 + \dot{\varepsilon}_{23}^2 - \left(\dot{\varepsilon}_{11}\dot{\varepsilon}_{22} + \dot{\varepsilon}_{11}\dot{\varepsilon}_{33} + \dot{\varepsilon}_{22}\dot{\varepsilon}_{33}\right)\right)^{\frac{1}{2}}$$

The (square root of the) second invariant of the viscous stress tensor is

$$\tau_{\mathrm{II}} = \left(\frac{1}{2}\left(\tau_{ij}\tau_{ij} - \tau_{kk}^2\right)\right)^{\frac{1}{2}} = \left(\frac{1}{2}\left(4\mu^2\dot{\varepsilon}_{ij}\dot{\varepsilon}_{ij} - 4\mu^2\dot{\varepsilon}_{kk}^2\right)\right)^{\frac{1}{2}} = 2\mu\dot{\varepsilon}_{\mathrm{II}}.$$

In case of incompressible fluids, the second invariants become

$$\dot{\varepsilon}_{\mathrm{II}} = \left(\frac{1}{2}\dot{\varepsilon}_{ij}\dot{\varepsilon}_{ij}\right)^{\frac{1}{2}} = \left(\frac{1}{2}\,\dot{\boldsymbol{\varepsilon}} : \dot{\boldsymbol{\varepsilon}}\right)^{\frac{1}{2}}$$

and

$$\tau_{\mathrm{II}} = \left(\frac{1}{2}\tau_{ij}\tau_{ij}\right)^{\frac{1}{2}} = \left(\frac{1}{2}\,\boldsymbol{\tau} : \boldsymbol{\tau}\right)^{\frac{1}{2}}.$$

The mantle can be subdivided into two regions. The upper mantle is the outer shell of the mantle ranging from the surface Γ_{surf} to $\sim$660 km depth and the lower mantle is the inner shell between core Γ_{core} and upper mantle. In the lower mantle, the dominant rheological mechanism is linear diffusion creep, whereas in the upper mantle nonlinear dislocation creep dominates.

Definition 2.6.2 (Diffusion creep and dislocation creep)**.** Given the parameters (assumed to be positive unless noted otherwise)

grain size:	d
grain size exponent:	p
pre-exponent:	A
water content:	C_{OH}
water content exponent:	r
stress exponent:	$n \geq 1$
activation energy:	E_a
activation volume:	V_a
lithostatic pressure:	P
gas constant:	R

we define the linear diffusion creep viscosity

$$\mu_{\text{df}}(T) \coloneqq \left(\frac{d^p}{AC_{OH}^r}\right) \exp\left(\frac{E_a + PV_a}{RT}\right) \tag{2.26}$$

and the nonlinear dislocation creep viscosity

$$\mu_{\text{ds}}(T, \dot{\varepsilon}_{\text{II}}) \coloneqq \left(\frac{d^p}{AC_{OH}^r}\right)^{\frac{1}{n}} \dot{\varepsilon}_{\text{II}}^{\frac{1-n}{n}} \exp\left(\frac{E_a + PV_a}{nRT}\right) = \left(\mu_{\text{df}}(T)\, \dot{\varepsilon}_{\text{II}}\right)^{\frac{1}{n}} \dot{\varepsilon}_{\text{II}}^{-1}. \tag{2.27}$$

Note that the diffusion creep viscosity (2.26) is a special case of the dislocation creep viscosity (2.27) if $n = 1$. Hence, Equation (2.27) is a general form of the viscosity that allows for strain rate weakening if $n > 1$.

In addition, the upper mantle is subdivided into lithosphere and asthenosphere, which is based on thermal or mechanical properties of the rock (see Section 2.4). The temperature of the lithosphere is relatively low and, since the lithosphere moves mostly like a rigid body far away from plate boundaries, it is subjected to little strain. Therefore (linear) diffusion creep is the dominating mechanism in the lithosphere, which corresponds to the case $n = 1$ in Equation (2.27). Closer to a plate boundary, however, strain rates in the lithosphere increase such that deformations occur due to high stresses. In such regions, dislocation creep is activated with $n \approx 3$ in (2.27) and, additionally, a yielding law is introduced.

Definition 2.6.3 (Yielding law)**.** The yielding law requires that the viscous stress is bounded from above by a given yield strength $\tau_{\text{yield}} > 0$, i.e.,

$$\tau_{\text{II}} \leftarrow \min\left(\tau_{\text{II}}, \tau_{\text{yield}}\right) = \min\left(2\mu\dot{\varepsilon}_{\text{II}}, \tau_{\text{yield}}\right).$$

Yielding results in a new viscosity

$$\mu \leftarrow \min\left(\mu, \frac{\tau_{\text{yield}}}{2\dot{\varepsilon}_{\text{II}}}\right).$$

With the definitions above, we are able to construct a composite formulation of the viscosity as follows (see [5] for more details).

Definition 2.6.4 (Composite viscosity model)**.** Given are lower and upper viscosity bounds $0 < \mu_{\min} < \mu_{\max} < \infty$, yield strength $\tau_{\text{yield}} > 0$, and a yielding regularization $0 \leq \epsilon_{\text{reg}} < 1$. We first define the harmonic mean of diffusion and dislocation creep according to [13] that is bounded from above:

$$\mu_{\text{df,ds}}(T, \dot{\varepsilon}_{\text{II}}) \coloneqq \min\left(\frac{\mu_{\text{df}}(T)\,\mu_{\text{ds}}(T, \dot{\varepsilon}_{\text{II}})}{\mu_{\text{df}}(T) + \mu_{\text{ds}}(T, \dot{\varepsilon}_{\text{II}})}, \mu_{\max}\right).$$

Our composite viscosity is more involved,

$$\mu_{\text{comp}}(T, \dot{\varepsilon}_{\text{II}}) \coloneqq \max\left(\mu_{\min}, w\left(\boldsymbol{x}\right)\left[(1-\epsilon_{\text{reg}})\min\left(\mu_{\text{df,ds}}(T, \dot{\varepsilon}_{\text{II}}), \frac{\tau_{\text{yield}}(\boldsymbol{x})}{2\dot{\varepsilon}_{\text{II}}}\right) + \epsilon_{\text{reg}}\mu_{\text{df,ds}}(T, \dot{\varepsilon}_{\text{II}})\right]\right),$$

incorporating plastic yielding, regularization for yielding, and a viscosity lower bound. This means that the composite viscosity is computed by the following sequence of steps:

"Viscosity = **U**pper bound → **Y**ielding → **W**eak zone → **L**ower bound"

and we refer to this sequence as the UYWL model of the composite viscosity.

2.7 The nonlinear viscosity model of choice

In this section, we formulate the model for the nonlinear viscosity that will be used in the remainder of this work as the constitutive relation in our mantle convection models. As in the composite viscosity model from Definition 2.6.4, we consider the (nonlinear) dislocation creep (2.27) in the upper mantle with $n > 1$. In the lower mantle, the nonlinear component in Equation (2.27) is eliminated by setting $n = 1$. To model the temperature dependence in the rheology, the diffusive component in (2.27) is replaced by the Arrhenius relationship from Definition 2.4.4 as in [93]. We describe the viscosity model in the following definition and illustrate it in Figure 2.3.

Definition 2.7.1 (Viscosity model)**.** Given lower and upper viscosity bounds $0 < \mu_{\min} < \mu_{\max} < \infty$, stress exponent $1 \leq n$, and yield strength $0 < \tau_{\text{yield}}$, the nonlinear viscosity is defined by

$$\mu(T, \dot{\varepsilon}_{\text{II}}) \coloneqq \max\left(\mu_{\min}, \min\left(\frac{\tau_{\text{yield}}}{2\dot{\varepsilon}_{\text{II}}}, w\left(\boldsymbol{x}\right)\min\left(\mu_{\max}, a\left(T\right)\,\dot{\varepsilon}_{\text{II}}^{\frac{1-n}{n}}\right)\right)\right) \tag{2.28}$$

This means that the viscosity is computed by the following sequence of steps:

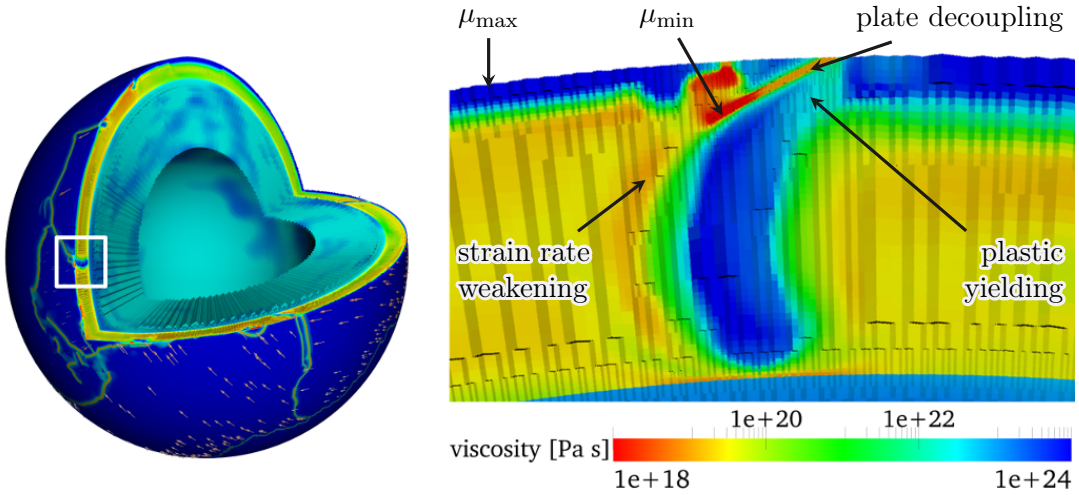


Figure 2.3: Cross section through a subducting slab (i.e., lithosphere subducting into mantle) of the Pacific plate showing effective mantle viscosity *(colors)* from Definition 2.7.1. Effective viscosity of plates reaches $\mu_{\max}$ *(dark blue)*, except at the hinge zone due to plastic yielding. In the thin plate boundary region, viscosity drops to $\mu_{\min}$ *(dark red)* creating a contrast of 10^6. Strain rate weakening reduces the viscosity underneath the plates in the asthenosphere and, combined with plastic yielding, is the reason for Earth's highly nonlinear rheology.

"Viscosity = **U**pper bound → **W**eak zone → **Y**ielding → **L**ower bound"

and we refer to this sequence as the UWYL model of the viscosity.

In Definition 2.7.1, both the stress exponent n as well as the scaling factor c_a in the Arrhenius relationship (2.24) take different values depending on the spatial location $\boldsymbol{x}$ being either in the lower or in the upper mantle. Moreover, note that even though the terms $\tau_{\text{yield}}/(2\dot{\varepsilon}_{\text{II}})$ and $\dot{\varepsilon}_{\text{II}}^{(1-n)/n}$ are unbounded as $\dot{\varepsilon}_{\text{II}} \to 0$, the viscosity (2.28) stays bounded because it incorporates the upper bound $\mu_{\max} < \infty$. In practice, we compute the viscosity as in Algorithm 2.7.1 and Table 2.3 lists typical values for the parameters of the constitutive relationship used in our numerical simulations.

Algorithm 2.7.1 Computing the viscosity μ from Equation (2.28)

Input fields: $T \in [0,1]$, $\dot{\varepsilon}_{\text{II}} \in [0,\infty)$, and $w \in (0,1]$
Input constants: $0 < \mu_{\min} < \mu_{\max} < \infty$, $0 < E_a, c_a, \tau_{\text{yield}}$, and $1 \leq n$
$a \leftarrow c_a \exp\left(E_a\left(0.5 - T\right)\right)$ ▷ compute temperature dependent component
$\mu \leftarrow a\,\dot{\varepsilon}_{\text{II}}^{\frac{1-n}{n}}$ ▷ compute strain-rate weakening component
$\mu \leftarrow \min\left(\mu, \mu_{\max}\right)$ ▷ apply upper bound
$\mu \leftarrow w\mu$ ▷ multiply in weak zone
if $\tau_{\text{yield}} < 2\mu\dot{\varepsilon}_{\text{II}}$ **then**
    $\mu \leftarrow \tau_{\text{yield}}/(2\dot{\varepsilon}_{\text{II}})$ ▷ apply yielding law
end if
$\mu \leftarrow \max\left(\mu_{\min}, \mu\right)$ ▷ apply lower bound

Table 2.3: Typical constitutive parameters for mantle convection.

Parameter	**Symbol**	**[Unit]**	**Value**
Activation energy	E_a	$[\mathrm{J\,mol^{-1}}]$	204×10^3
Viscosity scaling in lower mantle	$c_{a_{\mathrm{LM}}}$	–	4×10^5
Viscosity scaling in upper mantle	$c_{a_{\mathrm{UM}}}$	–	1×10^6
Minimum weak zone factor	$w_{\min}$	–	10^{-5}
Viscosity lower bound	$\mu_{\min}$	[Pa s]	10^{18}
Viscosity upper bound	$\mu_{\max}$	[Pa s]	10^{24}
Stress exponent	n	–	3
Yielding stress	τ_{yield}	[Pa]	123×10^6

2.8 Regularizing the nonlinear viscosity

Our goal is to use the viscosity model from Definition 2.7.1 in our mantle convection forward solver, where we will linearize the nonlinear differential equations with Newton's method. This requires to take the derivative of the viscosity μ in Equation (2.28) with respect to the second invariant of the strain rate $\dot{\varepsilon}_{\mathrm{II}}$, i.e., computing $\frac{\partial \mu}{\partial \dot{\varepsilon}_{\mathrm{II}}}$. The current form of the viscosity, presents three potential sources for discontinuities in this derivative: (i) applying the upper bound $\mu_{\max}$, (ii) applying the lower bound $\mu_{\min}$, and (iii) applying the yielding law. These discontinuities can cause convergence issues for Newton's method, therefore a regularized version of the viscosity model in Equation (2.28) is presented next.

When the upper bound $\mu_{\max}$ is applied to the viscosity (2.28), the nonlinear relationship between the second invariants of strain rate and viscous stress, which is originally $\tau_{\mathrm{II}}(\dot{\varepsilon}_{\mathrm{II}}) = 2a\dot{\varepsilon}_{\mathrm{II}}^{\frac{1}{n}}$, becomes linear: $\tau_{\mathrm{II}} = 2\mu_{\max}\dot{\varepsilon}_{\mathrm{II}}$. This switch can cause discontinuities in the derivative of the second invariant of the viscous stress, $\frac{\partial \tau_{\mathrm{II}}}{\partial \dot{\varepsilon}_{\mathrm{II}}}$, which implies a discontinuous viscosity derivative, $\frac{\partial \mu}{\partial \dot{\varepsilon}_{\mathrm{II}}}$. In order to obtain a continuously differentiable viscosity derivative, we employ a shift in the relationship between strain rate and viscous stress.

First, we want to find the value of the (second invariant of the) strain rate where the derivative of the nonlinear (second invariant of the) viscous stress matches the derivative of the linear viscous stress, which is a constant. This strain rate value will implicitly depend on the temperature component of the viscosity, $a = a\,(T)$. Thus we seek the solution of the problem:

$$\text{Find} \quad \dot{\varepsilon}_{\mathrm{deriv}} \quad \text{s.t.} \quad \frac{\partial \tau_{\mathrm{II}}}{\partial \dot{\varepsilon}_{\mathrm{II}}}\left(\dot{\varepsilon}_{\mathrm{deriv}}\right) = 2\mu_{\max}.$$

Let us postulate the equality

$$2\mu_{\max} \overset{!}{=} \frac{\partial \tau_{\mathrm{II}}}{\partial \dot{\varepsilon}_{\mathrm{II}}}\left(\dot{\varepsilon}_{\mathrm{II}}\right) = \frac{2}{n} a\, \dot{\varepsilon}_{\mathrm{II}}^{\frac{1-n}{n}}$$

and then solve for the strain rate to find

$$\dot{\varepsilon}_{\mathrm{deriv}} \leftarrow \dot{\varepsilon}_{\mathrm{II}} = \left(\frac{n\mu_{\max}}{a}\right)^{\frac{n}{1-n}}.$$

Second, we derive the minimum strain rate where the transition between linear and nonlinear viscous stress with matching derivatives occurs by solving the problem:

$$\text{Find} \quad \dot{\varepsilon}_{\min} \quad \text{s.t.} \quad \tau_{\text{II}}\left(\dot{\varepsilon}_{\text{deriv}}\right) = 2\mu_{\max}\dot{\varepsilon}_{\min}.$$

Therefore consider

$$2\mu_{\max}\dot{\varepsilon}_{\text{II}} \overset{!}{=} \tau_{\text{II}}\left(\dot{\varepsilon}_{\text{deriv}}\right) = 2a\,\dot{\varepsilon}_{\text{deriv}}^{\frac{1}{n}} = 2a\left(\frac{n\mu_{\max}}{a}\right)^{\frac{1}{1-n}},$$

which has the solution

$$\dot{\varepsilon}_{\min} \leftarrow \dot{\varepsilon}_{\text{II}} = \frac{\tau_{\text{II}}\left(\dot{\varepsilon}_{\text{deriv}}\right)}{2\mu_{\max}} = \frac{a}{\mu_{\max}}\left(\frac{n\mu_{\max}}{a}\right)^{\frac{1}{1-n}}.$$

Finally, we define the strain rate shift

$$d \coloneqq \dot{\varepsilon}_{\min} - \dot{\varepsilon}_{\text{deriv}}$$

and the shifted viscous stress $\tau_d\left(\dot{\varepsilon}_{\text{II}}\right) \coloneqq \tau_{\text{II}}\left(\dot{\varepsilon}_{\text{II}} - d\right) = 2\,a\left(\dot{\varepsilon}_{\text{II}} - d\right)^{\frac{1}{n}}$ that is now continuously differentiable at the transition between linear and nonlinear viscosity caused by the upper bound. This allows us to define a viscosity with regularized upper bound:

$$\mu_{\text{u}}(T, \dot{\varepsilon}_{\text{II}}) \coloneqq \min\left(a\left(T\right)\left(\dot{\varepsilon}_{\text{II}} - d\right)^{\frac{1}{n}}\dot{\varepsilon}_{\text{II}}^{-1}, \mu_{\max}\right) = \begin{cases} \mu_{\max} & \dot{\varepsilon}_{\text{II}} < \dot{\varepsilon}_{\min}, \\ a\left(T\right)\left(\dot{\varepsilon}_{\text{II}} - d\right)^{\frac{1}{n}}\dot{\varepsilon}_{\text{II}}^{-1} & \dot{\varepsilon}_{\min} \leq \dot{\varepsilon}_{\text{II}}. \end{cases} \tag{2.29}$$

Corollary 2.8.1. *The viscosity with regularized upper bound, $\mu_{\text{u}}(T, \dot{\varepsilon}_{\text{II}})$, in Equation* (2.29) *is continuously differentiable with respect to $\dot{\varepsilon}_{\text{II}}$.*

Proof. The derivative of the viscosity with regularized upper bound in Equation (2.29) for the case $\dot{\varepsilon}_{\min} \leq \dot{\varepsilon}_{\text{II}}$ is

$$\frac{\partial \mu_{\text{u}}}{\partial \dot{\varepsilon}_{\text{II}}}(T, \dot{\varepsilon}_{\text{II}}) = a\left(T\right)\left(\dot{\varepsilon}_{\text{II}} - d\right)^{\frac{1-n}{n}}\left(\frac{1}{n} - 1 + d\dot{\varepsilon}_{\text{II}}^{-1}\right)\dot{\varepsilon}_{\text{II}}^{-1}. \tag{2.30}$$

To show that the derivative (2.30) is continuous, it is sufficient to show the following:

$$\frac{1}{n} - 1 + d\dot{\varepsilon}_{\text{II}}^{-1} \to 0 \quad \text{as} \quad \dot{\varepsilon}_{\text{II}} \to \dot{\varepsilon}_{\min},\ \dot{\varepsilon}_{\min} \leq \dot{\varepsilon}_{\text{II}}.$$

Hence, we consider

$$\frac{1}{n} - 1 + d\dot{\varepsilon}_{\min}^{-1} = \frac{1}{n} - 1 + \frac{\dot{\varepsilon}_{\min} - \dot{\varepsilon}_{\text{deriv}}}{\dot{\varepsilon}_{\min}} = \frac{1}{n} - \frac{\dot{\varepsilon}_{\text{deriv}}}{\dot{\varepsilon}_{\min}}, \tag{2.31}$$

where we used the definition of d. Further, substituting the definitions of $\dot{\varepsilon}_{\text{deriv}}$ and $\dot{\varepsilon}_{\min}$ yields

$$\frac{\dot{\varepsilon}_{\text{deriv}}}{\dot{\varepsilon}_{\min}} = \left(\frac{n\mu_{\max}}{a}\right)^{\frac{n}{1-n}}\frac{\mu_{\max}}{a}\left(\frac{n\mu_{\max}}{a}\right)^{\frac{-1}{1-n}} = \frac{\mu_{\max}}{a}\left(\frac{n\mu_{\max}}{a}\right)^{-1} = \frac{1}{n}. \tag{2.32}$$

Combining Equations (2.31) and (2.32) gives the result. □

The yielding law that is incorporated in the viscosity model (2.28) is another source for a discontinuous derivative, $\frac{\partial \mu}{\partial \dot{\varepsilon}_{\text{II}}}$. We do not modify this feature of the viscosity. However, note that the application of the lower viscosity bound enforces that the viscosity is always bounded away from zero, i.e., $0 < \mu_{\min} \leq \mu$.

Finally, when the lower bound $\mu_{\min}$ is applied to the viscosity (2.28), we can simply introduce a regularization by avoiding the maximum

$$\mu \leftarrow \max\left(\mu_{\min}, \mu\right)$$

and instead applying the lower bound via addition:

$$\mu \leftarrow \mu_{\min} + \mu.$$

We summarize the derivations in the following Corollary 2.8.2, provide Algorithm 2.8.1 to describe the computation of the viscosity in practice, and illustrate the regularization in Figure 2.4.

Corollary 2.8.2 (Regularized viscosity model)**.** *With the same assumption as in Definition 2.7.1, the regularized nonlinear viscosity is*

$$\mu_{\text{reg}}(T, \dot{\varepsilon}_{\text{II}}) \coloneqq \mu_{\min} + \min\left(\frac{\tau_{\text{yield}}}{2\dot{\varepsilon}_{\text{II}}}, w\left(\boldsymbol{x}\right) \min\left(\mu_{\max}, a\left(T\right)\left(\dot{\varepsilon}_{\text{II}} - d\right)^{\frac{1}{n}} \dot{\varepsilon}_{\text{II}}^{-1}\right)\right), \tag{2.33}$$

where

$$d = \max\left(0, \dot{\varepsilon}_{\min} - \left(\frac{n\mu_{\max}}{a\left(T\right)}\right)^{\frac{n}{1-n}}\right) \quad \textit{and} \quad \dot{\varepsilon}_{\min} = \frac{a\left(T\right)}{\mu_{\max}}\left(\frac{n\mu_{\max}}{a\left(T\right)}\right)^{\frac{1}{1-n}}.$$

Algorithm 2.8.1 Computing the regularized viscosity μ_{reg} from Equation (2.33)

Input fields: $T \in [0,1]$, $\dot{\varepsilon}_{\text{II}} \in [0,\infty)$, and $w \in (0,1]$
Input constants: $0 < \mu_{\min} < \mu_{\max} < \infty$, $0 < E_a, c_a, \tau_{\text{yield}}$, and $1 \leq n$
$a \leftarrow c_a \exp\left(E_a\left(0.5 - T\right)\right)$ ▷ compute temperature dependent component
$\dot{\varepsilon}_{\min} \leftarrow \frac{a}{\mu_{\max}}\left(\frac{n\mu_{\max}}{a}\right)^{\frac{1}{1-n}}$ ▷ set min strain rate of nonlinear regime
$d \leftarrow \max\left(0, \dot{\varepsilon}_{\min} - \left(\frac{n\mu_{\max}}{a}\right)^{\frac{n}{1-n}}\right)$ ▷ set strain rate shift; enforce non-negativity
$\mu \leftarrow \begin{cases} a\left(\dot{\varepsilon}_{\text{II}} - d\right)^{\frac{1}{n}} \dot{\varepsilon}_{\text{II}}^{-1}, & \dot{\varepsilon}_{\min} < \dot{\varepsilon}_{\text{II}} \\ \infty, & \text{otherwise} \end{cases}$ ▷ compute strain-rate weakening component
$\mu \leftarrow \min\left(\mu, \mu_{\max}\right)$ ▷ apply upper bound
$\mu \leftarrow w\mu$ ▷ multiply in weak zone
if $\tau_{\text{yield}} < 2\mu\dot{\varepsilon}_{\text{II}}$ **then**
    $\mu \leftarrow \tau_{\text{yield}}/(2\dot{\varepsilon}_{\text{II}})$ ▷ apply yielding law
end if
$\mu \leftarrow \mu + \mu_{\min}$ ▷ apply lower bound

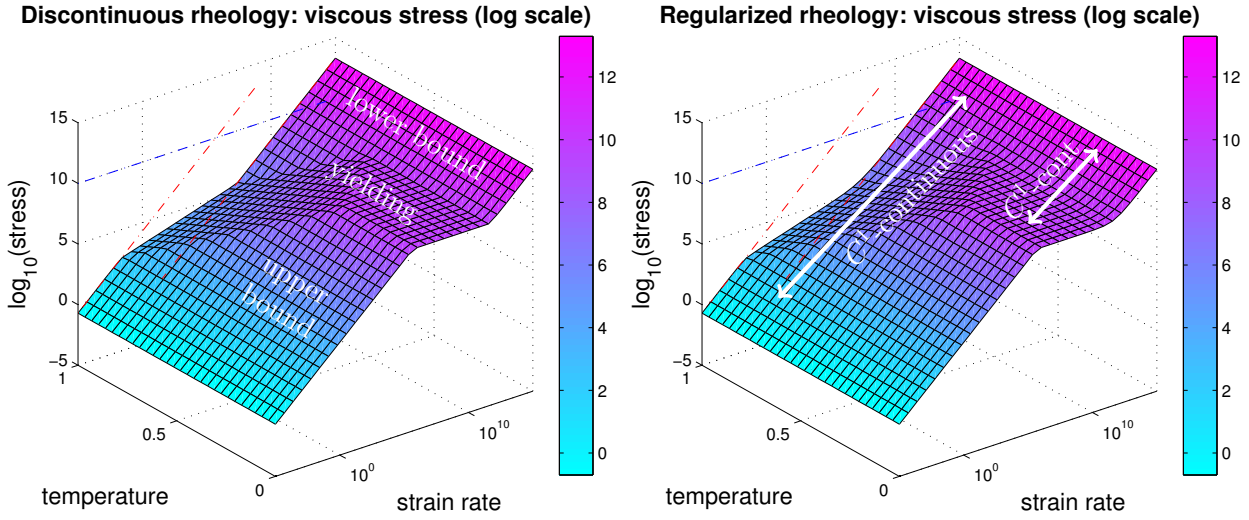


Figure 2.4: Temperature & strain rate vs. viscous stress relationships for viscosity model from Definition 2.7.1 *(left)* and its regularized version *(right)*. The regularized model incorporates upper and lower bounds for the viscosity in a continuously differentiable manner. The *white labels* in the left graph point out different regimes in the rheology, which form distinct planes in the graph.

3

Inverse Problem: Inferring Uncertain Parameters in Mantle Flow Models

This chapter describes the computational methods toward the goal of inferring uncertain parameters in Earth's mantle. We define the parameters that are of interest for the inversion and incorporate them into our mantle convection models. The available observational data that can be used in mantle convection inversions is described. Moreover, the observation operators and data misfit functionals are introduced. Finally, foreseeable challenges pertaining to the statistical inversion of parameters in Earth's mantle are presented and possible solutions are proposed. We begin with an introduction to inverse problems in a Bayesian framework, where the solution of a PDE establishes the connection between inversion parameters and observations.

3.1 Bayesian inverse problems governed by PDEs

Inverse problems are encountered where one makes indirect observations of a quantity of interest. In this dissertation this means that given observational data, we are interested in finding parameters of a physical model, where the observations from the model involve solving a PDE imposed by the model. The model PDE is also called forward problem. The particular model we are targeting is Earth's mantle convection introduced in Chapter 2. While forward problems are typically well posed by design, inverse problems are generally ill-posed since they do not satisfy the criteria for well-posedness: existence of a unique solution and continuous dependence of the observational data on parameter perturbations.

We consider inverse problems in a statistical setting, in particular we take the Bayesian approach toward inverse problems [31, 74, 112, 114]. In this approach, both the given observational data and the unknown model parameters are treated as random variables, and the solution of the Bayesian inverse problem (BIP) is a probability distribution for the parameters. Note that by restating the inverse problem as a well-posed extension in a larger space of probability distributions, we expose prior information on the parameters, while these would likely be hidden behind regularization schemes

in a deterministic setting [74].

To define a BIP, assume a finite-dimensional parameter space X and a finite-dimensional observation space Y. Let the state (or forward) variable $u = u(m) \in V$ be in a Hilbert space V and depend on parameters $m \in X$ through the solution of a model PDE referred to as the forward problem,

$$\mathcal{A}(m, u) = f, \quad \mathcal{A} : X \times V \to V', \quad f \in V', \tag{3.1}$$

where V' is the dual space of V and $\mathcal{A}(m, u)$ is generally considered to be nonlinear with respect to both m and u. In order to compare the solution of the forward model $u(m)$ to observational data in Y, assume the existence of an observation operator $\mathcal{F} : X \times V \to Y$. The observation operator implicitly depends on the model PDE (3.1) through the dependence of $u(m)$ on the parameters and it can, in general, be nonlinear with respect to both of its arguments. The observational data $d \in Y$ is given and we assume that it contains normally distributed additive noise such that

$$\big(d - \mathcal{F}(m, u(m))\big) \sim \mathcal{N}(0, \mathscr{C}_{\text{noise}})$$

with covariance $\mathscr{C}_{\text{noise}}$. Moreover, the prior distribution of the parameters is assumed to be Gaussian, $\mathcal{N}(m_{\text{pr}}, \mathscr{C}_{\text{pr}})$, with mean $m_{\text{pr}} \in X$ and covariance $\mathscr{C}_{\text{pr}}$. As a consequence of these assumptions and of properties of conditional probability densities, we obtain the posterior density from Bayes' formula

$$\pi_{\text{post}}(m) \coloneqq \pi\left(m \mid d\right) = \frac{\pi\left(d \mid m\right) \pi\left(m\right)}{\pi\left(d\right)}. \tag{3.2}$$

Finally, if we incorporate that noise, represented by likelihood $\pi\left(d \mid m\right)$, and prior, represented by prior density $\pi\left(m\right)$, are normally distributed, then Equation (3.2) becomes

$$\pi_{\text{post}}(m) \propto \exp\left(-\frac{1}{2}\left\|d - \mathcal{F}(m, u(m))\right\|^2_{\mathscr{C}^{-1}_{\text{noise}}} - \frac{1}{2}\left\|m - m_{\text{pr}}\right\|^2_{\mathscr{C}^{-1}_{\text{pr}}}\right). \tag{3.3}$$

In this framework, the major computational challenge lies in exploring the posterior, which typically is defined over a high-dimensional space due to large numbers of parameters. In order to characterize π_{post}, our first goal will be to find an estimate for the maximum point of π_{post} and the second goal will be to approximate π_{post} in a neighborhood around this maximum, which amounts to a local Gaussian approximation of the posterior, sometimes referred to as Laplace approximation.

The maximum a posteriori (MAP) estimate of the posterior density is defined by

$$m_{\text{MAP}} \coloneqq \arg\max_{m \in X} \pi_{\text{post}}(m),$$

which can be rewritten as a minimization problem governed by the model PDE (3.1):

$$m_{\text{MAP}} = \arg\min_{m \in X} \mathcal{J}(m, u(m)) \quad \text{subject to} \quad \mathcal{A}(m, u(m)) = f, \tag{3.4}$$

where the objective functional $\mathcal{J}$ is proportional to the negative log of π_{post},

$$\mathcal{J}(m, u(m)) \coloneqq \frac{1}{2}\left\|d - \mathcal{F}(m, u(m))\right\|^2_{\mathscr{C}^{-1}_{\text{noise}}} + \frac{1}{2}\left\|m - m_{\text{pr}}\right\|^2_{\mathscr{C}^{-1}_{\text{pr}}}. \tag{3.5}$$

We aim to solve the minimization problem (3.4) for m_{MAP} with Newton's method, which requires derivatives (gradients and Hessians). These operations will be performed in a scalable way using adjoint methods [15], which are described in Section 3.2.

Once the maximum a posteriori estimate m_{MAP} is computed, a Gaussian approximation about m_{MAP} can be used to quantify parameter uncertainties. This is a tractable approach to explore statistical properties of posteriors that correspond to computationally expensive forward models [24], as opposed to, e.g., sampling techniques that approximate the full non-Gaussian posterior. Furthermore, these Gaussian approximations can be used to accelerate sampling algorithms [82, 91]. To obtain a Gaussian approximation of the posterior, we linearize $\mathcal{F}$ at m_{MAP}. Then the resulting Gaussian measure of the posterior is associated to the normal distribution $\mathcal{N}(m_{\text{MAP}}, \mathscr{C}_{\text{post}})$ with mean m_{MAP} and covariance matrix given by

$$\mathscr{C}_{\text{post}} \coloneqq \left(\boldsymbol{F}^* \mathscr{C}_{\text{noise}}^{-1} \boldsymbol{F} + \mathscr{C}_{\text{pr}}^{-1}\right)^{-1}, \tag{3.6}$$

where $\boldsymbol{F}$ is the Fréchet derivative of the observation operator $\mathcal{F}$ evaluated at m_{MAP} and $\boldsymbol{F}^*$ is denoting its adjoint with respect to a suitable inner product on X.

Explicitly constructing the posterior covariance matrix (3.6) is prohibitively expensive in case of large numbers of parameters, because of storage requirements of the generally dense $\mathscr{C}_{\text{post}}$ and since it would require as many solves of the incremental forward and adjoint problems as there are parameters. However, a low-rank approximation of $\mathscr{C}_{\text{post}}$ is often possible [48], in which the computational cost corresponds to the amount of information encoded in the data.

3.2 Gradient and Hessian computation with adjoint methods

When solving the nonlinear optimization problem (3.4) for the MAP estimate, we need to compute gradients and Hessians of the objective functional (3.5) if we want to use Newton's method. To compute the gradient of (3.5) with respect to m, we define the Lagrangian for the gradient as the sum of $\mathcal{J}(m, u)$ and the variational form of the residual of the forward problem (3.1),

$$\mathcal{L}_g(m, u, v) \coloneqq \mathcal{J}(m, u) + \left[\left(\mathcal{A}(m, u)\,, v\right) - \left(f\,, v\right)\right].$$

We take variations of the Lagrangian with respect to v, u, and m and set the former two to zero, i.e.,

$$\delta_v[\mathcal{L}_g](\tilde{v}) \stackrel{!}{=} 0, \quad \delta_u[\mathcal{L}_g](\tilde{u}) \stackrel{!}{=} 0, \quad \delta_m[\mathcal{L}_g](\tilde{m}),$$

where, e.g., $\delta_v[\mathcal{L}_g](\tilde{v})$ denotes a variation with respect to v in direction $\tilde{v}$ defined by

$$\delta_v[\mathcal{L}_g](\tilde{v}) = \delta_v[\mathcal{L}_g(m, u, v)](\tilde{v}) \coloneqq \left.\frac{\partial \mathcal{L}_g(m, u, v + \epsilon\tilde{v})}{\partial \epsilon}\right|_{\epsilon=0}.$$

This results in a sequence of three steps to compute the gradient. First, we solve the (generally nonlinear) forward problem for u:

$$\left(\mathcal{A}(m, u)\,, \tilde{v}\right) = \left(f\,, \tilde{v}\right) \quad \text{for all } \tilde{v} \in V, \tag{3.7}$$

and then the linear adjoint problem for v:

$$(\delta_u[\mathcal{A}](\tilde{u})\,,\,v) = (\delta_u[\mathcal{F}](\tilde{u})\,,\,d-\mathcal{F}(m,u))_{\mathscr{C}_{\text{noise}}^{-1}} \quad \text{for all } \tilde{u}\in V. \tag{3.8}$$

After computing the state u and the adjoint state v, the gradient is found to be

$$\mathcal{G}(\tilde{m}) = (\delta_m[\mathcal{A}](\tilde{m})\,,\,v) + (\tilde{m}\,,\,m-m_{\text{pr}})_{\mathscr{C}_{\text{pr}}^{-1}} - (\delta_m[\mathcal{F}](\tilde{m})\,,\,d-\mathcal{F}(m,u))_{\mathscr{C}_{\text{noise}}^{-1}} \quad \text{for } \tilde{m}\in X. \tag{3.9}$$

We repeat the Lagrange multiplier approach to find expressions for applying the Hessian to a vector. The Lagrangian for the Hessian is the sum of the gradient (3.9) and the variational forms of the residuals of the forward (3.7) and adjoint (3.8) problems, respectively,

$$\begin{aligned}\mathcal{L}_H(m,u,v,\tilde{m},\tilde{u},\tilde{v}) \coloneqq \mathcal{G}(m,u,v,\tilde{m}) &+ [(\mathcal{A}(m,u)\,,\,\tilde{v}) - (f\,,\,\tilde{v})] \\ &+ \left[(\delta_u[\mathcal{A}](\tilde{u})\,,\,v) - (\delta_u[\mathcal{F}](\tilde{u})\,,\,d-\mathcal{F}(m,u))_{\mathscr{C}_{\text{noise}}^{-1}}\right].\end{aligned}$$

Similar to the procedure for the gradient, to apply the Hessian to a direction $\tilde{m}$, we perform the steps

$$\delta_v[\mathcal{L}_H](\hat{v}) \stackrel{!}{=} 0, \quad \delta_u[\mathcal{L}_H](\hat{u}) \stackrel{!}{=} 0, \quad \delta_m[\mathcal{L}_H](\hat{m}),$$

Hence, we first need to solve the linear incremental forward problem for $\tilde{u}$:

$$(\delta_u[\mathcal{A}](\tilde{u})\,,\,\hat{v}) = -\,(\delta_m[\mathcal{A}](\tilde{m})\,,\,\hat{v}) \quad \text{for all } \hat{v}\in V,$$

and then the linear incremental adjoint problem for $\tilde{v}$:

$$\begin{aligned}(\delta_u[\mathcal{A}](\hat{u})\,,\,\tilde{v}) = &-(\delta_u\delta_u[\mathcal{A}](\tilde{u})(\hat{u})\,,\,v) \\ &+(\delta_u\delta_u[\mathcal{F}](\tilde{u})(\hat{u})\,,\,d-\mathcal{F}(m,u))_{\mathscr{C}_{\text{noise}}^{-1}} \\ &-(\delta_u[\mathcal{F}](\tilde{u})\,,\,\delta_u[\mathcal{F}](\hat{u}))_{\mathscr{C}_{\text{noise}}^{-1}} \\ &-(\delta_u\delta_m[\mathcal{A}](\tilde{m})(\hat{u})\,,\,v) \\ &+(\delta_u\delta_m[\mathcal{F}](\tilde{m})(\hat{u})\,,\,d-\mathcal{F}(m,u))_{\mathscr{C}_{\text{noise}}^{-1}} \\ &-(\delta_m[\mathcal{F}](\tilde{m})\,,\,\delta_u[\mathcal{F}](\hat{u}))_{\mathscr{C}_{\text{noise}}^{-1}} \quad \text{for all } \hat{u}\in V.\end{aligned}$$

After computing the so-called incremental state $\tilde{u}$ and the incremental adjoint state $\tilde{v}$, the Hessian application to $\tilde{m}$ is computed as

$$\begin{aligned}\mathcal{H}(\tilde{m})(\hat{m}) = &(\delta_m[\mathcal{A}](\hat{m})\,,\,\tilde{v}) + (\tilde{m}\,,\,\hat{m})_{\mathscr{C}_{\text{pr}}^{-1}} \\ &+(\delta_m\delta_u[\mathcal{A}](\tilde{u})(\hat{m})\,,\,v) \\ &-(\delta_m\delta_u[\mathcal{F}](\tilde{u})(\hat{m})\,,\,d-\mathcal{F}(m,u))_{\mathscr{C}_{\text{noise}}^{-1}} \\ &+(\delta_u[\mathcal{F}](\tilde{u})\,,\,\delta_m[\mathcal{F}](\hat{m}))_{\mathscr{C}_{\text{noise}}^{-1}} \\ &+(\delta_m\delta_m[\mathcal{A}](\tilde{m})(\hat{m})\,,\,v) \\ &-(\delta_m\delta_m[\mathcal{F}](\tilde{m})(\hat{m})\,,\,d-\mathcal{F}(m,u))_{\mathscr{C}_{\text{noise}}^{-1}} \\ &+(\delta_m[\mathcal{F}](\tilde{m})\,,\,\delta_m[\mathcal{F}](\hat{m}))_{\mathscr{C}_{\text{noise}}^{-1}} \quad \text{for } \hat{m}\in X.\end{aligned} \tag{3.10}$$

Corollary 3.2.1 (Simplified observation operator)**.** *If the observation operator $\mathcal{F}(m, u(m))$ is linear in $u(m)$ and independent of m, the equations for the gradient computation and Hessian application simplify to the following. The forward and adjoint problems are*

$$(\mathcal{A}(m,u)\,,\tilde{v}) = (f\,,\tilde{v}) \quad \textit{for all } \tilde{v} \in V,$$

$$(\delta_u[\mathcal{A}](\tilde{u})\,,v) = (\mathcal{F}\tilde{u}\,,d-\mathcal{F}u)_{\mathscr{C}_{\text{noise}}^{-1}} \quad \textit{for all } \tilde{u} \in V,$$

and the gradient is

$$\mathcal{G}(\tilde{m}) = (\delta_m[\mathcal{A}](\tilde{m})\,,v) + (\tilde{m}\,,m-m_{\text{pr}})_{\mathscr{C}_{\text{pr}}^{-1}} \quad \textit{for } \tilde{m} \in X.$$

The incremental forward and adjoint problems are

$$(\delta_u[\mathcal{A}](\tilde{u})\,,\hat{v}) = -\,(\delta_m[\mathcal{A}](\tilde{m})\,,\hat{v}) \quad \textit{for all } \hat{v} \in V,$$

$$(\delta_u[\mathcal{A}](\hat{u})\,,\tilde{v}) = -\,(\delta_u\delta_u[\mathcal{A}](\tilde{u})(\hat{u})\,,v) - (\mathcal{F}\tilde{u}\,,\mathcal{F}\hat{u})_{\mathscr{C}_{\text{noise}}^{-1}} - (\delta_u\delta_m[\mathcal{A}](\tilde{m})(\hat{u})\,,v) \quad \textit{for all } \hat{u} \in V.$$

The Hessian in direction $\tilde{m}$ is

$$\begin{aligned}\mathcal{H}(\tilde{m})(\hat{m}) = {}& (\delta_m[\mathcal{A}](\hat{m})\,,\tilde{v}) + (\tilde{m}\,,\hat{m})_{\mathscr{C}_{\text{pr}}^{-1}} + \\ & (\delta_m\delta_u[\mathcal{A}](\tilde{u})(\hat{m})\,,v) + (\delta_m\delta_m[\mathcal{A}](\tilde{m})(\hat{m})\,,v) \quad \textit{for } \hat{m} \in X.\end{aligned}$$

3.3 Review of developments in inverse problems applied to mantle convection

Driving and resisting forces for plates at subduction zones are evaluated with inverse models in [50]. There, an inversion for the long-wavelength distribution of mantle buoyancy is performed using plate motion as data and with a radial viscosity. However, the essential character of slabs as stress guides while resisting plate motion through bending is not included. The inclusion is possible by incorporating constitutive relationships with thermally activated diffusion and dislocation creep and plastic yielding, which gives rise to essential processes within the bending plates with the result of producing plate-like motion at the surface [5, 6, 110].

Adjoint methods for the inference of mantle properties and initial temperature distributions in the mantle are developed in [72, 80, 109] for regional models and in [25, 67] for global models. All of these models rely on simplified rheologies that do not incorporate essential nonlinear physics of strain rate weakening and plastic yielding.

Inverse problems with a strain rate weakening rheology are solved in [119]. There, the setup of the problems consists of two- and three-dimensional cross sectional domains and includes simple subduction models with a single slab. Through solving deterministic inverse problems with Tikhonov or total variation (TV) regularization, constant parameters and spatially-varying fields in the rheology are recovered. The instantaneous nonlinear mantle convection forward problem is solved with Newton's method, whereas quasi-Newton is used as an inverse solver.

Building on results from [119], a strain rate weakening rheology with plastic yielding is considered in [93]. In a two-dimensional Cartesian domain with multiple slabs, inversions for plate boundary strength and constant rheological parameters are performed. Manufactured plate velocities at the surface serve as observational data. The inverse problems are solved deterministically and also within a Bayesian framework, where uncertainties in the parameters are quantified with Gaussian approximations about the MAP estimate and with Markov Chain Monte Carlo (MCMC) sampling methods.

3.4 Parametrization of the mantle convection model

We are interested in globally constant as well as locally varying parameters in the constitutive relationship (2.28). The global parameters are constants affecting the viscosity in the upper mantle, which is ∼660 km deep and includes the lithosphere and the asthenosphere. These constants are the scaling factor $\mu_{\text{UM}} > 0$, the stress exponent $n \geq 1$, and the yield stress $\tau_{\text{yield}} > 0$. Additionally, we aim to infer local spatially-varying parameters, which describe plate boundary strength. The coupling strength between plates is modeled through a plate decoupling factor in the viscosity (2.28). The decoupling is performed by prescribing thin regions of low viscosity between plates, called weak zones $0 < w(\boldsymbol{x}) \leq 1$, $\boldsymbol{x} \in \Omega$, see Section 2.5.

We transform the rheological and geometrical parameters of the mantle convection model into inversion parameters that are collected in the vector $\mathbf{m} \coloneqq [\mathbf{m}_{\text{glo}}, \mathbf{m}_{\text{weak}}]$. The global rheological parameters are related to the inversion parameters $\mathbf{m}_{\text{glo}} \coloneqq [m_1, m_2, m_3] \in \mathbb{R}^3$ through a parametrization that enforces lower bounds:

$$\mu_{\text{UM}} = \mu_{\text{UM}}(m_1) = \exp(m_1), \quad n = n(m_2) = 1 + \exp(m_2), \quad \tau_{\text{yield}} = \tau_{\text{yield}}(m_3) = \exp(m_3).$$

The spatially varying weak zones are associated with the parameter vector $\mathbf{m}_{\text{weak}} \coloneqq [m_i]_i$, $m_i \in \mathbb{R}$, of dimension greater or equal one. To parametrize the weak zone from Definition 2.5.1, we extend the definition of the minimum factor of the weak zone to

$$w_{\min}(\mathbf{m}_{\text{weak}}, \boldsymbol{x}) \coloneqq \exp\left(-\left(\sum_i m_i\, \theta_i(\boldsymbol{x})\right)^2\right) \in (0, 1], \quad \boldsymbol{x} \in \Omega. \tag{3.11}$$

Here, the functions $\theta_i : \Omega \to [0, 1]$ are indicator functions that are one near associated plate boundary manifolds and zero otherwise. For example in the simplest case, $\theta_i(\boldsymbol{x})$ for each i is discontinuous and equals one if $\boldsymbol{x}$ is a near plate boundary i. Alternatively, if a finer resolution of the weak zone factors is desired, the functions θ_i can represent components of a piecewise linear curve along plate boundaries. In that case, the number of indices i, and with it the dimension of $\mathbf{m}_{\text{weak}}$, increases. Finally, using the weak zone factor $w_{\min}$ from Equation (3.11), the parametrized weak zone is modeled as in Equation (2.25),

$$w\,(\boldsymbol{x}) = w\,(\mathbf{m}_{\text{weak}}, \boldsymbol{x}) = 1 - (1 - w_{\min}(\mathbf{m}_{\text{weak}}, \boldsymbol{x})) \exp\left(-\frac{\xi(\boldsymbol{x})^2}{2\sigma^2}\right), \quad \boldsymbol{x} \in \Omega. \tag{3.12}$$

3.5 Observational data and observation operators

The available observational data that we can fit to output from our instantaneous mantle convection model includes: *(i)* plate motion, *(ii)* plate deformation, *(iii)* average upper mantle viscosity, and *(iv)* topography data.

As plate motion data, one angular velocity vector (also called Euler vector or Euler pole) is given per plate. It describes the rotation of the plate, which is assumed to be a rigid body, on the surface of a sphere about the corresponding Euler pole. This data is obtained from seafloor spreading rates based on magnetic anomalies as well as from Global Positioning System (GPS) station velocities. Our main data source is a global plate motion model called NNR-MORVEL56 [7], which includes 56 plates and corresponding angular velocities that are given in a no-net-rotation (NNR) frame. The plate boundaries and velocities of the NNR-MORVEL56 data set are illustrated in Figure 3.1.

Plate deformation data is gathered with dense GPS networks as displacement data and processed into deformations within plates in form of horizontal velocity fields [77]. These displacements are also influenced by short-term effects from seismic cycles, which are not appropriate to be considered for mantle convection models. Therefore, short-term displacements have to be removed from the data. Another form of observations of plate deformations at the surface comes from strain rate models. These can be incorporated as a data misfit for the second invariant of the strain rate at the surface. However, the available data sets are based on older plate models that do not include important small plates [14]. This may render the strain rate data as not suitable for the inversion using our mantle model.

Information about average upper mantle viscosities exists for certain regions of the upper mantle where one can infer an average viscosity from post-glacial rebound of the surface topography [32, 60, 88, 107].

Finally, Earth's topography is influenced by plate dynamics [64, 121]. One use of such topography data would be possible using a free surface in the model. This approach is outside the scope of this work. However, as an alternative the topography data can be compared to a dynamic topography from the model, which is derived from the surface normal stress. Whether fitting the dynamic topography to the topography data is a feasible approach in an inverse problem is ongoing research.

After reviewing the available data, we discuss how to obtain observations from the model and compare it to the data. The motion of a single plate is modeled as motion of a rigid body on a sphere, therefore the observational data is an angular velocity vector $\mathbf{r}_{\text{plate}} \in \mathbb{R}^3$. This vector is transformed into a spatially distributed field at the surface, $\boldsymbol{u}_{\text{obs}}(\boldsymbol{x}) \coloneqq \mathbf{r}_{\text{plate}} \times \boldsymbol{x}$ for $\boldsymbol{x} \in \Gamma_{\text{plate}} \subseteq \Gamma_{\text{surf}}$ and $\boldsymbol{u}_{\text{obs}}(\boldsymbol{x}) = 0$ for $\boldsymbol{x} \notin \Gamma_{\text{plate}}$, where Γ_{plate} is the area of Earth's surface encompassing plate-like (i.e., rigid) motion. This process is repeated for each plate. Let $\mathscr{C}_{\text{plate}}$ be a suitable covariance operator that provides sufficient smoothing, for instance a Laplacian (see [112]), and let $\Pi_{\text{surf}} : \Omega \to \Gamma_{\text{surf}}$ restrict a volume field to the surface, then we define the data misfit functional

$$\int_{\Gamma_{\text{plate}}} \big(\boldsymbol{u}_{\text{obs}}(\boldsymbol{x}) - \Pi_{\text{surf}}\, \boldsymbol{u}(\boldsymbol{x})\big)\mathscr{C}_{\text{plate}}^{-1}\big(\boldsymbol{u}_{\text{obs}}(\boldsymbol{x}) - \Pi_{\text{surf}}\, \boldsymbol{u}(\boldsymbol{x})\big)\, \mathrm{d}\boldsymbol{x}. \tag{3.13}$$

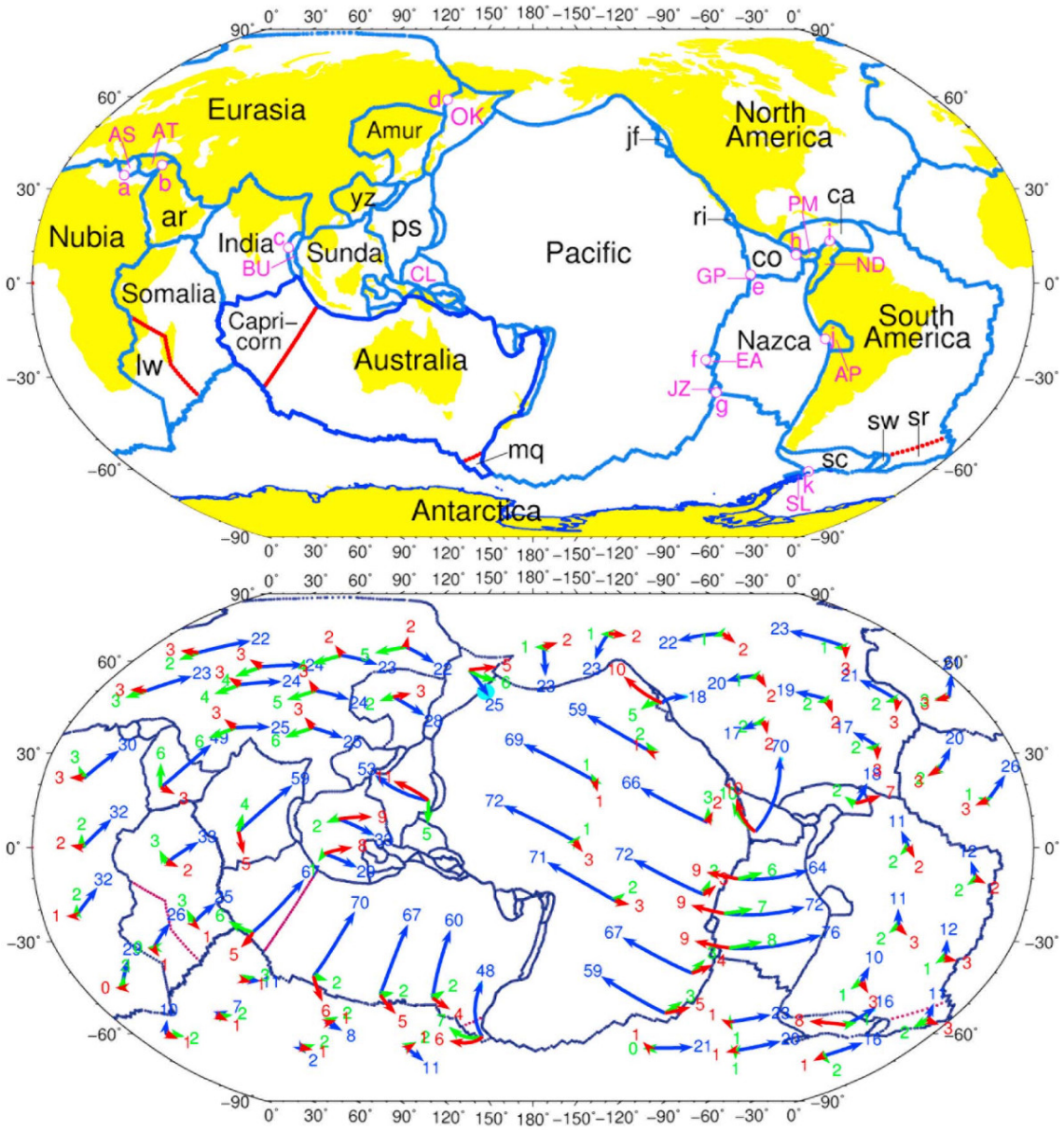


Figure 3.1: *Top:* Plate boundaries *(blue & red lines)* and plate geometries employed for NNR-MORVEL56 plate motion data set [7]. *Bottom:* The *blue arrows* and *numerals* depict the horizontal plate velocities in $\mathrm{mm\,yr^{-1}}$; *green and red arrows* show deviations from previous data sets. (Credit: Argus, Gordon, DeMets).

In addition to plate-like motion as in (3.13), plate deformation observations are available. For certain regions of the earth, where plates are not moving rigidly, plate deformations can be modeled in the form of displacement vectors. This data provides further observations of the velocity at the surface, complementing rigid plate motion data. It consists of spatially distributed fields and a similar data misfit functional as (3.13) can be used.

For a region with data about the average upper mantle viscosity from post-glacial rebound observations, we assume the average viscosity $\mu_{\text{rebound}} > 0$ to be given for a column of the upper mantle, $\Omega_{\text{rebound}} \subseteq \Omega$. Let $\gamma_{\text{rebound}} > 0$ be the misfit weight, then we define the data misfit functional based on a geometric average

$$\gamma_{\text{rebound}}^{-1} \left(\mu_{\text{rebound}} - \exp\left(\int_{\Omega_{\text{rebound}}} \frac{\ln\big(\mu(\mathbf{m}, \dot{\varepsilon}_{\text{II}}(\boldsymbol{u}))\big)}{|\Omega_{\text{rebound}}|} \, \mathrm{d}\boldsymbol{x} \right) \right)^2 . \tag{3.14}$$

Note that the resulting observation operator depends on $\mathbf{m}$ and $\boldsymbol{u}$ and is nonlinear with respect to both arguments.

To compute the normal stress acting on Earth's surface, we take the residual of the momentum equation of the Stokes system, where we apply the viscous stress operator A without enforcing Dirichlet boundary conditions, denoted by $A_{\text{Dir}-\text{free}}$. Thus the normal stress at the surface is computed as

$$s_n(\mathbf{m}, \boldsymbol{u}) \coloneqq \boldsymbol{n} \cdot \Pi_{\text{surf}} \left(A_{\text{Dir}-\text{free}}(\mathbf{m}, \boldsymbol{u}) + B^* p - \boldsymbol{f} \right),$$

with projector Π_{surf} defined as above. Given topography observations $t_n(\boldsymbol{x})$, $\boldsymbol{x} \in \Gamma_{\text{topo}}$ on sub-regions of the surface $\Gamma_{\text{topo}} \subseteq \Gamma_{\text{surf}}$, we define the data misfit functional

$$\int_{\Gamma_{\text{topo}}} \big(t_n(\boldsymbol{x}) - c_{\text{topo}}\, s_n(\mathbf{m}, \boldsymbol{u})\big) \mathscr{C}_{\text{topo}}^{-1} \big(t_n(\boldsymbol{x}) - c_{\text{topo}}\, s_n(\mathbf{m}, \boldsymbol{u})\big) \, \mathrm{d}\boldsymbol{x}, \tag{3.15}$$

with a suitable covariance operator $\mathscr{C}_{\text{topo}}$ (e.g., a Laplacian) and an appropriate scaling $c_{\text{topo}} \in \mathbb{R}$ to generate the dynamic topography from the normal surface stress.

3.6 Inversion challenges and inverse solver challenges

This section outlines the potential challenges toward the goal to infer uncertain parameters in global mantle convection models. We also propose ideas to tackle those challenges.

In Section 3.5, we have seen that observational data is only available at Earth's surface and, additionally, it is often sparse, e.g., the plate velocity data provides only one velocity vector per plate. This limited amount of observational data poses a challenge on the choice of parameters within mantle models. Namely, we want to find a set of parameters that is sufficiently well informed by the data, i.e., we want the posterior distribution to have a sufficiently narrow variance. Initial inversions could be carried out using constant weak zone factors (3.11) for each plate boundary. Subsequently, it is of interest whether the observational data is sufficiently informative to allow for higher fidelity in the parameters, e.g., piecewise linear weak zone factors with variability within a plate boundary.

Furthermore, related to the parameters, is the definition of the prior. We propose to assume a set of discrete, uncorrelated, finite-dimensional parameters equipped with a Gaussian prior informed by knowledge from geophysics. In cases of higher-resolution weak zone factors, sufficient smoothing has to be incorporated into the prior in order to arrive at a well-defined BIP. Similarly as for the prior, the covariance operators in the misfit functionals in Section 3.5 have to exhibit sufficient smoothing in the cases of spatially distributed observation fields.

A significant computational challenge is the cost for solving the BIP for the MAP estimate (3.4). One aspect of this is to find an effective preconditioner for the Hessian of the objective functional (3.5) associated to the MAP estimate. The typical approach is to use the prior as a preconditioner, but this can turn out to be insufficient for certain problems. It is an open research question how a more effective Hessian preconditioner can be constructed.

Another aspect pertaining to the computational cost is the solution of forward/adjoint and incremental forward/adjoint equations (see Section 3.1). We would like to reduce the accuracy of these solves and adjust the inexactness based on progress of the "outer" solver for the MAP estimate. As a consequence, we propose to use an inexact Newton–CG method for the MAP estimate. This method requires gradients (3.9) and Hessian-vector products (3.10) of the objective functional (3.5), which, in turn, are found by inexact "inner" solves for forward/adjoint and incremental forward/adjoint variables, respectively. These inner solves allow for inexactness at several levels. First, the Hessian-vector products can be performed inexactly to reduce the computational cost [2, 3, 66], which is analyzed in detail in [63]. These ideas are connected to the broader question of inexactness in Krylov methods [106, 117]. Second, the gradient computation can be performed inexactly [61, 79], which may enable a significant reduction in cost because it involves the nonlinear forward solution of the mantle convection problem.

4

Stokes Solver for the Forward Problem

This chapter begins with the presentation of our numerical methods to solve the mantle convection forward problem introduced in Chapter 2. Subsequent chapters will focus on specific parts of the nonlinear and linear solvers and preconditioners that are crucial for the parallel, scalable solution of the forward problem of global nonlinear mantle convection.

Earth's mantle convection is one of a large number of complex PDE problems that require implicit solution on extreme-scale systems. The complexity arises from the presence of a wide range of length scales and strong heterogeneities, as well as localizations and anisotropies. Complex PDE problems often require aggressive adaptive mesh refinement, such as that provided by the parallel forest-of-octree library p4est [30]. They also often require advanced discretizations, such as the high-order, hanging-node, mixed continuous-velocity/discontinuous-pressure element pair employed here. The physics complexities combined with the discretization complexities conspire to present enormous challenges for the design of solvers that are not only algorithmically optimal, but also scale well in parallel. These challenges are well documented in a number of blue ribbon panel reports (e.g., [40]).

In our context of time-independent non-Newtonian flows, *implicit solvers* means a combination of nonlinear and linear solvers and preconditioners. We employ Newton's method as our nonlinear solver. It can deliver asymptotic quadratic convergence, independent of problem size, for many problems. However, differentiating complex constitutive laws such as (2.33) to obtain the linearized Newton operator creates an even more complex system to be solved. Combining the Newton method with an appropriately truncated Krylov linear solver permits avoidance of *oversolving* far from the region of fast convergence [42]. The crucial point is then the preconditioner within Krylov, which must simultaneously globalize information to maximize algorithmic efficiency, while localizing it to maximize parallel performance. For preconditioning, we target multilevel solvers, which are algorithmically optimal for many problems (i.e., they require $\mathcal{O}(n)$ work, where n is the number of unknowns) and parallelize well (requiring $\mathcal{O}(\log n)$ depth), at least for simple elliptic PDE operators.

When $\mathcal{O}(10^5)$ cores and beyond are needed for implicit solution of such complex PDE problems, the usual approach has been to retreat to algorithmically suboptimal but easily-parallelizable solvers (such as explicit or simply-preconditioned implicit). This is clearly not a tenable situation, and the

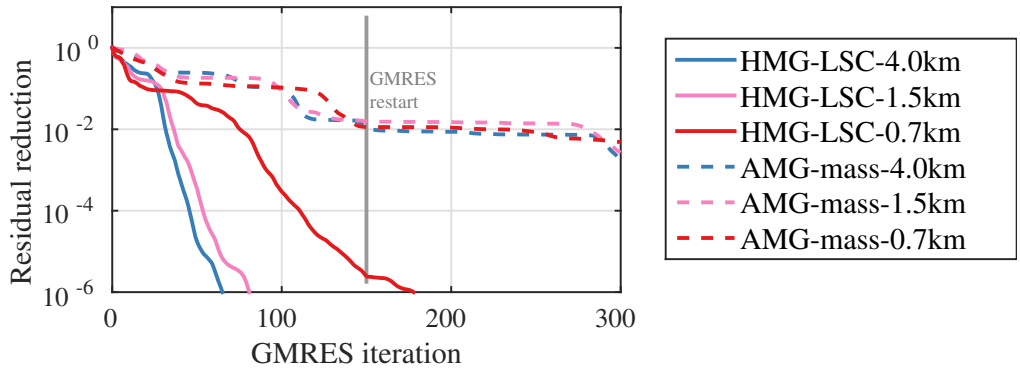


Figure 4.1: Comparison of algorithmic performance of previous state of the art *(dashed lines)* vs. new *(solid lines)* Stokes solver for a sequence of increasingly difficult problems (indicated by *colors*), reflecting increasingly narrower plate boundary regions.

performance gap between optimal and suboptimal solvers only increases as problems grow larger. Thus our goal here is to present an implicit solver (going beyond [26, 27, 113]) that delivers optimal algorithmic complexity while scaling with high parallel efficiency to the full size of leadership-class supercomputers for the class of complex PDE problems targeted here, with particular application to our driving global mantle convection problem.

Figure 4.1 illustrates the power of algorithmically optimal solvers for our mantle convection problem. The curves show the reduction in residual as a function of Krylov iterations for a sequence of increasingly difficult problems *(different colors)*. The *dashed curves* represent a contemporary, well-regarded solver, such as that found in the community mantle convection code ASPECT [78]. This combines algebraic multigrid (AMG) to precondition the (1,1) block of the Stokes system along with a diagonal mass matrix approximation of the (2,2) Schur complement. Our new solver combines a sophisticated hybrid spectral–geometric–algebraic multigrid (HMG) along with a novel HMG-preconditioned improved Schur complement approximation. The massive enhancement in algorithmic performance (over 4 orders of magnitude lower residual for the same number of iterations) seen in the figure is due to the improvement of the Schur complement. This is what makes the solution of the high-fidelity mantle flow models we are targeting tractable. It increases however the algorithmic complexity, but as we will see in Chapter 8, we are still able to obtain excellent scalability out to 1.6M cores, to go with the several orders of magnitude improvement in run time. Key to achieving this scalability is: Avoiding AMG setup/communication costs with a spectral and geometric multigrid approach; eliminating AMG's requirement for matrix assembly and storage for differential operators and intergrid transfer operations.

The three key solver contribution are: First, a robust preconditioner for the Schur complement of the linearized and discretized Stokes system, which is presented in Chapter 6. Second, hybrid spectral–geometric–algebraic multigrid methods constitute a core preconditioning component of the Stokes solver and are essential for preconditioning efficacy and algorithmic and parallel scalability (see Chapter 7). Third, the aforementioned linear iterative methods are complemented by inexact

Newton–Krylov methods to solve the highly nonlinear mantle convection problems (see Chapter 5).

4.1 Finite element discretization

We want to recall the governing equations for mantle convection from Chapter 2 and state them in a form that is well suited for the development and analysis of numerical methods. Given are a bounded domain $\Omega \subset \mathbb{R}^d$, $d \in \{2, 3\}$, with a smooth boundary $\partial\Omega$ and right-hand side forcing $\boldsymbol{f} \in V'$, which lies in the dual of the velocity space found below in Equation (4.2). In addition, we have a generally nonlinear, spatially-varying, sufficiently regular, and bounded viscosity $\mu \geq \mu_{\min} > 0$. In accordance with the geophysical model (2.12), we consider the incompressible Stokes equations with free-slip and no-normal flow boundary conditions:

$$-\nabla \cdot \left[\mu(\boldsymbol{u})\left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\mathsf{T}}\right)\right] + \nabla p = \boldsymbol{f} \quad \text{in } \Omega, \tag{4.1a}$$

$$-\nabla \cdot \boldsymbol{u} = 0 \quad \text{in } \Omega, \tag{4.1b}$$

$$\boldsymbol{T}\left[\mu(\boldsymbol{u})\left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\mathsf{T}}\right) - p\mathbf{I}\right]\boldsymbol{n} = 0 \quad \text{on } \partial\Omega, \tag{4.1c}$$

$$\boldsymbol{u} \cdot \boldsymbol{n} = 0 \quad \text{on } \partial\Omega, \tag{4.1d}$$

where $\boldsymbol{u} \in V$ and $p \in Q$ are the unknown velocity and pressure fields, respectively. For the boundary conditions (4.1c) and (4.1d), we define outward normal vectors $\boldsymbol{n} \in \mathbb{R}^d$ and tangential projectors $\boldsymbol{T} \coloneqq \mathbf{I} - \boldsymbol{n}\boldsymbol{n}^{\mathsf{T}}$. The function space for the velocity is

$$V \coloneqq \left\{\boldsymbol{u} \in (H^1(\Omega))^d \mid \boldsymbol{n} \cdot \boldsymbol{u} = 0 \text{ on } \partial\Omega\right\}, \tag{4.2}$$

which incorporates Dirichlet boundary conditions in normal direction, and the function space for the pressure is $Q \coloneqq L^2(\Omega)/\mathbb{R}$, which is the subspace of $L^2(\Omega)$ not containing non-zero constant functions.

In the context of mantle convection problems, we assume the nonlinear viscosity to be

$$\mu(\boldsymbol{u}) = \mu_{\text{reg}}(T, \dot{\varepsilon}_{\text{II}}(\boldsymbol{u})) = \mu_{\min} + \min\left(\frac{\tau_{\text{yield}}}{2\dot{\varepsilon}_{\text{II}}}, w(\boldsymbol{x}) \min\left(\mu_{\max}, a(T)\left(\dot{\varepsilon}_{\text{II}} - d\right)^{\frac{1}{n}} \dot{\varepsilon}_{\text{II}}^{-1}\right)\right), \tag{4.3}$$

which was defined previously in Corollary 2.8.2. This viscosity is highly heterogeneous, which stems from its dependence on temperature T and strain rate $\dot{\varepsilon}_{\text{II}}$, it exhibits sharp viscosity gradients in narrow regions (six orders of magnitude drop in $\sim$5 km) modeling tectonic plate boundaries as described in Section 2.5. This leads to a wide range of spatial scales since small localized features at plate boundaries of size $\mathcal{O}(1\,\text{km})$ influence plate motion at continental scales of $\mathcal{O}(1000\,\text{km})$. Furthermore, the strain rate dependence in (4.3) develops challenging anisotropies upon linearization with Newton's method (more in Chapter 5). Therefore, when aiming to simulate realistic Earth's mantle convection at a global scale, the complex character of the flow presents severe computational challenges for iterative solvers due to poor conditioning of the nonlinear and linear systems that arise.

To derive the variational formulation of the Stokes system (4.1), formally, multiply with test functions $\boldsymbol{v} \in V$ and $q \in Q$,

$$\int_\Omega -\nabla \cdot \left[\mu(\boldsymbol{x}) \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^\mathsf{T}\right)\right] \cdot \boldsymbol{v} \,\mathrm{d}\boldsymbol{x} + \int_\Omega \nabla p \cdot \boldsymbol{v} \,\mathrm{d}\boldsymbol{x} = \int_\Omega \boldsymbol{f} \cdot \boldsymbol{v} \,\mathrm{d}\boldsymbol{x},$$
$$\int_\Omega (\nabla \cdot \boldsymbol{u}) q \,\mathrm{d}\boldsymbol{x} = 0,$$

and integrate by parts to obtain

$$A(\boldsymbol{u}\,,\boldsymbol{v}) + B(\boldsymbol{v}\,,p) + E(p\,,\boldsymbol{u}\,,\boldsymbol{v}) = F(\boldsymbol{v})\,,$$
$$B(\boldsymbol{u}\,,q) = 0,$$

where we defined the following linear forms

$$\begin{aligned}
A(\boldsymbol{u}\,,\boldsymbol{v}) &\coloneqq \int_\Omega \frac{\mu(\boldsymbol{x})}{2} (\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^\mathsf{T}) : (\nabla \boldsymbol{v} + \nabla \boldsymbol{v}^\mathsf{T}) \,\mathrm{d}\boldsymbol{x}, \\
B(\boldsymbol{u}\,,q) &\coloneqq -\int_\Omega (\nabla \cdot \boldsymbol{u}) q \,\mathrm{d}\boldsymbol{x}, \\
E(p\,,\boldsymbol{u}\,,\boldsymbol{v}) &\coloneqq \int_{\partial\Omega} \left[\left(p\mathbf{I} - \mu(\boldsymbol{x}) \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^\mathsf{T}\right)\right) \boldsymbol{n}\right] \cdot \boldsymbol{v} \,\mathrm{d}S(\boldsymbol{x}), \\
F(\boldsymbol{v}) &\coloneqq \int_\Omega \boldsymbol{f} \cdot \boldsymbol{v} \,\mathrm{d}\boldsymbol{x}.
\end{aligned}$$

The boundary conditions (4.1c) and (4.1d) eliminate boundary integrals, thus $E(p\,,\boldsymbol{u}\,,\boldsymbol{v}) = 0$. This leaves the following variational formulation for the Stokes system corresponding to the strong form (4.1):

$$A(\boldsymbol{u}\,,\boldsymbol{v}) + B(\boldsymbol{v}\,,p) = F(\boldsymbol{v}) \quad \text{for all } \boldsymbol{v} \in V, \tag{4.4a}$$
$$B(\boldsymbol{u}\,,q) = 0 \qquad \text{for all } q \in Q. \tag{4.4b}$$

In general, our Stokes system in Equations (4.1) and (4.4) is nonlinear, therefore the nonlinear infinite dimensional system is first linearized and then discretized. The linearization is discussed later in Chapter 5 and we proceed here assuming a linear (or linearized) Stokes problem.

We approximate the solution of the (now linear) infinite dimensional variational problem (4.4) using finite elements [16, 23, 46, 53]. Consequently, we choose finite dimensional subspaces $V^h \subset V$ and $Q^h \subset Q$ with (finite) sets of basis functions $\{\boldsymbol{\phi}_i\}_i \subset V^h$ and $\{\psi_k\}_k \subset Q^h$, respectively, and define the discretized Stokes problem[1]: Find $(\boldsymbol{u}^h, p^h) \in V^h \times Q^h$ such that

$$A\left(\boldsymbol{u}^h\,,\boldsymbol{v}^h\right) + B\left(\boldsymbol{v}^h\,,p^h\right) = F\left(\boldsymbol{v}^h\right) \quad \text{for all } \boldsymbol{v}^h \in V^h, \tag{4.5a}$$
$$B\left(\boldsymbol{u}^h\,,q^h\right) = 0 \qquad \text{for all } q^h \in Q^h. \tag{4.5b}$$

[1]Note that after linearization, e.g., with Newton's or Picard's method, the right-hand side becomes the negative Stokes residual and the velocity and pressure are Newton/Picard steps.

We identify the corresponding linear algebraic Stokes system as

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}^\mathsf{T} \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix}, \tag{4.6}$$

where the entries of the discretized viscous stress operator, $\mathbf{A}$, and the discretized divergence, $\mathbf{B}$, are

$$\mathbf{A} = [\mathbf{a}_{i,j}], \quad \mathbf{a}_{i,j} \coloneqq A(\boldsymbol{\phi}_i \,, \boldsymbol{\phi}_j) = \int_\Omega \frac{\mu(\boldsymbol{x})}{2} (\nabla \boldsymbol{\phi}_i + \nabla \boldsymbol{\phi}_i^\mathsf{T}) : (\nabla \boldsymbol{\phi}_j + \nabla \boldsymbol{\phi}_j^\mathsf{T}) \, \mathrm{d}\boldsymbol{x},$$
$$\mathbf{B} = [\mathbf{b}_{k,j}], \quad \mathbf{b}_{k,j} \coloneqq B(\boldsymbol{\phi}_j \,, \psi_k) = -\int_\Omega (\nabla \cdot \boldsymbol{\phi}_j) \psi_k \, \mathrm{d}\boldsymbol{x},$$

and the discretized gradient, $\mathbf{B}^\mathsf{T}$, is the transpose of matrix $\mathbf{B}$. The discretization is carried out by high-order finite elements on aggressively adaptively refined hexahedral meshes with velocity–pressure pairings $V^h \times Q^h = \mathbb{Q}_k \times \mathbb{P}_{k-1}^{\text{disc}}$ of polynomial order $k \geq 2$ with a continuous, nodal velocity approximation $\mathbb{Q}_k$ and a discontinuous, modal pressure approximation $\mathbb{P}_{k-1}^{\text{disc}}$. These pairings yield optimal asymptotic convergence rates of the finite element approximation to the infinite-dimensional solution with decreasing mesh element size, are inf-sup stable on general, non-conforming hexahedral meshes with "hanging nodes," and have the advantage of preserving mass locally at the element level due to the discontinuous pressure [46, 62, 111].

While these properties have been recognized to be important for geophysics applications (e.g., see [83, 84]), the high-order discretization, adaptivity, and discontinuous pressure approximation present significant additional difficulties for iterative solvers (relative to low order, uniform grid, continuous discretizations). Finally, a number of frontier geophysical problems, such as global mantle convection with plate boundary-resolving meshes and continental ice sheet models with grounding line-resolving meshes, result in billions of degrees of freedom $(\mathbf{u}, \mathbf{p})$, demanding efficient execution and scalability on leading edge supercomputers [27, 97, 110].

To complete the discussion on discretizations, we mention other popular choices for finite element spaces $V^h \times Q^h$. First, the stable high-order velocity–pressure pairing $\mathbb{Q}_k \times \mathbb{Q}_{k-2}^{\text{disc}}$ of polynomial order $k \geq 2$ uses a continuous, nodal velocity approximation $\mathbb{Q}_k$ and a discontinuous, nodal pressure approximation $\mathbb{Q}_{k-2}^{\text{disc}}$. This finite element pairing can exhibit better approximation properties for distorted elements [71], but does not provide the asymptotically optimal approximation property as $\mathbb{P}_{k-1}^{\text{disc}}$. Second, stabilized tri-linear $\mathbb{Q}_1 \times \mathbb{Q}_1$ pairings employ continuous velocity and pressure spaces of equal order. To achieve stability, the linear algebraic Stokes system (4.6) is modified:

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}^\mathsf{T} \\ \mathbf{B} & -\mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix}.$$

The stabilization matrix $\mathbf{C}$, which relaxes the incompressibility condition and thereby weakens mass conservation, is generated via the bilinear form

$$C(p \,, q) \coloneqq \sum_{\Omega_e} \int_{\Omega_e} \frac{1}{\mu(\boldsymbol{x})} \left(p - \Pi \, p\right) \left(q - \Pi \, q\right) \, \mathrm{d}\boldsymbol{x},$$

where $\Omega_e \subset \Omega$ denotes an element with index e and Π is the L^2-projection onto the space of element-wise constant functions.

4.2 Iterative methods for the linear algebraic Stokes system

The discretization of the Stokes system in the context of mantle convection applications results in a very poorly conditioned algebraic system (4.6) with up to hundreds of billions of unknowns, which requires a preconditioned Krylov iterative method. Such a Krylov method needs only the application of the left hand side operator in Equation (4.6) to vectors, which we implement in a matrix-free fashion using elemental loops. This exploits the tensor-product structure of the element-level basis functions $\{\boldsymbol{\phi}_i\}_i$ in $\mathbb{Q}_k$, resulting in a reduced number of operations [38].

We derive an alternative form of (4.6) by performing blockwise Gaussian elimination

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}^\mathsf{T} \\ \mathbf{0} & \mathbf{B}\mathbf{A}^{-1}\mathbf{B}^\mathsf{T} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B}^\mathsf{T} \\ \mathbf{0} & \mathbf{S} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{B}\mathbf{A}^{-1}\mathbf{f} \end{bmatrix},$$

where we identified the (2,2) block of the matrix as the (negative[2]) Schur complement, $\mathbf{S} \coloneqq (\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^\mathsf{T})$. This motivates the use of an upper triangular block matrix within GMRES as the Krylov solver [102] with right preconditioning:

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}^\mathsf{T} \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{A}} & \mathbf{B}^\mathsf{T} \\ \mathbf{0} & \tilde{\mathbf{S}} \end{bmatrix}^{-1} \begin{bmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{p}} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix}. \tag{4.7}$$

Note that the original solution to (4.6) is recovered by applying the preconditioner once to the solution of (4.7). Here, approximations of the inverse of the viscous block, $\tilde{\mathbf{A}}^{-1} \approx \mathbf{A}^{-1}$, and the inverse of the Schur complement, $\tilde{\mathbf{S}}^{-1} \approx (\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^\mathsf{T})^{-1}$, are required. This particular combination of Krylov method and preconditioner is known to converge in just two iterations for exact choices of $\tilde{\mathbf{A}}^{-1}$ and $\tilde{\mathbf{S}}^{-1}$ [12]. The approximation of the inverse of the viscous block $\tilde{\mathbf{A}}^{-1}$ is computed by a V-cycle of our *hybrid spectral–geometric–algebraic multigrid* method as detailed in Chapter 7. For the approximative inverse of the Schur complement $\tilde{\mathbf{S}}^{-1}$, we developed a method called *weighted BFBT*, which is described in Chapter 6.

Null space due to unconstrained mean pressure

The infinite-dimensional Stokes system (4.1) and therefore also its discretized forms in Equations (4.5) and (4.6) all have a non-trivial null space for the pressure component,

$$\mathrm{Ker}(\mathbf{B}^\mathsf{T}) = \mathrm{span}(\mathbf{1}),$$

spanned by the vector, $\mathbf{1}$, which is the coefficient vector for a basis $\{\psi_k\}_k$ of Q^h representing the constant function with value 1 everywhere. Note that for a nodal discretization space Q^h, the vector has ones in all entries, but for modal discretizations the entries of $\mathbf{1}$ generally differ from one. In order to avoid possible computational issues in the iterative methods arising from an unconstrained null space, one can enforce the pressure component to have zero mean. This is enabled by projecting out the mean value of the pressure.

[2] Our definition is the negative Schur complement. However, as in [46], we prefer to work with positive-definite operators and thus define the Schur complement to be positive rather than negative definite.

We begin by observing that we can compute the integrals

$$\int_\Omega p \,\mathrm{d}\boldsymbol{x} = \mathbf{1}^\mathsf{T}\mathbf{M}_p\,\mathbf{p} \quad \text{and} \quad |\Omega| = \int_\Omega 1 \,\mathrm{d}\boldsymbol{x} = \mathbf{1}^\mathsf{T}\mathbf{M}_p\,\mathbf{1},$$

where $\mathbf{M}_p$ is the mass matrix of the pressure space, $p \in Q^h$, $\mathbf{p}$ is the vector containing the degrees of freedom corresponding to p, and $\mathbf{1}$ describes the unit function of the pressure space. These integrals are used to define the projection operator

$$\Pi_{p\text{-NSP}} \coloneqq \mathbf{I} - (\mathbf{1}^\mathsf{T}\mathbf{M}_p\,\mathbf{1})^{-1}\mathbf{1}(\mathbf{M}_p\,\mathbf{1})^\mathsf{T}, \tag{4.8}$$

where $\mathbf{I}$ is the identity matrix. The projector $\Pi_{p\text{-NSP}}$ then has the desired property that applying it to a vector $\mathbf{p}$ of pressure degrees of freedom yields

$$\overline{\mathbf{p}} \coloneqq \Pi_{p\text{-NSP}}\mathbf{p} = \left(\mathbf{I} - (\mathbf{1}^\mathsf{T}\mathbf{M}_p\,\mathbf{1})^{-1}\mathbf{1}(\mathbf{M}_p\,\mathbf{1})^\mathsf{T}\right)\mathbf{p} = \mathbf{p} - \mathbf{1}\frac{\mathbf{1}^\mathsf{T}\mathbf{M}_p\,\mathbf{p}}{\mathbf{1}^\mathsf{T}\mathbf{M}_p\,\mathbf{1}}$$

and the resulting vector $\overline{\mathbf{p}}$ has zero mean because

$$\int_\Omega \overline{p} \,\mathrm{d}\boldsymbol{x} = \mathbf{1}^\mathsf{T}\mathbf{M}_p\,\overline{\mathbf{p}} = \mathbf{1}^\mathsf{T}\mathbf{M}_p\left(\mathbf{p} - \mathbf{1}\frac{\mathbf{1}^\mathsf{T}\mathbf{M}_p\,\mathbf{p}}{\mathbf{1}^\mathsf{T}\mathbf{M}_p\,\mathbf{1}}\right) = \mathbf{1}^\mathsf{T}\mathbf{M}_p\,\mathbf{p} - \mathbf{1}^\mathsf{T}\mathbf{M}_p\,\mathbf{p} = 0.$$

Null space due to unconstrained mean velocity rotations

While the null space for the pressure component generally occurs in Stokes problems as long as boundary conditions for the pressure are not prescribed, the Stokes operator on the left-hand side of Equation (4.1) can additionally have a non-trivial null space for the velocity component in mantle convection applications. If the domain Ω has spherical boundaries, which is the case for a global mantle domain with spherical surface and core–mantle boundaries (Γ_{surf} and Γ_{core}, respectively), the boundary conditions for the velocity (4.1d) leave a three-dimensional null space of mean velocity rotations.

To project out nonzero mean velocity rotations with respect to a given center of rotation, $\mathbf{c} = \left[c_x, c_y, c_z\right]^\mathsf{T}$, we first define three vector fields, $\mathbf{r}_x$, $\mathbf{r}_y$, $\mathbf{r}_z$, each of which describe a rotation about one coordinate axis:

$$\begin{aligned}
\mathbf{r}_x(\boldsymbol{x}) = \mathbf{r}_x(x,y,z) &\coloneqq \left[0, -(z-c_z), (y-c_y)\right]^\mathsf{T},\\
\mathbf{r}_y(\boldsymbol{x}) = \mathbf{r}_y(x,y,z) &\coloneqq \left[(z-c_z), 0, -(x-c_x)\right]^\mathsf{T},\\
\mathbf{r}_z(\boldsymbol{x}) = \mathbf{r}_z(x,y,z) &\coloneqq \left[-(y-c_y), (x-c_x), 0\right]^\mathsf{T},
\end{aligned}$$

where points $\boldsymbol{x} = [x,y,z]^\mathsf{T}$ correspond to the discretization nodes of the velocity. In addition to the center of rotation, we require the moment of inertia corresponding to domain Ω, which is denoted as $\mathbf{m} = \left[m_x, m_y, m_z\right]^\mathsf{T}$. Due to derivations for integrals and mean values analogous to the mean pressure above, we define the projection operator

$$\Pi_{\boldsymbol{u}\text{-NSP}} \coloneqq \mathbf{I} - \frac{1}{m_x}\mathbf{r}_x(\mathbf{M}_{\boldsymbol{u}}\,\mathbf{r}_x)^\mathsf{T} - \frac{1}{m_y}\mathbf{r}_y(\mathbf{M}_{\boldsymbol{u}}\,\mathbf{r}_y)^\mathsf{T} - \frac{1}{m_z}\mathbf{r}_z(\mathbf{M}_{\boldsymbol{u}}\,\mathbf{r}_z)^\mathsf{T}, \tag{4.9}$$

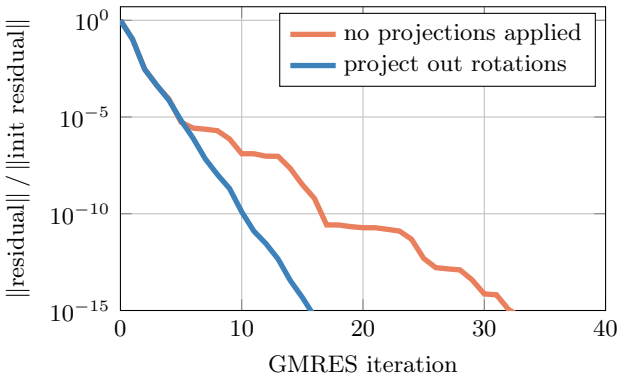


Figure 4.2: Improvement in convergence stability if nonzero mean velocity rotations are projected out during iterations *(blue curve)*. If mean rotations are ignored, the corresponding null space causes sections of stagnating residual reduction *(red curve)*.

which removes mean rotations from vectors that represent velocity fields.

Finally, we demonstrate the effect of a non-trivial rotational null space on the convergence of iterative Krylov methods. To this end, we prescribe velocity boundary conditions (4.1d), where only the normal component is constrained by Dirichlet boundary conditions. Additionally, we consider a constant viscosity and the sub-system corresponding to the (1,1) block of the Stokes system; hence, we want to compute solution $\mathbf{u}$ in $\mathbf{A}\mathbf{u} = \mathbf{f}$. We apply right-preconditioning as in (4.7) and use GMRES as a Krylov solver. In the reference calculations, we do not apply any projections,

$$\mathbf{A}\tilde{\mathbf{A}}^{-1}\tilde{\mathbf{u}} = \mathbf{f},$$

and in the second case, we remove rotations from $\mathrm{Ran}(\mathbf{A})$ as well as before and after applying the preconditioner $\tilde{\mathbf{A}}^{-1}$,

$$\left(\Pi_{\boldsymbol{u}\text{-NSP}}^{\mathsf{T}}\mathbf{A}\right)\left(\Pi_{\boldsymbol{u}\text{-NSP}}\tilde{\mathbf{A}}^{-1}\Pi_{\boldsymbol{u}\text{-NSP}}^{\mathsf{T}}\right)\tilde{\mathbf{u}} = \Pi_{\boldsymbol{u}\text{-NSP}}^{\mathsf{T}}\mathbf{f}.$$

For both cases the same multigrid V-cycle is used as a preconditioner. The improvement in stability for the convergence of the Krylov solver can be observed in Figure 4.2. The relative residual reduction with projections *(blue curve)* is fast and stable as is expected for a simple elliptic problem. Without projections *(red curve)*, however, the GMRES solver takes twice as many iterations—for a very simple low-resolution, constant viscosity problem setup—because the residual reduction is stagnating during some iterations and resembles a staircase shape. The performance gap is expected to increase with larger and more complex problems. This shows that the Krylov subspaces cause GMRES to iterate over the null space in $\mathrm{Ran}(\mathbf{A})$, in which the residual cannot be reduced. The projections avoid this, because $\mathrm{Ran}(\Pi_{\boldsymbol{u}\text{-NSP}}^{\mathsf{T}})$ excludes non-trivial mean rotations. Furthermore, the projections around the preconditioner are most likely necessary for consistency between matrix $\mathbf{A}$ and its approximation $\tilde{\mathbf{A}}$. While these projection tests include only the critical Stokes block that is generating the null space, we also performed numerical experiments with the whole Stokes system and obtained similar results on convergence stability.

5

Inexact Newton–Krylov Methods

Newton's method is used to iteratively approximate solutions of nonlinear problems by means of subsequent approximations of the nonlinearities using derivatives. In a nutshell, given some real valued function $J(u)$ that we like to minimize over all possible arguments u, Newton's method uses a truncated Taylor series at u as an approximation for $J(u + \hat{u})$ in a neighborhood around u, expressed as $u + \hat{u}$. While u is the known current iterate, $\hat{u}$ is the Newton step to improve the iterate such that $J(u+\hat{u}) < J(u)$. From the Taylor series we obtain a linear system for $\hat{u}$, namely $H(u)\hat{u} = -g(u)$, which we call linearized system or linearization. In this system $g(u)$ is the first-order variation (gradient) of $J(u)$ and $H(u)$ is the second-order variation (Hessian) of $J(u)$. What follows in this chapter are derivations of gradients and Hessians for specific challenging nonlinear problems, modifications to Newton's linearization, and its numerical solution.

5.1 Inexact Newton–Krylov methods for nonlinear Stokes

The Stokes system (4.1) for mantle convection becomes nonlinear through the dependence of the viscosity (4.3) on the second invariant of the strain rate. We employ an inexact Newton–Krylov method [41, 42, 89] for the nonlinear Stokes equations, i.e., we use a sequence of linearizations of (4.1) and approximately solve the resulting linearized systems using a preconditioned Krylov method as described in Chapter 4, 6 and 7. The rheology (4.3) is modified such that it incorporates upper and lower bounds for the viscosity in a differentiable manner (see Section 2.8), permitting the use of Newton's method. To compute a Newton update, we find the (inexact) solution of the linearized Stokes system presented in the following lemma.

Lemma 5.1.1 (General linearized Stokes system)**.** *Given a current iterate, a velocity–pressure pair* $(\boldsymbol{u}, p)$*, the Newton step* $(\hat{\boldsymbol{u}}, \hat{p})$ *is computed by solving the linearized Stokes system*

$$-\nabla \cdot (2\mu'(\boldsymbol{u})\, \nabla_{\mathrm{s}} \hat{\boldsymbol{u}}) + \nabla \hat{p} = -\boldsymbol{r}_{\mathrm{mom}}, \tag{5.1a}$$

$$-\nabla \cdot \hat{\boldsymbol{u}} = -r_{\mathrm{mass}}, \tag{5.1b}$$

with coefficient

$$\mu'(\boldsymbol{u}) \coloneqq \mu(\boldsymbol{u})\,\mathbf{I} + \dot{\varepsilon}_{\text{II}} \frac{\partial \mu}{\partial \dot{\varepsilon}_{\text{II}}} \frac{\nabla_{\text{s}}\boldsymbol{u} \otimes \nabla_{\text{s}}\boldsymbol{u}}{|\nabla_{\text{s}}\boldsymbol{u}|_F^2}, \tag{5.2}$$

where the residuals of the nonlinear Stokes momentum and mass equations, Equations (4.1a) *and* (4.1b)*, respectively appear on the right-hand side of* (5.1)*. We also introduced the notation for the symmetric gradient,* $\nabla_{\text{s}}\boldsymbol{u} \coloneqq \frac{1}{2}(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^{\mathsf{T}})$*, and the Frobenius norm,* $|\nabla_{\text{s}}\boldsymbol{u}|_F \coloneqq \sqrt{\nabla_{\text{s}}\boldsymbol{u} : \nabla_{\text{s}}\boldsymbol{u}}$*.*

Proof. We calculate the first-order variation of the following Frobenius norm:

$$\delta_{\boldsymbol{u}}[|\nabla_{\text{s}}\boldsymbol{u}|_F](\hat{\boldsymbol{u}}) = \delta_{\boldsymbol{u}}[\sqrt{\nabla_{\text{s}}\boldsymbol{u} : \nabla_{\text{s}}\boldsymbol{u}}](\hat{\boldsymbol{u}}) = \frac{\nabla_{\text{s}}\boldsymbol{u} : \nabla_{\text{s}}\hat{\boldsymbol{u}}}{|\nabla_{\text{s}}\boldsymbol{u}|_F}.$$

Using this together with the product rule, chain rule, and the identity $\dot{\varepsilon}_{\text{II}} = \frac{1}{\sqrt{2}}\,|\nabla_{\text{s}}\boldsymbol{u}|_F$, we get

$$\begin{aligned}
\delta_{\boldsymbol{u}}[\mu(\dot{\varepsilon}_{\text{II}}(\boldsymbol{u}))\,\nabla_{\text{s}}\boldsymbol{u}](\hat{\boldsymbol{u}}) &= \delta_{\boldsymbol{u}}[\mu(\dot{\varepsilon}_{\text{II}}(\boldsymbol{u}))](\hat{\boldsymbol{u}})\,\nabla_{\text{s}}\boldsymbol{u} + \mu(\dot{\varepsilon}_{\text{II}})\nabla_{\text{s}}\hat{\boldsymbol{u}} \\
&= \frac{\partial \mu}{\partial \dot{\varepsilon}_{\text{II}}}\,\delta_{\boldsymbol{u}}[\dot{\varepsilon}_{\text{II}}(\boldsymbol{u})](\hat{\boldsymbol{u}})\,\nabla_{\text{s}}\boldsymbol{u} + \mu(\dot{\varepsilon}_{\text{II}})\nabla_{\text{s}}\hat{\boldsymbol{u}} \\
&= \frac{\partial \mu}{\partial \dot{\varepsilon}_{\text{II}}} \frac{1}{\sqrt{2}} \frac{\nabla_{\text{s}}\boldsymbol{u} : \nabla_{\text{s}}\hat{\boldsymbol{u}}}{|\nabla_{\text{s}}\boldsymbol{u}|_F} \nabla_{\text{s}}\boldsymbol{u} + \mu(\dot{\varepsilon}_{\text{II}})\nabla_{\text{s}}\hat{\boldsymbol{u}}.
\end{aligned}$$

Observing the outer product identity

$$(\nabla_{\text{s}}\boldsymbol{u} : \nabla_{\text{s}}\hat{\boldsymbol{u}})\,\nabla_{\text{s}}\boldsymbol{u} = (\nabla_{\text{s}}\boldsymbol{u} \otimes \nabla_{\text{s}}\boldsymbol{u})\,\nabla_{\text{s}}\hat{\boldsymbol{u}}$$

and using the relation between Frobenius norm and second invariant one more time gives us (5.2). □

Note that what plays the role of viscosity in the Newton step (5.1a) is an anisotropic fourth-order tensor (5.2). Our multigrid method from Chapter 7 allows treatment of the full fourth-order tensor anisotropic coefficient of a Newton step. In order to fully describe the linearized Stokes system for mantle's rheology, we need to calculate the derivative of the viscosity with respect to $\dot{\varepsilon}_{\text{II}}$, which is done in the next lemma.

Lemma 5.1.2 (Derivative of mantle's viscosity)**.** *Consider the regularized viscosity model from Equation* (2.28) *rewritten in the four possible cases:*

$$\mu(\dot{\varepsilon}_{\text{II}}) = \begin{cases} w\,(\boldsymbol{x})\,\mu_{\max} & \textit{(upper viscosity bound reached)}, \\ w\,(\boldsymbol{x})\,a\,(T)\,\dot{\varepsilon}_{\text{II}}^{\frac{1-n}{n}} & \textit{(strain rate weakening)}, \\ \frac{\tau_{\text{yield}}}{2\dot{\varepsilon}_{\text{II}}} & \textit{(plastic yielding)}, \\ \mu_{\min} & \textit{(lower viscosity bound reached)}. \end{cases}$$

Its derivative with respect to $\dot{\varepsilon}_{\text{II}}$ *is*

$$\frac{\partial \mu}{\partial \dot{\varepsilon}_{\text{II}}} = \mu(\dot{\varepsilon}_{\text{II}})\,\dot{\varepsilon}_{\text{II}}^{-1}\,\theta \quad \textit{where} \quad \theta \coloneqq \begin{cases} 0 & \textit{(upper viscosity bound reached)}, \\ \frac{1-n}{n} & \textit{(strain rate weakening)}, \\ -1 & \textit{(plastic yielding)}, \\ 0 & \textit{(lower viscosity bound reached)}. \end{cases} \tag{5.3}$$

Corollary 5.1.3 (Linearized Stokes system for mantle convection)**.** *Given a current iterate, a velocity–pressure pair* $(\boldsymbol{u}, p)$*, the Newton step* $(\hat{\boldsymbol{u}}, \hat{p})$ *is computed by solving the linearized Stokes system*

$$-\nabla \cdot \left(2\mu(\boldsymbol{u}) \left(\mathbf{I} + \theta \frac{\nabla_{\mathrm{s}} \boldsymbol{u} \otimes \nabla_{\mathrm{s}} \boldsymbol{u}}{|\nabla_{\mathrm{s}} \boldsymbol{u}|_F^2} \right) \nabla_{\mathrm{s}} \hat{\boldsymbol{u}} \right) + \nabla \hat{p} = -\boldsymbol{r}_{\mathrm{mom}}, \tag{5.4a}$$

$$-\nabla \cdot \hat{\boldsymbol{u}} = -r_{\mathrm{mass}}, \tag{5.4b}$$

where we use $-1 \leq \theta \leq 0$ *from* (5.3)*.*

Note that for $\theta = -1$ the coefficient in Equation (5.4a) resembles an orthogonal projector, therefore generating a null space for the linearized system (5.4). In practice, however, we choose the regularized viscosity (2.33) with the effect that θ is bounded away from -1. The (now near) orthogonal projector in (5.4a) still remains a challenge for plastic yielding rheologies, where θ is close to -1, and causes issues with Newton convergence. Therefore, the remaining sections in this chapter analyze the corresponding issues and present alternative linearizations.

A Newton step $(\hat{\boldsymbol{u}}, \hat{p})$ computed as the inexact solution of (5.4) is only accepted if the norm of the updated momentum and mass residuals is reduced. If this is not the case, the step length is reduced by a factor $0 < \alpha < 1$ via a backtracking line search algorithm [89], the update therefore being

$$(\boldsymbol{u}, p) \leftarrow (\boldsymbol{u}, p) + \alpha(\hat{\boldsymbol{u}}, \hat{p}).$$

The residual of the momentum equation generally satisfies $\boldsymbol{r}_{\mathrm{mom}} \in H^{-1}(\Omega)$ and we measure its norm in the H^{-1}-norm for backtracking line search, as opposed to the more common L^2-norm. The norm corresponding to the space of lower regularity functions, H^{-1}, avoids overly conservative Newton steps that are significantly reduced from one. We observed these overly conservative steps for mantle convection problems during initial nonlinear iterations, whenever an iterate $(\boldsymbol{u}, p)$ was far away from the nonlinear solution. The next Lemma 5.1.4 determines how the H^{-1}-norm can be computed in practice.[1]

Lemma 5.1.4 (H^{-1}-norm)**.** *Let* $u' \in H^{-1}(\Omega)$ *be given, then its norm satisfies*

$$\left\| u' \right\|_{H^{-1}} = \left(u' \, , u \right)_{L^2}^{1/2}, \tag{5.5}$$

where $u \in H_0^1(\Omega)$ *is the unique solution of the elliptic PDE with homogeneous Dirichlet boundary conditions,*

$$-\Delta u + u = u'. \tag{5.6}$$

Proof. For an arbitrary $u \in H_0^1(\Omega)$, the H^1-norm is defined by

$$\|u\|_{H^1}^2 = (u \, , u)_{H^1} = (\nabla u \, , \nabla u)_{L^2} + (u \, , u)_{L^2} \, . \tag{5.7}$$

[1] Lemma 5.1.4 describes the H^{-1}-norm for a scalar valued function. To compute the norm of a vector valued velocity field, we apply the result of the lemma to each component of the velocity.

If u satisfies the boundary value problem (5.6), the H^{-1}-norm of u' has the form

$$\left\|u'\right\|_{H^{-1}} = \sup_{v \in H_0^1} \frac{(u' \,,\, v)_{L^2}}{\|v\|_{H^1}} = \sup_{v \in H_0^1} \frac{((-\Delta + 1)u \,,\, v)_{L^2}}{\|v\|_{H^1}},$$

where we implicitly assume $\|v\|_{H^1} \neq 0$. Since the supremum is reached for $v \equiv u$, we substitute and integrate by parts

$$\left\|u'\right\|_{H^{-1}} = \frac{((-\Delta + 1)u \,,\, u)_{L^2}}{\|u\|_{H^1}} = \frac{1}{\|u\|_{H^1}} \left((\nabla u \,,\, \nabla u)_{L^2} + \int_{\partial\Omega} (\nabla u \cdot \boldsymbol{n}) u \,\mathrm{d}\boldsymbol{x} + (u \,,\, u)_{L^2} \right)$$

and use (5.7) to arrive at

$$\left\|u'\right\|_{H^{-1}} = \|u\|_{H^1} + \frac{1}{\|u\|_{H^1}} \int_{\partial\Omega} (\nabla u \cdot \boldsymbol{n})\, u \,\mathrm{d}\boldsymbol{x} = \|u\|_{H^1} \,,$$

where the boundary term vanishes because $u \in H_0^1(\Omega)$. Finally, Equation (5.7), relation (5.6), and integration by parts yield

$$\left\|u'\right\|_{H^{-1}}^2 = \|u\|_{H^1}^2 = \left(u' \,,\, (-\Delta + 1)^{-1} u'\right)_{L^2} \,,$$

which shows claim (5.5). □

Algorithm 5.1.1 Inexact Newton–Krylov for nonlinear Stokes flow

input initial guess $(\boldsymbol{u}_0, p_0)$ and corresponding residual $R_0 = (\|\boldsymbol{r}_{\mathrm{mom},0}\|_{H^{-1}}^2 + \|r_{\mathrm{mass},0}\|_{L^2}^2)^{1/2}$, max. Krylov relative tolerance $\varepsilon_{\max} \in (0,1)$, parameters $\beta = (1+\sqrt{5})/2$, $\gamma = 10^{-4}$
for $k = 0, 1, \ldots$ until convergence **do**
  if $k = 0$ **then**
    $\varepsilon \leftarrow \varepsilon_{\max}$ ▷ set initial Krylov rel. tol.
  else
    $\varepsilon \leftarrow \varepsilon_{\max} \left(\frac{R_k}{R_{k-1}}\right)^{\beta}$ ▷ set adaptive Krylov rel. tol.
  end if
  Solve inexactly to rel. tol. ε for $(\hat{\boldsymbol{u}}_k, \hat{p}_k)$ in Equation (5.4) ▷ compute Newton step
  $\alpha \leftarrow 1$
  $(\boldsymbol{u}_{k+1}, p_{k+1}) \leftarrow (\boldsymbol{u}_k, p_k) + \alpha(\hat{\boldsymbol{u}}_k, \hat{p}_k)$
  $R_{k+1} \leftarrow (\|\boldsymbol{r}_{\mathrm{mom},k+1}\|_{H^{-1}}^2 + \|r_{\mathrm{mass},k+1}\|_{L^2}^2)^{1/2}$
  while $R_{k+1} > (1 - \gamma(1-\varepsilon))R_k$ **do** ▷ backtracking line search
    $\alpha \leftarrow \alpha/2$
    $(\boldsymbol{u}_{k+1}, p_{k+1}) \leftarrow (\boldsymbol{u}_k, p_k) + \alpha(\hat{\boldsymbol{u}}_k, \hat{p}_k)$
    $R_{k+1} \leftarrow (\|\boldsymbol{r}_{\mathrm{mom},k+1}\|_{H^{-1}}^2 + \|r_{\mathrm{mass},k+1}\|_{L^2}^2)^{1/2}$
  end while
end for

We summarize our inexact Newton–Krylov method in Algorithm 5.1.1, where the choice of the parameters that control inexactness is motivated by [42]. Additionally, during the initial Newton iterations, grid continuation is performed in between Newton steps, where the mesh is adapted to variations in the viscosity that arise from the nonlinear dependence on the velocity.

5.2 Abstract derivation of perturbed Newton linearizations

Many applications in computational science and engineering are modelled by optimization problems with Hessians that exhibit a problematic (near) null space upon linearization with Newton's method. The null space we will investigate here is caused by a projector-type coefficient in the Hessian expression, which, in turn, is created by terms in the objective functional that resemble the L^1-norm. One example is nonlinear Stokes flow in the mantle with plastic yielding discussed previously in Section 5.1. Another example is total variation regularization for inverse problems, which is a popular choice for image restoration problems due to its edge preserving properties [100]. In this section, we analyze issues with the standard Newton linearization in an abstract setting and propose an improved linearization. The next Section 5.3 applies these ideas to example problems and applications, which include Stokes flow with yielding and total variation regularization.

Basic definitions

We denote a general inner product $\langle \cdot\,, \cdot \rangle : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ for $n \in \{1, d, d \times d\}$, $d \in \mathbb{N}$, that means for scalar, vector, and matrix valued functions by

$$\langle U\,, V \rangle = \langle U(\boldsymbol{x})\,, V(\boldsymbol{x}) \rangle \coloneqq \begin{cases} U(\boldsymbol{x})\, V(\boldsymbol{x}) & \text{for } U, V : \Omega \to \mathbb{R}, \\ U(\boldsymbol{x}) \cdot V(\boldsymbol{x}) & \text{for } U, V : \Omega \to \mathbb{R}^d, \\ U(\boldsymbol{x}) : V(\boldsymbol{x}) & \text{for } U, V : \Omega \to \mathbb{R}^{d\times d}, \end{cases}$$

operating pointwise on values of functions $U(\boldsymbol{x})$ and $V(\boldsymbol{x})$ at $\boldsymbol{x} \in \Omega$, where $\Omega \subseteq \mathbb{R}^d$. The norm induced by this inner product,

$$|U| \coloneqq |U(\boldsymbol{x})| = \sqrt{\langle U(\boldsymbol{x})\,, U(\boldsymbol{x}) \rangle} = \sqrt{\langle U\,, U \rangle},$$

is thus understood as the magnitude of $U(\boldsymbol{x})$. Further, we write the L^2-inner product of functions $U, V \in L^2(\Omega)$ as

$$(U\,, V) \coloneqq \int_\Omega \langle U\,, V \rangle = \int_\Omega \langle U(\boldsymbol{x})\,, V(\boldsymbol{x}) \rangle \,\mathrm{d}\boldsymbol{x}.$$

The space $L^1(\Omega)^n$, $n \in \{1, d, d \times d\}$, is equipped with the following norm, assuming $U \in L^1(\Omega)^n$,

$$\|U\|_{L^1} = \|U\|_{L^1(\Omega)^n} \coloneqq \int_\Omega |U| = \int_\Omega |U(\boldsymbol{x})| \,\mathrm{d}\boldsymbol{x}$$

and for $\Omega_{\mathrm{NZ}} \coloneqq \Omega \setminus \{\boldsymbol{x} \in \Omega \mid |U(\boldsymbol{x})| = 0\}$ we can write

$$\|U\|_{L^1} = \int_\Omega |U| = \int_{\Omega_{\mathrm{NZ}}} |U| = \int_{\Omega_{\mathrm{NZ}}} \frac{|U|^2}{|U|} = \int_{\Omega_{\mathrm{NZ}}} \left\langle U\,, \frac{U}{|U|} \right\rangle. \tag{5.8}$$

The L^1-norm also satisfies the definition using its dual space L^∞,

$$\|U\|_{L^1} = \sup_{S \in L^\infty} \frac{(U\,, S)}{\|S\|_{L^\infty}} = \sup_{S \in L^\infty} \frac{1}{\|S\|_{L^\infty}} \int_\Omega \langle U\,, S \rangle\,, \tag{5.9}$$

where we implicitly assume $\|S\|_{L^\infty} \neq 0$. Thus, combining (5.8) and (5.9),

$$0 \leq \|U\|_{L^1} - \int_\Omega \langle U \, , S \rangle = \int_\Omega |U| - \langle U \, , S \rangle = \int_{\Omega_{\mathrm{NZ}}} \left\langle U \, , \frac{U}{|U|} - S \right\rangle ,$$

which is minimized by

$$S^* = \frac{U}{|U|} \quad \text{in } \Omega_{\mathrm{NZ}}. \tag{5.10}$$

Model for the L^1-norm and model perturbation

Motivated by the formulations for the L^1-norm that involve a dual variable, we define the following functions.

Definition 5.2.1. Let $U, S \in L^2(\Omega)^n$, $n \in \{1, d, d \times d\}$, and define the real-valued function

$$c(U, S) \coloneqq |U| - \langle U \, , S \rangle \quad \text{in } \Omega.$$

Further, define the following functions with values in $\mathbb{R}^n$:

$$E(U, S) \coloneqq \frac{U}{|U|} - S \quad \text{in } \Omega_{\mathrm{NZ}} \quad \text{(model error)} \tag{5.11}$$

and

$$D(U, S) \coloneqq U - |U| \, S \quad \text{in } \Omega \quad \text{(model perturbation)} \tag{5.12}$$

that we name *model error* and *model perturbation*, respectively due to their intended uses below.

Note that $E(U, S)$ in (5.11) is not well-defined where $U = 0$, while $c(U, S)$ and $D(U, S)$ are well-defined and continuous. The current goal is to create a model for the L^1-norm. Based on (5.10), we formulate an equation that has to be satisfied for a pair (U, S) such that S is the minimizing dual variable. Therefore, we assume for the remainder of this section that $U \in L^2(\Omega)^n$ and $S \in L^\infty(\Omega)^n$, $\|S\|_{L^\infty} \leq 1$, which implies $U \in L^1(\Omega)^n$ and $S \in L^2(\Omega)^n$ for appropriate assumptions on Ω. Our model for the L^1-norm in variational form is

$$\left(D(U, S) \, , \tilde{S} \right) = \int_\Omega \left\langle U - |U| \, S \, , \tilde{S} \right\rangle = 0 \quad \text{for all } \tilde{S} \in L^2(\Omega)^n. \tag{5.13}$$

The choice to include $D(U, S)$ in the model (5.13), as opposed to, e.g., $E(U, S)$, is motivated by its continuity where $U = 0$. We call $D(U, S)$ *model perturbation* because we now consider (5.13) to be not exactly satisfied:

$$D(U, S) \neq 0 \quad \Leftrightarrow \quad \exists \, \tilde{S} \in L^2(\Omega)^n \quad \text{such that} \quad \left(D(U, S) \, , \tilde{S} \right) \neq 0. \tag{5.14}$$

To compute a correction, $\hat{S}$, for the perturbed model, we employ Newton's method and consider the linearized system

$$\left(\delta_U [D(U, S)](\hat{U}) + \delta_S [D(U, S)](\hat{S}) \, , \tilde{S} \right) = - \left(D(U, S) \, , \tilde{S} \right) \quad \text{for all } \tilde{S} \in L^2(\Omega)^n. \tag{5.15}$$

By plugging-in the variations of $D(\cdot,\cdot)$, which are ("$\otimes$" denoting the outer product)

$$\delta_U[D(U,S)](\hat{U}) = \left(\mathbf{I} - \frac{U \otimes S}{|U|}\right)\hat{U}, \quad \delta_S[D(U,S)](\hat{S}) = -\,|U|\,\hat{S},$$

we obtain for (5.15)

$$\int_{\Omega_{\mathrm{NZ}}} \left\langle \left(\mathbf{I} - \frac{U \otimes S}{|U|}\right)\hat{U} - |U|\,\hat{S}\,,\,\tilde{S}\right\rangle = \int_{\Omega_{\mathrm{NZ}}} \left\langle |U|\,S - U\,,\,\tilde{S}\right\rangle \quad \text{for all } \tilde{S} \in L^2(\Omega)^n. \tag{5.16}$$

Note that the linearization (5.16) is an underdetermined system since we have one equation for two unknowns, $\hat{U}$ and $\hat{S}$. Furthermore, it allows the explicit expression for the dual step,

$$\hat{S} = \frac{U}{|U|} - S + \frac{1}{|U|}\left(\mathbf{I} - \frac{U \otimes S}{|U|}\right)\hat{U} \quad \text{in } \Omega_{\mathrm{NZ}}, \tag{5.17}$$

hence, the Newton update for the dual variable S is computed as

$$S \leftarrow S + \hat{S} = \frac{U}{|U|} + \frac{1}{|U|}\left(\mathbf{I} - \frac{U \otimes S}{|U|}\right)\hat{U}. \tag{5.18}$$

An interesting observation from Equation (5.18) is that the updated dual variable depends on the previous U and on the step $\hat{U}$ applied to the linearization of the model. It can therefore be interpreted as the Newton prediction for (or tracking of) S, in contrast to setting S exactly using (5.10) and the updated U.

Abstract nonlinear problem and standard Newton linearization

Consider the following minimization problem: Find $U^* \in L^2(\Omega)^n$ minimizing

$$\min_U J(U), \quad J(U) \coloneqq \int_\Omega |U| - F(U), \quad U \in L^2(\Omega)^n, \tag{5.19}$$

where $F : L^2(\Omega)^n \to \mathbb{R}$ is a given operator that is linear and bounded. To apply Newton's method, we need to derive the gradient and Hessian operators of the functional J, which are the first- and second-order variations of J:

$$g(U)\tilde{U} \coloneqq \delta_U[J(U)](\tilde{U}) = \int_{\Omega_{\mathrm{NZ}}} \left\langle \frac{U}{|U|}\,,\,\tilde{U}\right\rangle - F(\tilde{U}), \tag{5.20}$$

$$\left(H(U)\hat{U}\,,\,\tilde{U}\right) \coloneqq \delta_U\delta_U[J(U)](\tilde{U})(\hat{U}) = \int_{\Omega_{\mathrm{NZ}}} \left\langle \frac{1}{|U|}\left(\mathbf{I} - \frac{U \otimes U}{|U|^2}\right)\hat{U}\,,\,\tilde{U}\right\rangle, \tag{5.21}$$

for $\tilde{U}, \hat{U} \in L^2(\Omega)^n$. Hence, computing the Newton update, $U^+ \leftarrow U + \hat{U}$, requires solving the following linearized system for step $\hat{U}$:

$$\int_{\Omega_{\mathrm{NZ}}} \left\langle \frac{1}{|U|}\left(\mathbf{I} - \frac{U \otimes U}{|U|^2}\right)\hat{U}\,,\,\tilde{U}\right\rangle = -\int_{\Omega_{\mathrm{NZ}}} \left\langle \frac{U}{|U|}\,,\,\tilde{U}\right\rangle - F(\tilde{U}) \quad \text{for all } \tilde{U} \in L^2(\Omega)^n. \tag{5.22}$$

If one would solve the system (5.22) numerically, typically a regularization is incorporated to the inverse magnitude of U in order to avoid numerical instabilities that arise when dividing by

values near zero. The regularization $\epsilon > 0$ is included in the objective functional of the minimization problem (5.19) in the following way

$$J_\epsilon(U) \coloneqq \int_\Omega \sqrt{\langle U\,,U\rangle + \epsilon^2} - F(U) = \int_\Omega |U|_\epsilon - F(U) \quad \text{with} \quad |U|_\epsilon \coloneqq \sqrt{\langle U\,,U\rangle + \epsilon^2}. \tag{5.23}$$

This functional gives rise to a linearized system different from Equation (5.22), namely

$$\int_\Omega \left\langle \frac{1}{|U|_\epsilon}\left(\mathbf{I} - \frac{U\otimes U}{|U|^2+\epsilon^2}\right)\hat{U}\,,\tilde{U}\right\rangle = -\int_\Omega \left\langle \frac{U}{|U|_\epsilon}\,,\tilde{U}\right\rangle - F(\tilde{U}) \quad \text{for all } \tilde{U}\in L^2(\Omega)^n. \tag{5.24}$$

The properties of the Hessian operator of the *standard* Newton linearization in Equation (5.24) are summarized in the next corollary.

Corollary 5.2.2 (Properties of the standard Hessian)**.** *The coefficient of the left-hand side Hessian operator in Equation* (5.24) *includes an anisotropic second-order or fourth-order tensor depending on* $n \in \{d, d\times d\}$*. The tensor component,*

$$\Pi_\epsilon : \mathbb{R}^n \to \mathbb{R}^n, \quad \Pi_\epsilon \coloneqq \mathbf{I} - \frac{U\otimes U}{|U|^2+\epsilon^2}, \tag{5.25}$$

of the Hessian's coefficient approaches an orthogonal projector as $\epsilon \to 0$*. Moreover, the eigenvalue of the Hessian associated to* U *is*

$$\frac{\epsilon^2}{\left(|U|^2+\epsilon^2\right)^{3/2}} \tag{5.26}$$

and, for a function U *with non-trivial magnitude, the eigenvalue* (5.26) *approaches zero as* $\epsilon \to 0$*.*

The projection-type property of the Hessian coefficient and the arbitrarily small eigenvalue cause the Hessian operator to have a near null space. While null spaces can be accommodated by Krylov methods as seen previously in Section 4.2, Newton updates execute less control over this subspace. This causes instabilities in Newton convergence as long as U is not within a small region of the exact solution U^*. The instabilities are observed in numerical experiments in Section 5.4 and in the literature, the problematic Newton step in the initial iterations of the nonlinear solve can significantly delay fast Newton convergence locally around the solution. For challenging problems, even stagnating convergence is observed due to very small step sizes, which are reduced in backtracking algorithms.

Another issue with steps $\hat{U}$ from standard Newton linearizations manifests itself if the objective functional includes switching between different terms depending on $|U|$, e.g., for more complex physical phenomena as we see for mantle's rheology. As an example, let $\alpha > 0$ and add a quadratic term, $|U|^2$, to the objective, which by itself would be addressed effectively with Newton's method:

$$J_\alpha(U) \coloneqq \int_\Omega \chi_{\alpha\le|U|}\,|U|_\epsilon + \chi_{|U|<\alpha}\frac{1}{2}\,|U|^2 - F(U), \tag{5.27}$$

where the indicator functions

$$\chi_{\alpha\le|U|} \coloneqq \begin{cases} 1, & \alpha \le |U|\,, \\ 0, & \text{otherwise,} \end{cases} \quad \text{and} \quad \chi_{|U|<\alpha} \coloneqq \begin{cases} 1, & |U| < \alpha, \\ 0, & \text{otherwise,} \end{cases}$$

subdivide the domain Ω with respect to the magnitude of U. Since we expect the Newton step $\hat{U}$ to exhibit instabilities in the subspace $\operatorname{span}\{U\}$, the behavior of the updated indicators $\chi_{\alpha \leq |U+\hat{U}|}$ and $\chi_{|U+\hat{U}|<\alpha}$ are largely uncontrollable by the nonlinear solver. This propagates to the evaluation of the updated functional $J_\alpha(U+\hat{U})$ and hence affects Newton convergence.

Perturbed and reduced Newton linearization

When deriving Equation (5.22), we implicitly assume that the dual variable S of the L^1-norm solves the model (5.13) exactly. This assumption can be relaxed, especially far from the solution U^*. To accommodate the model error (5.11), we augment the gradient (5.20) by the perturbation of the model from Equation (5.14)

$$\begin{aligned} g(U)\tilde{U} &= \int_\Omega \left\langle S \,,\, \tilde{U} \right\rangle - F(\tilde{U}), \\ \left(D(U,S) \,,\, \tilde{S} \right) &= \int_\Omega \left\langle U - |U|_\epsilon \, S \,,\, \tilde{S} \right\rangle, \end{aligned}$$

where we regularized the magnitude of U as in (5.23) using $\epsilon > 0$. At first, this results in an increase in degrees of freedom and hence a larger system to solve:

$$\begin{aligned} \int_\Omega \left\langle \hat{S} \,,\, \tilde{U} \right\rangle &= -\int_\Omega \left\langle S \,,\, \tilde{U} \right\rangle - F(\tilde{U}) \quad \text{for all } \tilde{U} \in L^2(\Omega)^n, \\ \int_\Omega \left\langle \left(\mathbf{I} - \frac{U \otimes S}{|U|_\epsilon} \right) \hat{U} - |U|_\epsilon \, \hat{S} \,,\, \tilde{S} \right\rangle &= -\int_\Omega \left\langle U - |U|_\epsilon \, S \,,\, \tilde{S} \right\rangle \quad \text{for all } \tilde{S} \in L^2(\Omega)^n. \end{aligned}$$

However, Equation (5.17) provides an explicit form for the step of the dual variable $\hat{S}$. The initially larger linearized system can therefore be reduced to its size prior to the perturbation, i.e., we have only to solve for $\hat{U}$ in

$$\int_\Omega \left\langle \frac{1}{|U|_\epsilon} \left(\mathbf{I} - \frac{U \otimes S}{|U|_\epsilon} \right) \hat{U} \,,\, \tilde{U} \right\rangle = -\int_\Omega \left\langle \frac{U}{|U|_\epsilon} \,,\, \tilde{U} \right\rangle - F(\tilde{U}) \quad \text{for all } \tilde{U} \in L^2(\Omega)^n. \tag{5.28}$$

Comparing the standard linearization (5.22) from above with the perturbed and reduced linearization (5.28) here, we find that the right-hand sides are identical and that only the coefficient in the left-hand side Hessian operator has changed. The effects of this change are summarized in the next corollary.

Corollary 5.2.3 (Properties of the perturbed Hessian)**.** *The anisotropic second-order or fourth-order tensor component (depending on $n = d$ or $n = d \times d$) of the left-hand side Hessian operator in Equation* (5.28) *is*

$$\Xi_\epsilon : \mathbb{R}^n \to \mathbb{R}^n, \quad \Xi_\epsilon \coloneqq \mathbf{I} - \frac{U \otimes S}{|U|_\epsilon} = \Pi_\epsilon + \frac{U \otimes E(U,S)}{|U|_\epsilon}, \tag{5.29}$$

where we used definition (5.25) *of Π_ϵ and definition* (5.11) *of the model error $E(U,S)$. The oblique projector Ξ_ϵ approaches the near orthogonal projector Π_ϵ,*

$$\Xi_\epsilon \to \Pi_\epsilon \quad \textit{as} \quad E(U,S) \to 0,$$

and the magnitude of Ξ_ϵ *is bounded by* $|\Xi_\epsilon| \leq 2+|S|$*. Moreover, the perturbed Hessian in Equation* (5.28) *maps* U *to the model error* $E(U,S)$ *as* $\epsilon \to 0$*.*

Proof. Substituting the model error (5.11) gives

$$\Xi_\epsilon = \mathbf{I} - \frac{U \otimes S}{|U|_\epsilon} = \mathbf{I} - \frac{U}{|U|_\epsilon} \otimes \left(\frac{U}{|U|_\epsilon} - E(U,S) \right) = \mathbf{I} - \frac{U \otimes U}{|U|_\epsilon^2} + \frac{U \otimes E(U,S)}{|U|_\epsilon},$$

which is the alternate form of Ξ_ϵ in (5.29) due to the definition of Π_ϵ in (5.25). To derive a bound for the magnitude of Ξ_ϵ, let $V \in \mathbb{R}^n$ be arbitrary such that $|V| = 1$ and consider

$$\langle E(U,S)\,,\, V \rangle = \frac{\langle U\,,\, V \rangle}{|U|_\epsilon} - \langle S\,,\, V \rangle \leq 1 + |S|$$

and, additionally, the near orthogonal projector Π_ϵ has a magnitude bounded by one. To show the mapping of U to the model error $E(U,S)$ by the perturbed Hessian, we use property (5.26) of the unperturbed Hessian and obtain

$$\frac{1}{|U|_\epsilon} \left(\mathbf{I} - \frac{U \otimes U}{|U|_\epsilon^2} + \frac{U \otimes E(U,S)}{|U|_\epsilon} \right) U = \frac{\epsilon^2 U}{\left(|U|^2 + \epsilon^2 \right)^{3/2}} + \frac{|U|^2}{|U|_\epsilon^2} E(U,S) \to E(U,S) \quad \text{as} \quad \epsilon \to 0.$$

□

The properties of the perturbed Hessian from Corollary 5.2.3 describe a regularization of the unperturbed Hessian that is adapted to the model error $E(U,S)$. Since the model error is decreasing as U gets closer to the solution U^*, we find that the regularization is more pronounced while U is far from U^* and gets smaller as $U \to U^*$. Thus the regularization is adapting to the model error. In addition, the regularization is not isotropic but it is directed parallel to the model error. This results in improvements of the numerical stability of Newton's method, less backtracking at initial Newton steps, while maintaining a super-linear Newton convergence in a sufficiently small region around the solution U^*. The numerical evidence is presented in Section 5.4.

We close this section with practical notes. First, the perturbed Newton linearization only alters the coefficient in the left-hand side Hessian while keeping the right-hand side of the unperturbed linearization. This is advantageous for implementing the linearization into existing Newton solvers. Second, it is not required to allocate additional storage to track the dual variable S during nonlinear iterations, because it is not explicitly required in the update (see Equation (5.18) and the corresponding comment). Third, as a starting value for S, we can simply assume no model perturbation, $E(U,S) = 0$, and set $S = U/|U|_\epsilon$.

5.3 Applications and examples of perturbed linearizations

Example in 1D

We formulate a one-dimensional minimization problem: Find $x^* \in \mathbb{R}$ minimizing

$$\min_{x \in \mathbb{R}} J(x), \quad J(x) \coloneqq \frac{a}{1+\theta} |x|_\epsilon^{1+\theta} - bx, \tag{5.30}$$

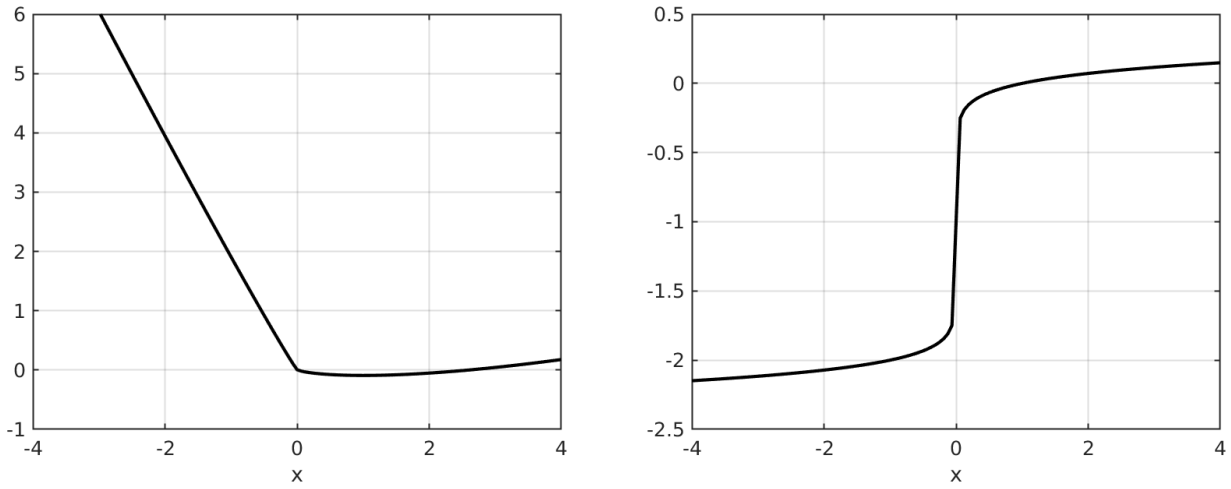


Figure 5.1: Objective functional in *left graph* and gradient in *right graph* of the 1D minimization problem (5.30) with constants $a, b = 1$, $\epsilon = 0.01$, and $\theta = 0.1$. The gradient varies slowly far away from $x = 0$, thereby far away from the solution $x^* \approx 1$, but it changes rapidly around $x = 0$.

where $a, b \in \mathbb{R}$ are given constants and $0 \leq \theta \leq 1$ is a constant parameter. θ controls the nonlinearity such that $J(x)$ is governed by a quadratic term for $\theta = 1$, therefore well-behaved, and by a term of norm-type for $\theta = 0$ with properties as discussed in Section 5.2. The gradient and Hessian for problem (5.30) are

$$g(x) \coloneqq J'(x) = \frac{ax}{|x|_\epsilon^{1-\theta}} - b, \quad H(x) \coloneqq J''(x) = \frac{a}{|x|_\epsilon^{1-\theta}} \left(1 - (1-\theta)\frac{x^2}{|x|_\epsilon^2}\right).$$

They constitute the standard Newton linearization for step $\hat{x}$, namely:

$$H(x)\,\hat{x} = -g(x) \quad \Leftrightarrow \quad \frac{a}{|x|_\epsilon^{1-\theta}} \left(1 - (1-\theta)\frac{x^2}{|x|_\epsilon^2}\right) \hat{x} = -\left(\frac{ax}{|x|_\epsilon^{1-\theta}} - b\right). \tag{5.31}$$

Figure 5.1 shows the objective functional $J(x)$ and the gradient $g(x)$. The gradient varies slowly away from $x = 0$, which is also far away from the solution $x^* \approx 1$, but it changes rapidly around $x = 0$. This property of the gradient indicates why the standard linearization (5.31) produces steps that are highly inaccurate in predicting the nonlinearities.

To derive the perturbed Newton linearization, we follow the procedure from the abstract framework in Section 5.2. Hence, we initially define the model error

$$E(x, y) \coloneqq \frac{x}{|x|_\epsilon^{1-\theta}} - y$$

and the corresponding model perturbation

$$D(x, y) \coloneqq x - |x|_\epsilon^{1-\theta}\, y$$

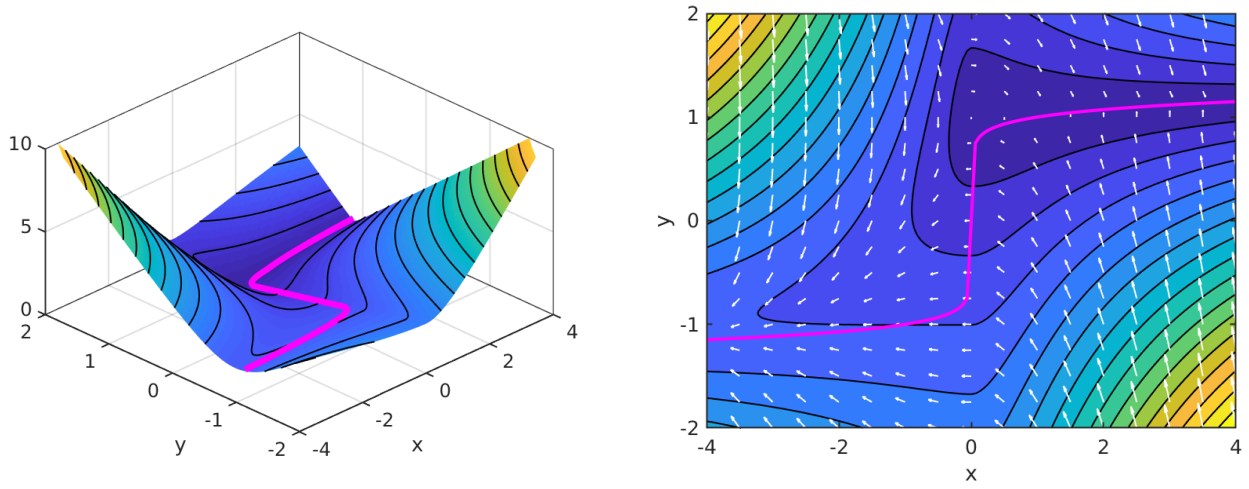


Figure 5.2: Magnitude of the perturbed (2D) gradient of (5.32) shown as a surface *(left)* and as contours *(right)*. The *arrows* in the *right plot* give direction and magnitude of the perturbed gradient. The *magenta curve* in both plots represents the absolute value of the unperturbed gradient in (x, y)-space such that $E(x, y) = 0$.

for $x, y \in \mathbb{R}$. This leads to the perturbed Newton linearization for step $\hat{x}$ and dual step $\hat{y}$,

$$a\hat{y} = -\left(ay - b\right), \tag{5.32a}$$

$$\left(1 - (1-\theta)\frac{xy}{|x|_\epsilon^{1+\theta}}\right)\hat{x} - |x|_\epsilon^{1-\theta}\,\hat{y} = -\left(x - |x|_\epsilon^{1-\theta}\,y\right), \tag{5.32b}$$

and by substituting for $\hat{y}$ analogously to Equation (5.17), we obtain the perturbed and reduced Newton linearization for step $\hat{x}$:

$$\frac{a}{|x|_\epsilon^{1-\theta}}\left(1 - (1-\theta)\frac{xy}{|x|_\epsilon^{1+\theta}}\right)\hat{x} = -\left(\frac{ax}{|x|_\epsilon^{1-\theta}} - b\right). \tag{5.33}$$

This simple 1D example allows us to visualize the perturbed linearization in Figure 5.2. We depict the magnitude of the perturbed (2D) gradient of (5.32) as a surface in the *left plot* and as contours in the *right plot* of Figure 5.2. Additionally on the *right*, the *arrows* give direction and magnitude of the perturbed gradient. The *magenta curve* represents the absolute value of the unperturbed gradient $g(x)$, i.e., on the manifold where $E(x, y) = 0$. It can be observed that the perturbation relaxes the rapid changes of the unperturbed $g(x)$ around $x = 0$, which can improve numerical stability for the computed Newton steps and convergence of the nonlinear solver.

Application to total variation (TV) regularization

Image restoration problems typically assume a given blurred and noisy image as observational data and a giving blurring operator. The goal is to find a solution that approximates the true image as well as possible. This problem is one example of an inverse problem. As discussed in Chapter 3, we need

to include a regularization term in order for the image restoration problem to be well posed. Total variation (TV) regularization offers the advantage that edges in the restored image are preserved [100], it has therefore been a popular choice.

We interpret an image as a function $u : \Omega \to \mathbb{R}$, supported on a domain $\Omega \subset \mathbb{R}^2$. Given are observational data $d \in L^2(\Omega)$ and a blurring operator $B : L^2(\Omega) \to L^2(\Omega)$ that is linear and bounded. Further, we define the TV operator by

$$\mathrm{TV}(u) \coloneqq \int_\Omega |\nabla u|_\epsilon = \int_\Omega \sqrt{|\nabla u|^2 + \epsilon^2}.$$

Note that for $\epsilon = 0$ the TV operator becomes a norm, called TV-norm, in fact $\mathrm{TV}(u) \to \|\nabla u\|_{L^1(\Omega)}$ as $\epsilon \to 0$. We formulate the minimization problem for image restoration: Find u^* minimizing

$$\min_u J(u), \quad J(u) \coloneqq \frac{1}{2} \|Bu - d\|_{L^2}^2 + \gamma\, \mathrm{TV}(u), \tag{5.34}$$

where the regularization $\gamma > 0$ was introduced.

We aim to solve (5.34) with Newton's method, which requires the gradient and Hessian operator to form the linearized system for a Newton step. Since our intention is to highlight analogies between the TV operator and derivations in Section 5.2, we focus on the first- and second-order variations of $\mathrm{TV}(u)$. These are

$$\delta_u[\mathrm{TV}(u)](\tilde{u}) = \int_\Omega \frac{\nabla u}{|\nabla u|_\epsilon} \cdot \nabla \tilde{u} = \left(\frac{\nabla u}{|\nabla u|_\epsilon}, \nabla \tilde{u} \right)$$

and

$$\delta_u \delta_u[\mathrm{TV}(u)](\tilde{u})(\hat{u}) = \left(\frac{1}{|\nabla u|_\epsilon} \left(\mathbf{I} - \frac{\nabla u \otimes \nabla u}{|\nabla u|^2 + \epsilon^2} \right) \nabla \hat{u}\,, \nabla \tilde{u} \right).$$

Taking variations of the remaining terms in (5.34), we obtain the standard linearized system for the Newton step $\hat{u}$:

$$\left(\frac{\gamma}{|\nabla u|_\epsilon} \left(\mathbf{I} - \frac{\nabla u \otimes \nabla u}{|\nabla u|^2 + \epsilon^2} \right) \nabla \hat{u}\,, \nabla \tilde{u} \right) + (B^* B \hat{u}\,, \tilde{u}) = -\left(\gamma \frac{\nabla u}{|\nabla u|_\epsilon}\,, \nabla \tilde{u} \right) - (B^*(Bu - d)\,, \tilde{u})\,. \tag{5.35}$$

Equation (5.35) has a Poisson operator on the left-hand side with an anisotropic second-order tensor coefficient with a projection-type property (see Corollary 5.2.2). This Poisson coefficient is problematic for the same reasons as we documented in an abstract setting for system (5.22) in Section 5.2. In that section, we also derived and analyzed a perturbed regularization (5.28) with improved properties.

We can now simply use the procedure for the abstract framework in Section 5.2. Hence, we begin by defining the model error for the TV operator

$$E(u, S) \coloneqq \frac{\nabla u}{|\nabla u|_\epsilon} - S$$

and the corresponding model perturbation

$$D(u, S) \coloneqq \nabla u - |\nabla u|_\epsilon\, S$$

for $S \in L^\infty(\Omega)$, $\|S\|_{L^\infty} \leq 1$. This leads to the perturbed and reduced linearized system for the Newton step $\hat{u}$:

$$\left(\frac{\gamma}{|\nabla u|_\epsilon}\left(\mathbf{I} - \frac{\nabla u \otimes S}{|\nabla u|_\epsilon}\right)\nabla\hat{u}\,,\,\nabla\tilde{u}\right) + (B^*B\hat{u}\,,\,\tilde{u}) = -\left(\gamma\frac{\nabla u}{|\nabla u|_\epsilon}\,,\,\nabla\tilde{u}\right) - (B^*(Bu-d)\,,\,\tilde{u})\,. \tag{5.36}$$

The perturbed linearization (5.36) can dramatically improve Newton convergence for image restoration problems with TV regularization [33,65]. Since its first presentation in [33], a different and less general derivation as in Section 5.2 was used and the procedure was called the *primal-dual Newton method*.

Application to yielding rheology in nonlinear Stokes flow

We went through the process of deriving a linearized Stokes system to solve for a Newton step in Section 5.1. In Equation (5.4) when yielding is active, i.e., $\theta = -1$, we encountered a case analogous to the Newton linearization in an abstract setting (Section 5.2). The standard linearization of the momentum equation (5.4a) in the yielding case is

$$-\nabla\cdot\left(2\mu(\boldsymbol{u})\left(\mathbf{I} - \frac{\nabla_{\mathrm{s}}\boldsymbol{u}\otimes\nabla_{\mathrm{s}}\boldsymbol{u}}{|\nabla_{\mathrm{s}}\boldsymbol{u}|_F^2}\right)\nabla_{\mathrm{s}}\hat{\boldsymbol{u}}\right) + \nabla\hat{p} = -\boldsymbol{r}_{\mathrm{mom}}, \tag{5.37}$$

where we assume that $0 < |\nabla_{\mathrm{s}}\boldsymbol{u}|_F$. Our previous observations implicate the same issues for (5.37) as discussed in Corollary 5.2.2 and that we can formulate an improved linearization by augmenting (5.37) with a perturbed model for the norm, which is $|\nabla_{\mathrm{s}}\boldsymbol{u}|_F$ here.

Therefore, analogously to the procedure for the abstract framework in Section 5.2, we define a model error

$$E(\boldsymbol{u},\boldsymbol{S}) \coloneqq \frac{\nabla_{\mathrm{s}}\boldsymbol{u}}{|\nabla_{\mathrm{s}}\boldsymbol{u}|_F} - \boldsymbol{S}$$

and the corresponding model perturbation

$$D(\boldsymbol{u},\boldsymbol{S}) \coloneqq \nabla_{\mathrm{s}}\boldsymbol{u} - |\nabla_{\mathrm{s}}\boldsymbol{u}|_F\,\boldsymbol{S}$$

for $\boldsymbol{u} \in H^1(\Omega)^d$ and $\boldsymbol{S} \in L^\infty(\Omega)^{d\times d}$, $\|\boldsymbol{S}\|_{L^\infty} \leq 1$. This leads to the perturbed and reduced linearized momentum equation:

$$-\nabla\cdot\left(2\mu(\boldsymbol{u})\left(\mathbf{I} - \frac{\nabla_{\mathrm{s}}\boldsymbol{u}\otimes\boldsymbol{S}}{|\nabla_{\mathrm{s}}\boldsymbol{u}|_F}\right)\nabla_{\mathrm{s}}\hat{\boldsymbol{u}}\right) + \nabla\hat{p} = -\boldsymbol{r}_{\mathrm{mom}}. \tag{5.38}$$

The improvement in Newton convergence using the perturbed linearization (5.38) is show in Section 5.4.

Finally, when solving the whole linearized Stokes system with momentum equation (5.38) using a Krylov iterative method, we enforce positive definiteness of the fourth-order tensor coefficient by projecting $\boldsymbol{S}$ into the unit ball such that $|\boldsymbol{S}|_F \leq 1$. In addition, it can be advantageous to the convergence of the linear solver, to symmetrize the outer product $\nabla_{\mathrm{s}}\boldsymbol{u}\otimes\boldsymbol{S}$ by replacing it with

$$\frac{1}{2}(\nabla_{\mathrm{s}}\boldsymbol{u} + \boldsymbol{S}) \otimes \frac{1}{2}(\nabla_{\mathrm{s}}\boldsymbol{u} + \boldsymbol{S}).$$

5.4 Numerical experiments for nonlinear Stokes flow

This section presents a benchmark model problem with a challenging yielding rheology, which we utilize in numerical experiments. The model problem is defined on the (open) unit cube domain $\Omega = (0,1)^3$, with a linear viscosity component $a(\boldsymbol{x}) \in [0, \mu_{\max}]$, $\boldsymbol{x} \in \Omega$, $0 < \mu_{\min} < \mu_{\max} < \infty$, which is generated by rescaling a C^∞ indicator function $\chi_n(\boldsymbol{x}) \in [1,2]$ that accumulates n plumes via a product of modified Gaussian functions:

$$
\begin{aligned}
a(\boldsymbol{x}) &\coloneqq (\mu_{\max} - \mu_{\min})(\chi_n(\boldsymbol{x}) - 1), \quad \boldsymbol{x} \in \Omega, \\
\chi_n(\boldsymbol{x}) &\coloneqq \prod_{i=1}^{n} 1 + \exp\left(-\delta \max\left(0, |\boldsymbol{c}_i - \boldsymbol{x}| - \frac{\omega}{2}\right)^2\right), \quad \boldsymbol{x} \in \Omega,
\end{aligned}
$$

where $\boldsymbol{c}_i \in \Omega$, $i = 1, \ldots, n$, are the centers of the plumes, $\delta > 0$ controls the exponential decay of the Gaussian smoothing, and $\omega \geq 0$ is the diameter of a plume where $\mu_{\min}$ is attained. Since all plumes are equal in size, inserting more of them inside the domain will eventually result in overlapping with each other and possible intersections with the domain's boundary. Throughout the section, we fix parameters $n = 4$, $\delta = 100$, $\omega = 0.1$, $\mu_{\min} = 10^{-2}$, $\mu_{\max} = 10^{+2}$, and use the same set of precomputed random points $\boldsymbol{c}_i$ in all numerical experiments. Note that we use a similar model viscosity when developing the weighted BFBT Schur complement preconditioner (see Section 6.2). The difference here is that plumes are inserted into the domain instead of sinkers and that the variations in the viscosity are not as severe since we focus on the nonlinearity and Newton convergence.

The right-hand side forcing is defined by $\boldsymbol{f}(\boldsymbol{x}) \coloneqq (0, 0, \beta(\chi_n(\boldsymbol{x}) - 1))$, $\beta = 100$ constant, such that it forces the low-viscosity plumes upward, similarly to a buoyancy that forces the rise of low-density inclusions within a medium of higher density.

The background viscosity of the medium surrounding the plumes attains the maximum value $\mu_{\max}$. In this part of the domain plastic yielding phenomena get activated by introducing the nonlinear viscosity

$$
\mu(\boldsymbol{x}, \dot{\varepsilon}_{\text{II}}) \coloneqq \min\left(\frac{\tau_{\text{yield}}}{2\dot{\varepsilon}_{\text{II}}}, a(\boldsymbol{x})\right) + \mu_{\min} \in [\mu_{\min}, \mu_{\max}], \tag{5.39}
$$

which contains a yield strength $\tau_{\text{yield}} > 0$. In our numerical experiments, we vary the yield strength to increase the yielding volume in the domain, i.e., where yielding is active, by lowering τ_{yield}.

The numerical results in Table 5.1 show a comparison of nonlinear Stokes solver convergence for the standard and the perturbed Newton linearizations found in Equations (5.37) and (5.38), respectively. For the viscosity (5.39), we vary the nonlinearity via yield strength τ_{yield}, which is expressed in different relative yielding volumes in the table. To additionally test for algorithmic scalability, the level ℓ of refinement of the uniform mesh is refined. We utilize the $\mathbb{Q}_2 \times \mathbb{P}_1^{\text{disc}}$ finite element pairing for velocity and pressure. We observe for the perturbed linearization that the yielding volume only moderately affects the number of Newton iterations until convergence (10^{-7} nonlinear residual reduction) and that the iteration count is largely unaffected by the mesh refinement level. Accordingly, also the total sum of backtracking iterations remains stable across refinements. In contrast, the standard

Table 5.1: Comparison of standard and perturbed Newton linearizations using the nonlinear viscosity (5.39) while varying yield strength τ_{yield}, expressed in different relative yielding volumes, and level ℓ of refinement of the uniform mesh ($\mathbb{Q}_2 \times \mathbb{P}_1^{\text{disc}}$ finite element pairing). For each type of linearization we state the number of Newton iterations until convergence (10^{-7} nonlinear residual reduction), the total count of backtracking iterations (each reducing a Newton step by the factor 0.5), and the total number of GMRES iterations. The perturbed linearization demonstrates a significant improvement in stability of Newton's method and overall computational cost (total GMRES iterations).

Yielding	**Mesh**	**Regular Newton**			**Perturbed Newton**		
volume	**level ℓ**	It. Newton	#backtr.	It. GMRES	It. Newton	#backtr.	It. GMRES
∼45%	4	33	20	1469	10	0	379
∼45%	5	36	25	2255	12	0	664
∼45%	6	57	49	4255	13	0	876
∼65%	4	29	21	1559	18	10	965
∼65%	5	37	26	2464	17	9	1245
∼65%	6	48	39	3892	20	9	1707
∼90%	4	35	25	1505	19	11	872
∼90%	5	40	32	2147	21	11	1267
∼90%	6	32	21	2312	23	11	1811

Newton linearization behaves in an unstable manner with fluctuating Newton iteration counts and vastly more backtracking. Therefore, the overall computational cost, which is determined by the total count of GMRES iterations, is significantly higher than for the perturbed linearization. The numerical experiments confirm the previous theoretical analysis that the perturbation adds stability to Newton's method, improves the computed Newton step directions, and results in a robust and fast convergence for these class of problems.

6

Schur Complement Preconditioning with Weighted BFBT

The global mantle convection applications we target exhibit the difficulties and computational challenges described above in Chapter 4 (severe heterogeneity, very large scale, need for aggressively-adapted meshes, need for high-order, mass-conserving discretization). Hence, they demand robust and effective preconditioners for (4.6), resulting in iterative solvers with optimal (or nearly optimal) algorithmic and parallel scalability. This chapter and [98, 99] describes the design of such a preconditioner and its analysis and performance evaluation for Stokes problems with highly heterogeneous viscosity. The preconditioner—which we call *weighted BFBT* (w-BFBT)—is of Schur complement type, and we study its robustness as well as its algorithmic and parallel scalability.

6.1 Introduction to Schur complement approximations

An effective approximation of the Schur[1] complement $\mathbf{S} \coloneqq \mathbf{B}\mathbf{A}^{-1}\mathbf{B}^{\mathsf{T}}$ is an essential ingredient for attaining fast convergence of Schur complement-based iterative solvers for the linear algebraic Stokes system in Equation (4.6). More precisely, a sufficiently good approximation of the inverse Schur complement, $\tilde{\mathbf{S}}^{-1} \approx \mathbf{S}^{-1}$, is sought. As we have outlined in Section 4.2, approximation $\tilde{\mathbf{S}}^{-1}$ is combined with an approximation of the inverse viscous block, $\tilde{\mathbf{A}}^{-1} \approx \mathbf{A}^{-1}$, in an iterative scheme with right preconditioning based on an upper triangular block matrix:

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}^{\mathsf{T}} \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{A}} & \mathbf{B}^{\mathsf{T}} \\ \mathbf{0} & \tilde{\mathbf{S}} \end{bmatrix}^{-1} \begin{bmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{p}} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix}. \tag{6.1}$$

Note that the original solution to (4.6) is recovered by applying the preconditioner once to the solution of (6.1).

The most widely used approximation of the Schur complement (for variable viscosity Stokes systems) is the inverse viscosity-weighted mass matrix of the pressure space [29, 71, 78, 84], denoted

[1] Strictly speaking, our definition is the negative Schur complement. However, as in [46], we prefer to work with positive-definite operators and thus define the Schur complement to be positive rather than negative definite.

by $\mathbf{M}_p(1/\mu)$, with entries $[\mathbf{M}_p(1/\mu)]_{i,j} = \int_\Omega q_i(\boldsymbol{x})\, q_j(\boldsymbol{x})/\mu(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$, where $q_i, q_j \in \mathbb{P}_{k-1}^{\mathrm{disc}}$ are global basis functions of the finite dimensional space $\mathbb{P}_{k-1}^{\mathrm{disc}}$. Since the basis functions of $\mathbb{P}_{k-1}^{\mathrm{disc}}$ are modal and not orthogonal to each other, the mass matrix is not diagonal, and thus $\mathbf{M}_p(1/\mu)$ is typically diagonalized to further simplify its inversion. One common way to obtain a diagonalized version is mass lumping. For nodal discretizations, the corresponding diagonal elements are computed by summation of the entries of each matrix row, i.e., $\mathbf{M}_p(1/\mu)\mathbf{1}$, where $\mathbf{1}$ is the vector with ones in all entries. For modal discretizations, we generalize the lumping procedure by using the coefficient vector, $\mathbf{1}_{\{q_i\}_i}$, representing the constant function having value 1 in the associated basis $\{q_i\}_i$, i.e.,

$$\tilde{\mathbf{M}}_p(1/\mu) \coloneqq \mathrm{diag}(\mathbf{M}_p(1/\mu)\mathbf{1}_{\{q_i\}_i}). \tag{6.2}$$

Provided that μ is sufficiently smooth, $\mathbf{M}_p(1/\mu)$ can be an effective approximation of $\mathbf{S}$ in numerical experiments [28] and spectral equivalence can be shown [57]. However, it has been observed in applications with highly heterogeneous viscosities μ (e.g., mantle convection [85, 97]) that convergence slows down significantly due to a poor Schur complement approximation by $\mathbf{M}_p(1/\mu)$. Therefore, we propose a new approximation, w-BFBT, that remains robust when $\mathbf{M}_p(1/\mu)$ fails.

Preconditioners based on BFBT approximations for the Schur complement were initially proposed in [43] for the Navier–Stokes equations. Over the years, these ideas were refined and extended [44, 45, 47, 76, 105] to arrive at a class of closely related Schur complement approximations: Pressure Convection–Diffusion, BFBT, and Least Squares Commutator. The underlying principle, now in a Stokes setting, is that one seeks a commutator matrix $\mathbf{X}$ such that the following commutator nearly vanishes,

$$\mathbf{A}\mathbf{D}^{-1}\mathbf{B}^{\mathsf{T}} - \mathbf{B}^{\mathsf{T}}\mathbf{X} \approx \mathbf{0}, \tag{6.3}$$

for a given diagonal matrix $\mathbf{D}^{-1}$. The Navier–Stokes case differs from Stokes in that the viscous stress matrix $\mathbf{A}$ contains an additional convection term. The motivation for seeking a near-commutator $\mathbf{X}$ is that (6.3) can be rearranged by multiplying (6.3) with $\mathbf{B}\mathbf{A}^{-1}$ from the left and, provided the inverse exists, with $\mathbf{X}^{-1}$ from the right to obtain $\mathbf{S} \approx \mathbf{B}\mathbf{D}^{-1}\mathbf{B}^{\mathsf{T}}\mathbf{X}^{-1}$, where the closer the commutator is to zero, the more accurate the approximation [46]. The goal of finding a vanishing commutator can be recast as solving the following least-squares minimization problem:

$$\text{Find matrix } \mathbf{X} \text{ minimizing} \quad \left\| \mathbf{A}\mathbf{D}^{-1}\mathbf{B}^{\mathsf{T}}\mathbf{e}_j - \mathbf{B}^{\mathsf{T}}\mathbf{X}\mathbf{e}_j \right\|_{\mathbf{C}^{-1}}^2 \quad \text{for all } j, \tag{6.4}$$

where $\mathbf{e}_j$ is the j-th Cartesian unit vector and the norm arises from a symmetric and positive definite matrix $\mathbf{C}$. The solution is given by $\mathbf{X} = (\mathbf{B}\mathbf{C}^{-1}\mathbf{B}^{\mathsf{T}})^{-1}(\mathbf{B}\mathbf{C}^{-1}\mathbf{A}\mathbf{D}^{-1}\mathbf{B}^{\mathsf{T}})$. Then the BFBT approximation of the inverse Schur complement is derived by algebraic rearrangement of the commutator (6.3):

$$\tilde{\mathbf{S}}_{\mathrm{BFBT}}^{-1} \coloneqq \left(\mathbf{B}\mathbf{C}^{-1}\mathbf{B}^{\mathsf{T}}\right)^{-1} \left(\mathbf{B}\mathbf{C}^{-1}\mathbf{A}\mathbf{D}^{-1}\mathbf{B}^{\mathsf{T}}\right) \left(\mathbf{B}\mathbf{D}^{-1}\mathbf{B}^{\mathsf{T}}\right)^{-1}. \tag{6.5}$$

In the literature cited above (which addresses preconditioning of Navier–Stokes equations with constant viscosity), the diagonal weighting matrices are chosen as $\mathbf{C} = \mathbf{D} = \tilde{\mathbf{M}}_{\boldsymbol{u}}$, i.e., a diagonalized version of

the velocity space mass matrix; hence we call this the $\mathbf{M}_{\boldsymbol{u}}$-BFBT approximation of the Schur complement. $\mathbf{M}_{\boldsymbol{u}}$-BFBT can be used for Stokes problems with constant viscosities providing convergence similar to $\mathbf{M}_p(1/\mu)$. However, the computational cost of applying (6.5) is significantly higher than the (cheap) application of a possibly diagonalized inverse of $\mathbf{M}_p(1/\mu)$. Moreover, $\mathbf{M}_{\boldsymbol{u}}$-BFBT is not an option for heterogeneous viscosities because convergence becomes extremely slow or stagnates, as observed in [85]. Instead, in [85] for finite element and in [52] for staggered grid finite difference discretizations, a re-scaling of the discrete Stokes system (4.6) was performed, which essentially alters the diagonal weighting matrices $\mathbf{C}$, $\mathbf{D}$. By choosing entries from $\mathbf{A}$ for these weighting matrices, it was possible to demonstrate improved convergence with BFBT compared to $\mathbf{M}_p(1/\mu)$ for certain benchmark problems with strong viscosity variations. Building on ideas from [85], [97] chose the weighting matrices such that $\mathbf{C} = \mathbf{D} = \text{diag}(\mathbf{A})$, which led to superior performance compared to $\mathbf{M}_p(1/\mu)$ for highly heterogeneous mantle convection problems. Hence we refer to this approach as diag($\mathbf{A}$)-BFBT.

However, even diag($\mathbf{A}$)-BFBT can fail to achieve fast convergence for some problems and/or discretizations, as shown below (Section 6.2). Moreover, choosing the weighting matrices as diag($\mathbf{A}$) is problematic for high-order discretizations, where diag($\mathbf{A}$) becomes a poor approximation of $\mathbf{A}$. These drawbacks lead us to propose the following w-BFBT approximation for the inverse Schur complement:

$$\tilde{\mathbf{S}}^{-1}_{\text{w-BFBT}} \coloneqq \left(\mathbf{B}\mathbf{C}^{-1}_{w_l}\mathbf{B}^{\mathsf{T}}\right)^{-1}\left(\mathbf{B}\mathbf{C}^{-1}_{w_l}\mathbf{A}\mathbf{D}^{-1}_{w_r}\mathbf{B}^{\mathsf{T}}\right)\left(\mathbf{B}\mathbf{D}^{-1}_{w_r}\mathbf{B}^{\mathsf{T}}\right)^{-1}, \tag{6.6}$$

where $\mathbf{C}_{w_l} = \tilde{\mathbf{M}}_{\boldsymbol{u}}(w_l)$ and $\mathbf{D}_{w_r} = \tilde{\mathbf{M}}_{\boldsymbol{u}}(w_r)$ are lumped velocity space mass matrices (lumping analogously to (6.2)) that are weighted by the square root of the viscosity, $w_l(\boldsymbol{x}) = \sqrt{\mu(\boldsymbol{x})} = w_r(\boldsymbol{x})$, $\boldsymbol{x} \in \Omega$.

Outline and summary of key results

After defining a class of benchmark problems (Section 6.2), we compare the convergence obtained with different Schur complement approximations to motivate preconditioning with w-BFBT (Section 6.2). Theoretical estimates for spectral equivalence of w-BFBT are derived in Section 6.3. This is followed by a detailed numerical study showing when w-BFBT is advantageous over $\mathbf{M}_p(1/\mu)$ (Section 6.4), and a discussion of boundary modifications for w-BFBT that accelerate convergence (Section 6.5). In Section 7.2 we describe an algorithm for w-BFBT-based Stokes preconditioning, which uses hybrid spectral–geometric–algebraic multigrid (HMG). Finally, in Chapter 8, we provide numerical evidence for near-optimal algorithmic and parallel scalability. In particular, we demonstrate that the preconditioner's parallel efficiency remains high when weak scaling out to tens of thousands of threads and even millions of threads.

To motivate our study of w-BFBT, we give an example for a possible improvement in convergence in Figure 6.1. There, a comparison is drawn between the $\mathbf{M}_p(1/\mu)$ and w-BFBT approximations for the Schur complement. The Stokes problem that is being solved is the multi-sinker benchmark problem from Section 6.2. The difficulty of the problem can be increased by adding more and more

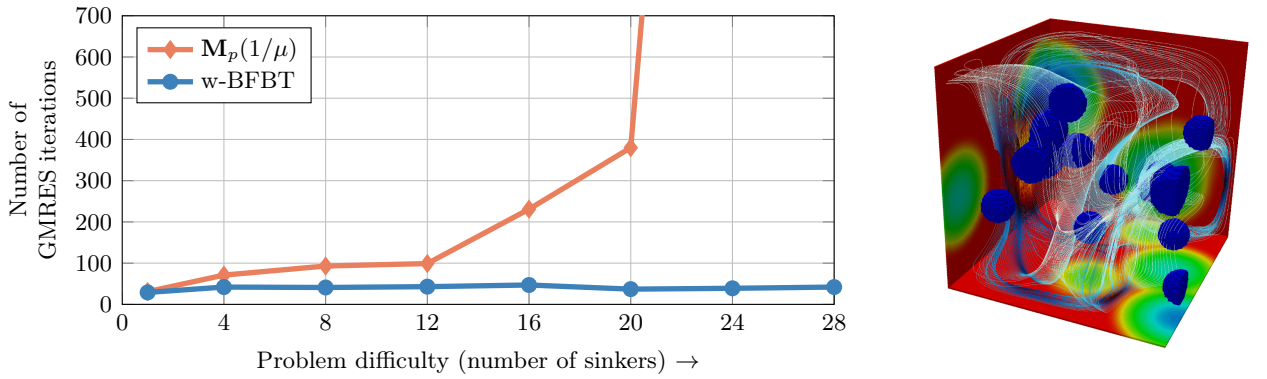


Figure 6.1: *Left image* shows the improvement in convergence obtained with the proposed w-BFBT preconditioner over a preconditioner using the inverse viscosity-weighted pressure mass matrix as Schur complement approximation. The number of randomly placed sinkers (high viscosity inclusions in low viscosity medium) increases along the horizontal axis. The vertical axis depicts the number of GMRES iterations required for 10^6 residual reduction for the most popular $\mathbf{M}_p(1/\mu)$ and the proposed w-BFBT preconditioner. Fixed problem parameters are the dynamic ratio $\mathrm{DR}(\mu) = \max(\mu)/\min(\mu) = 10^8$, discretization order $k = 2$, and the mesh refinement level $\ell = 7$, resulting in 128^3 finite elements. *Right image* shows an example viscosity field with 16 sinkers (*blue spheres* depict highly viscous centers of Gaussian-like sinkers, low viscosity medium in *red color*) and the *streamlines* of the computed velocity field.

high-viscosity inclusions, called sinkers, into a low-viscosity background medium, which, as a result, introduces more variation in the viscosity. As can be seen in the figure, the number of GMRES iterations remain flat when preconditioning with w-BFBT, whereas the number of GMRES iterations increases significantly for higher sinker counts when using $\mathbf{M}_p(1/\mu)$, rendering $\mathbf{M}_p(1/\mu)$ inefficient for these types of difficult problems. Therefore we propose w-BFBT as an alternative Schur complement approximation for Stokes flow problems with a highly varying viscosity.

6.2 Benchmark problem and comparison of Schur complement approximations

This section further motivates the need for more effective Schur complement preconditioners. We first present a class of benchmark problems that range from relatively mild viscosity variations to severely heterogeneous. Then a challenging problem is used to compare Stokes solver convergence with different Schur complement approximations to demonstrate the limitations of established methods and motivate the development of w-BFBT.

Multi-sinker benchmark problem

The design of suitable benchmark problems is critical to conduct studies that can give useful convergence estimates for challenging applications. We seek complex geometrical structures in the viscosity that

generate irregular, nonlocal, multiscale flow fields. Additionally, the viscosity should exhibit sharp gradients and its dynamic ratio $\mathrm{DR}(\mu) \coloneqq \max(\mu)/\min(\mu)$ (also commonly referred to as viscosity contrast) can be six orders of magnitude or higher in demanding applications. As in [83], we use a multi-sinker test problem with randomly positioned inclusions (e.g., as in Figure 6.1, *right image*) to study solver performance. We find that the arising viscosity structure is a suitable test for challenging, highly heterogeneous coefficient Stokes problems, and that the solver performance observed for such models can be indicative of the performance for other challenging applications.

In the (open) unit cube domain $\Omega = (0,1)^3$, we define the viscosity coefficient $\mu(\boldsymbol{x}) \in [\mu_{\min}, \mu_{\max}]$, $\boldsymbol{x} \in \Omega$, $0 < \mu_{\min} < \mu_{\max} < \infty$, with dynamic ratio $\mathrm{DR}(\mu) = \mu_{\max}/\mu_{\min}$ by means of rescaling a C^∞ indicator function $\chi_n(\boldsymbol{x}) \in [0,1]$ that accumulates n sinkers via a product of modified Gaussian functions:

$$\begin{aligned}
\mu(\boldsymbol{x}) &\coloneqq (\mu_{\max} - \mu_{\min})(1 - \chi_n(\boldsymbol{x})) + \mu_{\min}, \quad \boldsymbol{x} \in \Omega, \\
\chi_n(\boldsymbol{x}) &\coloneqq \prod_{i=1}^{n} 1 - \exp\left(-\delta \max\left(0, |\boldsymbol{c}_i - \boldsymbol{x}| - \frac{\omega}{2}\right)^2\right), \quad \boldsymbol{x} \in \Omega,
\end{aligned}$$

where $\boldsymbol{c}_i \in \Omega$, $i = 1, \ldots, n$, are the centers of the sinkers, $\delta > 0$ controls the exponential decay of the Gaussian smoothing, and $\omega \geq 0$ is the diameter of a sinker where $\mu_{\max}$ is attained. Since all sinkers are equal in size, inserting more of them inside the domain will eventually result in overlapping with each other and possible intersections with the domain's boundary. Throughout the chapter, we fix $\delta = 200$, $\omega = 0.1$, and use the same set of precomputed random points $\boldsymbol{c}_i$ in all numerical experiments. Two parameters are varied: (i) the number of sinkers n at random positions (the label Sn-rand indicates a multi-sinker problem with n randomly positioned sinkers) and (ii) the dynamic ratio $\mathrm{DR}(\mu)$ which in turn determines $\mu_{\min} \coloneqq \mathrm{DR}(\mu)^{-1/2}$ and $\mu_{\max} \coloneqq \mathrm{DR}(\mu)^{1/2}$. The right-hand side of (4.6), $\boldsymbol{f}(\boldsymbol{x}) \coloneqq (0, 0, \beta(\chi_n(\boldsymbol{x}) - 1))$, $\beta = 10$ constant, is such that it forces the high-viscosity sinkers downward, similarly to a gravity that pulls on high-density inclusions within a medium of lower density.

Comparison of Schur complement approximations

We compare convergence of the Stokes solver using the Schur complement approximation $\mathbf{M}_p(1/\mu)$ with diag($\mathbf{A}$)-BFBT and with the proposed w-BFBT. The problem parameters are held fixed to S16-rand and $\mathrm{DR}(\mu) = 10^8$. The numerical experiments are carried out using different levels of mesh refinement $\ell = 5, \ldots, 7$ (for fixed order $k = 2$) and different discretization orders $k = 2, \ldots, 5$ (for fixed level $\ell = 5$). A level ℓ corresponds to a mesh of $2^{3\ell}$ elements due to uniform refinement. Note that for these tests, the applications of $\mathbf{A}^{-1}$, $\left(\mathbf{B}\mathbf{C}_{w_l}^{-1}\mathbf{B}^\mathsf{T}\right)^{-1}$, and $\left(\mathbf{B}\mathbf{D}_{w_r}^{-1}\mathbf{B}^\mathsf{T}\right)^{-1}$ are approximated using a multigrid method (introduced in Section 7.2). These approximations are sufficiently accurate, such that the comparison is indicative of the effectiveness of the different Schur complement approximations. In particular, improving the approximation does not change the results, which are presented in Figure 6.2.

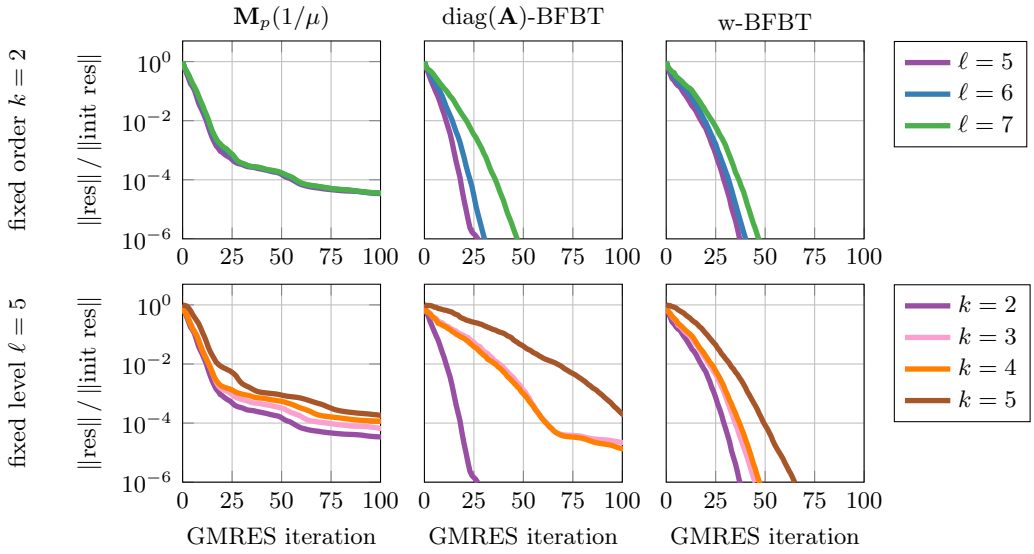


Figure 6.2: Comparison of Stokes solver convergence with $\mathbf{M}_p(1/\mu)$ *(left column)*, diag($\mathbf{A}$)-BFBT *(middle column)*, and w-BFBT *(right column)* preconditioning. We fix the problem S16-rand, $\text{DR}(\mu) = 10^8$ while varying mesh refinement level ℓ *(top row)* and discretization order k *(bottom row)*. This comparison shows that w-BFBT combines robust convergence of diag($\mathbf{A}$)-BFBT with improved algorithmic scalability when k is increased.

In the *left* two plots, the poor Schur complement approximation by $\mathbf{M}_p(1/\mu)$ for this problem setup can be observed clearly. Convergence stagnates (similar results are found in [83, 97]).

Preconditioner diag($\mathbf{A}$)-BFBT (Figure 6.2, *middle*) is able to achieve fast convergence for discretization order $k = 2$. A limitation of diag($\mathbf{A}$)-BFBT is a strong dependence on the order k. This can be explained by the decreasing diagonal dominance in the viscous block $\mathbf{A}$ with increasing order k: for higher k the approximation of $\mathbf{A}$ by diag($\mathbf{A}$) deteriorates. Note that numerical experiments with $\mathbf{M}_{\boldsymbol{u}}$-BFBT are not presented, because it performs poorly in the presence of spatially-varying viscosities. This leads to the conclusion that the choice of the weighting matrices $\mathbf{C}$, $\mathbf{D}$ in $\tilde{\mathbf{S}}^{-1}_{\text{BFBT}}$ crucially affects the quality of the Schur complement approximation.

The w-BFBT approximation delivers convergence that is nearly as fast as in the diag($\mathbf{A}$)-BFBT, $k = 2$ case, but without the severe deterioration when k is increased (see Figure 6.2, *right*). Thus, w-BFBT exhibits the robustness of diag($\mathbf{A}$)-BFBT and additionally shows superior algorithmic scalability with respect to k. Having illustrated the efficacy of w-BFBT for certain problem parameters, we next establish spectral equivalence of w-BFBT (Section 6.3) and then analyze in more detail how crucial parameters influence convergence in Section 6.4.

6.3 Spectral equivalence of w-BFBT

Before we show spectral equivalence, we introduce notation and basic definitions.

Basic definitions

Let $\Omega \subset \mathbb{R}^d$ be a bounded domain with Lipschitz boundary $\partial\Omega$. We denote as $L^2(\Omega)$ the class of real-valued square-integrable functions, equipped with the usual L^2-inner product $(u\,,v)_{L^2(\Omega)}$ and induced norm $\|u\|_{L^2(\Omega)}$, $u, v \in L^2(\Omega)$. We also consider corresponding spaces of d-dimensional vector-valued functions and $(d \times d)$-dimensional tensor-valued functions with component-wise multiplication, denoted by $L^2(\Omega)^d$ and $L^2(\Omega)^{d\times d}$. The subspace of $L^2(\Omega)$ that does not contain constant functions is denoted by $L^2(\Omega)/\mathbb{R}$. A bounded function, say $\alpha = \alpha(\boldsymbol{x})$, belongs to the space $L^\infty(\Omega)$ by satisfying the following finite norm: $\|\alpha\|_{L^\infty(\Omega)} \coloneqq \operatorname{ess\,sup}_{\boldsymbol{x}\in\Omega} |\alpha(\boldsymbol{x})| < \infty$. We generalize the L^2-norms to classes of weighted L^2_α-norms for functions $f \in L^2(\Omega)^n$, $n \in \{1, d, d \times d\}$, defined by

$$\|f\|_{L^2_\alpha(\Omega)^n} \coloneqq \left\|\alpha^{\frac{1}{2}} f\right\|_{L^2(\Omega)^n} \quad \text{for } \alpha \in L^\infty(\Omega),\ 0 < \alpha(\boldsymbol{x}) \text{ a.e. in } \Omega.$$

Next, we introduce $H^m(\Omega)$, with $m \geq 0$, which is the Sobolev space of m derivatives in $L^2(\Omega)$, and for $m = 1$ we use the inner product $(u\,,v)_{H^1(\Omega)} \coloneqq (u\,,v)_{L^2(\Omega)} + (\nabla u\,,\nabla v)_{L^2(\Omega)}$, inducing the norm $\|u\|_{H^1(\Omega)}$. Functions in $H^m(\Omega)$ with vanishing trace on the boundary $\partial\Omega$ belong to the space $H^m_0(\Omega)$. Finally, we say that a function belongs to the class of $C^\infty(\Omega)$ if it has partial derivatives of any order in Ω, and these derivatives are continuous.

We transition from abstract definitions to fluid mechanics. The differential operators acting on velocity $\boldsymbol{u} \in H^1(\Omega)^d$ and pressure $p \in L^2(\Omega)$ within the Stokes equations are defined in the sense of distributions:

$$\nabla_{\mathrm{s}}\boldsymbol{u} \coloneqq \frac{1}{2}(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^{\mathsf{T}}), \quad B\boldsymbol{u} \coloneqq -\nabla\cdot\boldsymbol{u}, \quad B^*p \coloneqq \nabla p.$$

Moreover, assume a sufficiently regular, bounded viscosity $\mu \in H^1(\Omega) \cap L^\infty(\Omega)$ such that $0 < \mu_{\min} \leq \mu(\boldsymbol{x})$ a.e. in Ω and then define the viscous stress tensor $\boldsymbol{\tau} \coloneqq 2\mu\nabla_{\mathrm{s}}\boldsymbol{u}$. We denote the function space for velocity by

$$V \coloneqq \left\{\boldsymbol{u} \in (H^1(\Omega))^d \mid \boldsymbol{n}\cdot\boldsymbol{u} = 0 \text{ on } \partial\Omega\right\}, \tag{6.7}$$

where $\boldsymbol{n} \in \mathbb{R}^d$ is the outward unit normal vector at the boundary $\partial\Omega$, and the function space for pressure by $Q \coloneqq L^2(\Omega)/\mathbb{R}$, and we introduce the viscous stress operator with a heterogeneous viscosity

$$A_\mu : V \to V', \quad A_\mu\boldsymbol{u} \coloneqq -\nabla\cdot(2\mu\nabla_{\mathrm{s}}\boldsymbol{u}) = -\nabla\cdot\boldsymbol{\tau}.$$

Given exterior forces acting on the fluid $\boldsymbol{f} \in V'$, we consider the incompressible Stokes problem with free-slip and no-normal flow boundary conditions:

$$A_\mu\boldsymbol{u} + B^*p = \boldsymbol{f} \quad \text{in } \Omega, \tag{6.8a}$$

$$B\boldsymbol{u} = 0 \quad \text{in } \Omega, \tag{6.8b}$$

$$\boldsymbol{T}\left[\boldsymbol{\tau} - p\mathbf{I}\right]\boldsymbol{n} = 0 \quad \text{on } \partial\Omega, \tag{6.8c}$$

$$\boldsymbol{u}\cdot\boldsymbol{n} = 0 \quad \text{on } \partial\Omega, \tag{6.8d}$$

in which we seek the velocity $\boldsymbol{u} \in V$ and pressure $p \in Q$. On the boundary, we have outward unit normal vectors $\boldsymbol{n} \in \mathbb{R}^d$ and tangential projectors $\boldsymbol{T} \coloneqq \mathbf{I} - \boldsymbol{n}\boldsymbol{n}^{\mathsf{T}}$.

For the definition of the w-BFBT approximation of the Schur complement, we introduce a Poisson operator for higher regularity pressure functions

$$K_w^* : H^2(\Omega) \to L^2(\Omega), \quad K_w^* p \coloneqq BwB^* p, \tag{6.9}$$

with an appropriate coefficient w (see below) and augmented with homogeneous Neumann boundary conditions, $\boldsymbol{n} \cdot B^* p = 0$. The L^2-adjoint of K_w^* is denoted by K_w. Finally, we define the w-BFBT approximation of the Schur complement $S = BA_\mu^{-1}B^*$ by:

$$\tilde{S}_{\text{w-BFBT}} \coloneqq K_{w_r}^* (Bw_l \, A_\mu \, w_r B^*)^{-1} K_{w_l}, \tag{6.10}$$

with sufficiently regular, bounded weight functions $w_l, w_r \in H^1(\Omega) \cap L^\infty(\Omega)$ such that $0 < w_{\min} \leq w_l(\boldsymbol{x}), w_r(\boldsymbol{x})$ a.e. in Ω. Note that the definitions of the w-BFBT weight functions in the discrete case (6.6) are reciprocal to definitions in (6.9) and (6.10), because in the discrete case the weight functions were embedded into inverses of mass matrices.

Main theorem on spectral equivalence of w-BFBT

One measure for the efficacy of a preconditioner consists of the ratio of the maximal to minimal eigenvalues of the preconditioned system $\tilde{\mathbf{S}}_{\text{w-BFBT}}^{-1}\mathbf{S}$. This section establishes inequalities for spectral equivalence of w-BFBT by providing bounds on that ratio. The derivations are carried out in an infinite-dimensional setting. We begin by stating the main result of this section in Theorem 6.3.1 and continue with proving this result using a sequence of lemmas.

Theorem 6.3.1 (Main result)**.** *Let $\hat{Q} = L^2(\Omega)/\mathbb{R} \cap H^1(\Omega)$. If the left and right w-BFBT weight functions are equal to*

$$w_l(\boldsymbol{x}) = \mu(\boldsymbol{x})^{-\frac{1}{2}} = w_r(\boldsymbol{x}) \quad \text{for a.a. } \boldsymbol{x} \in \Omega,$$

then the exact Schur complement is equivalent to the w-BFBT approximation such that

$$\left(\tilde{S}_{\text{w-BFBT}}\, q \,,\, q\right) \leq (Sq \,,\, q) \leq C_{\text{w-BFBT}} \left(\tilde{S}_{\text{w-BFBT}}\, q \,,\, q\right) \quad \text{for all } q \in \hat{Q},$$

where

$$C_{\text{w-BFBT}} \coloneqq \left(1 + \frac{1}{4} \|\nabla \mu\|^2_{L^\infty(\Omega)^d}\right)\left(C_{P,\mu}^2 + 1\right) C_{K,\mu}^2$$

and the constants $C_{P,\mu}, C_{K,\mu} > 0$ stem from weighted Poincaré–Friedrichs' and Korn's inequalities, respectively (see Remark 6.3.7 for more information); the viscosity μ assumes the role of the weight function in the weighted inequalities.

If the viscosity and the w-BFBT weight functions are constant,

$$\mu \equiv 1, \quad w_l \equiv 1 \equiv w_r,$$

then the exact Schur complement is equivalent to the w-BFBT approximation such that

$$\left(\tilde{S}_{\text{w-BFBT}}\, q \,,\, q\right) \leq (Sq \,,\, q) \leq (C_P^2 + 1)\, C_K^2 \left(\tilde{S}_{\text{w-BFBT}}\, q \,,\, q\right) \quad \text{for all } q \in \hat{Q}$$

with constants $C_P, C_K > 0$ stemming from (the classical) Poincaré–Friedrichs' and Korn's inequalities, respectively.

Proofs

The proof of Theorem 6.3.1 is established in the remainder of this section. In what follows, suprema are understood over spaces excluding operator kernels that would cause a supremum to blow up. The following basic, but hereafter frequently used, result is shown for completeness of the discussion.

Lemma 6.3.2 (sup-form of inverse operator)**.** *Let V be a complete Hilbert space and $W \subseteq V$ be a dense subspace. Assume the linear operator $T : V \to V'$ to be bounded, invertible, symmetric, and positive definite. Then for any $f \in V'$ follows*

$$\left(T^{-1} f \,,\, f\right) = \sup_{w \in W} \frac{(w \,,\, f)^2}{(Tw \,,\, w)}.$$

Proof. Let $w \in W$, then with Hölder's inequality follows

$$(w \,,\, f)^2 = \left(T^{1/2} w \,,\, T^{-1/2} f\right)^2 \leq \left\| T^{1/2} w \right\|^2 \left\| T^{-1/2} f \right\|^2 = (Tw \,,\, w) \left(T^{-1} f \,,\, f\right).$$

Additionally, let $v = T^{-1} f$ and since $W \subseteq V$ dense, there exists a sequence $\{w_k\}_k \subset W$ such that $w_k \to v = T^{-1} f \in V$, hence

$$\frac{(w_k \,,\, f)^2}{(Tw_k \,,\, w_k)} \to \frac{(v \,,\, f)^2}{(Tv \,,\, v)} = \frac{\left(T^{-1} f \,,\, f\right)^2}{(f \,,\, T^{-1} f)} = \left(T^{-1} f \,,\, f\right),$$

which shows that equality is achieved in the limit. □

The next lemma establishes Schur complement properties that are essential for deriving lower and upper bounds in the spectral equivalence estimates.

Lemma 6.3.3 (sup-form of Schur complement)**.** *With the definitions from Section 6.3, the following two equalities hold:*

$$\left(\tilde{S}_{\text{w-BFBT}}\, q \,,\, q\right) = \sup_{p \in \hat{P}} \frac{(B^* p \,,\, w_r B^* q)^2}{(w_l A_\mu w_r B^* p \,,\, B^* p)} \quad \text{for all } q \in \hat{Q}, \tag{6.11}$$

where $\hat{P} \coloneqq C^\infty(\Omega)$ and $\hat{Q} \coloneqq L^2(\Omega)/\mathbb{R} \cap H^1(\Omega)$, and

$$(Sq \,,\, q) = \sup_{\boldsymbol{v} \in V} \frac{(\boldsymbol{v} \,,\, w_r B^* q)^2}{(w_l A_\mu w_r \boldsymbol{v} \,,\, \boldsymbol{v})} \quad \text{for all } q \in Q. \tag{6.12}$$

Proof. For $q \in \hat{Q}$, we use integration by parts on the left hand side of (6.11) to obtain

$$\begin{aligned}\left(\tilde{S}_{\text{w-BFBT}}\, q \,,\, q\right) &= \int_\Omega \left(w_r B^* (B w_l A_\mu w_r B^*)^{-1} K_{w_l} q\right) (B^* q) \,\mathrm{d}\boldsymbol{x} + b_1(q) \\ &= \int_\Omega \left((B w_l A_\mu w_r B^*)^{-1} K_{w_l} q\right) (K_{w_r} q) \,\mathrm{d}\boldsymbol{x} + b_1(q) + b_2(q)\end{aligned}$$

with boundary terms

$$b_1(q) \coloneqq -\int_{\partial\Omega} \left(\boldsymbol{n} \cdot w_r B^* (B w_l A_\mu w_r B^*)^{-1} K_{w_l} q\right) q \,\mathrm{d}\boldsymbol{x},$$
$$b_2(q) \coloneqq \int_{\partial\Omega} \left((B w_l A_\mu w_r B^*)^{-1} K_{w_l} q\right) (\boldsymbol{n} \cdot w_r B^* q) \,\mathrm{d}\boldsymbol{x}.$$

Using that $\hat{P} \subset H^2(\Omega)$ is dense, application of Lemma 6.3.2 and further integration by parts yields

$$\begin{aligned}\left(\tilde{S}_{\text{w-BFBT}}\, q \,,\, q\right) &= \sup_{p \in \hat{P}} \frac{(p \,,\, K_{w_r} q)^2}{(B w_l A_\mu w_r B^* p \,,\, p)} + b_1(q) + b_2(q) \\ &= \sup_{p \in \hat{P}} \frac{((B^* p \,,\, w_r B^* q) + b_3(p,q))^2}{(w_l A_\mu w_r B^* p \,,\, B^* p) + b_4(p)} + b_1(q) + b_2(q)\end{aligned}$$

with boundary terms

$$b_3(p,q) \coloneqq -\int_{\partial\Omega} p \,(\boldsymbol{n} \cdot w_r B^* q) \,\mathrm{d}\boldsymbol{x},$$
$$b_4(p) \coloneqq -\int_{\partial\Omega} (\boldsymbol{n} \cdot w_l A_\mu w_r B^* p)\, p \,\mathrm{d}\boldsymbol{x}.$$

Because the operator $K_{w_r}^*$ from (6.9) is augmented with homogeneous Neumann boundary conditions, $\boldsymbol{n} \cdot B^* p = 0$, the boundary terms $b_1(q)$, $b_2(q)$, and $b_3(p,q)$ vanish. In addition, $p \in \hat{P}$ is sufficiently regular for the term $b_4(p)$ to be well-defined and it equals to zero because the velocity $\boldsymbol{u}$ satisfies $\boldsymbol{n} \cdot \boldsymbol{u} = 0$ on $\partial\Omega$. Hence, (6.11) follows.

To show (6.12), let $q \in Q$, then for the exact Schur complement we apply integration by parts with a vanishing boundary term

$$(Sq \,,\, q) = \left(B A_\mu^{-1} B^* q \,,\, q\right) = \left(A_\mu^{-1} B^* q \,,\, B^* q\right) = \left(w_r^{-1} A_\mu^{-1} w_l^{-1} w_l B^* q \,,\, w_r B^* q\right)$$

and (6.12) follows from Lemma 6.3.2. □

A direct consequence of Lemma 6.3.3 is the following lower bound.

Corollary 6.3.4 (Lower bound, $\tilde{S}_{\text{w-BFBT}} \lesssim S$)**.** *The exact Schur complement is bounded by the w-BFBT approximation from below, i.e.,*

$$\left(\tilde{S}_{\text{w-BFBT}}\, q \,,\, q\right) \leq (Sq \,,\, q) \quad \textit{for all } q \in \hat{Q}, \tag{6.13}$$

where $\hat{Q} \coloneqq L^2(\Omega)/\mathbb{R} \cap H^1(\Omega)$.

Proof. Let $\hat{P} = C^\infty(\Omega)$ and $q \in \hat{Q}$. Since B^* maps $\hat{P}$ into V, we combine (6.11) and (6.12) to get

$$\left(\tilde{S}_{\text{w-BFBT}}\, q \,,\, q\right) = \sup_{p \in \hat{P}} \frac{(B^* p \,,\, w_r B^* q)^2}{(w_l A_\mu w_r B^* p \,,\, B^* p)} \leq \sup_{\boldsymbol{v} \in V} \frac{(\boldsymbol{v} \,,\, w_r B^* q)^2}{(w_l A_\mu w_r \boldsymbol{v} \,,\, \boldsymbol{v})} = (Sq \,,\, q)\,,$$

and obtain the result (6.13). □

We begin the derivation of an upper bound for the case of constant viscosity $\mu \equiv 1$. Note that $\tilde{S}_{\text{w-BFBT}}$ is scaling invariant with respect to constants multiplied to the w-BFBT weight functions w_l, w_r. Hence, it always assumes the correct scaling of S independent of the viscosity constant. The result for constant viscosity presented below in Lemma 6.3.5 is generalized to variable viscosity in Lemma 6.3.6. While Lemma 6.3.5 is a special case of Lemma 6.3.6, we first prove the result for constant viscosity as the arguments are less technical and easier to follow. In the proof of the result for variable viscosity, we build on some of the arguments from the constant viscosity case and thus avoid unnecessary duplication.

Lemma 6.3.5 (Upper bound, $S \lesssim \tilde{S}_{\text{w-BFBT}}$, for constant μ)**.** *Assume a constant viscosity $\mu \equiv 1$ and constant w-BFBT weight functions $w_l \equiv 1 \equiv w_r$, and, as before in Lemma 6.3.3, let $\hat{Q} = L^2(\Omega)/\mathbb{R} \cap H^1(\Omega)$. Then the exact Schur complement is bounded by the w-BFBT approximation from above by*

$$(Sq\,,\,q) \leq (C_P^2 + 1)\, C_K^2 \left(\tilde{S}_{\text{w-BFBT}}\, q\,,\, q\right) \quad \text{for all } q \in \hat{Q} \tag{6.14}$$

with constants $C_P, C_K > 0$ stemming from Poincaré–Friedrichs' and Korn's inequalities, respectively.

Proof. Let $\hat{P} = C^\infty(\Omega)$ and $q \in \hat{Q}$, then due to (6.11) we can write

$$\left(\tilde{S}_{\text{w-BFBT}}\, q\,,\, q\right) = \sup_{p \in \hat{P}} \frac{(B^*p\,,\,B^*q)^2}{\|B^*p\|^2_{H^1(\Omega)^d}} \frac{\|B^*p\|^2_{H^1(\Omega)^d}}{(A_1 B^*p\,,\,B^*p)}. \tag{6.15}$$

To estimate the second factor on the right-hand side of (6.15), note that

$$(A_1 B^*p\,,\,B^*p) = 2\,(\nabla_{\!\text{s}} B^*p\,,\,\nabla_{\!\text{s}} B^*p) = 2\,(\nabla B^*p\,,\,\nabla B^*p) = 2\,\|\nabla B^*p\|^2_{L^2(\Omega)^{d\times d}}\,,$$

where we used that $\boldsymbol{n} \cdot B^*p = 0$ on the boundary and that ∇B^*p is symmetric. Thus,

$$(A_1 B^*p\,,\,B^*p) \leq 2\,\|B^*p\|^2_{H^1(\Omega)^d}\,. \tag{6.16}$$

For the first factor on the right-hand side of (6.15), observe that for any $\boldsymbol{v} \in V$ there exists a sequence $\{p_i\}_i \subset \hat{P} \subset H^2(\Omega)$ such that $K_1^* p_i = BB^* p_i \to B\boldsymbol{v}$, since K_1^* is invertible, where convergence is with respect to the L^2-norm. Thus,

$$\sup_{p \in \hat{P}} \frac{(B^*p\,,\,B^*q)}{\|B^*p\|_{H^1(\Omega)^d}} = \sup_{\boldsymbol{v} \in V} \frac{(\boldsymbol{v}\,,\,B^*q)}{\|\boldsymbol{v}\|_{H^1(\Omega)^d}} = \|B^*q\|_{H^{-1}(\Omega)^d}\,. \tag{6.17}$$

Combining (6.16) and (6.17) provides the following estimate for the w-BFBT Schur complement approximation (6.15):

$$\left(\tilde{S}_{\text{w-BFBT}}\, q\,,\, q\right) \geq \frac{1}{2}\,\|B^*q\|^2_{H^{-1}(\Omega)^d}\,. \tag{6.18}$$

The exact Schur complement, on the other hand, in the form (6.12) from Lemma 6.3.3, can be bounded by

$$(Sq\,,\,q) = \sup_{\boldsymbol{v} \in V} \frac{(\boldsymbol{v}\,,\,B^*q)^2}{(A_1\boldsymbol{v}\,,\,\boldsymbol{v})} \leq \sup_{\boldsymbol{v} \in V} \frac{\|\boldsymbol{v}\|^2_{H^1(\Omega)^d}\,\|B^*q\|^2_{H^{-1}(\Omega)^d}}{2\,\|\nabla_{\!\text{s}}\boldsymbol{v}\|^2_{L^2(\Omega)^{d\times d}}}. \tag{6.19}$$

With Poincaré–Friedrichs' inequality,

$$\|\boldsymbol{v}\|_{L^2(\Omega)^d} \leq C_P \, \|\nabla \boldsymbol{v}\|_{L^2(\Omega)^{d\times d}} \,,$$

where the constant $C_P > 0$ depends on the domain Ω, and Korn's inequality,

$$\|\nabla \boldsymbol{v}\|_{L^2(\Omega)^{d\times d}} \leq C_K \, \|\nabla_{\mathrm{s}} \boldsymbol{v}\|_{L^2(\Omega)^{d\times d}} \,,$$

with constant $C_K > 0$, we obtain

$$\frac{1}{(C_P^2+1)} \, \|\boldsymbol{v}\|^2_{H^1(\Omega)^d} \leq \|\nabla \boldsymbol{v}\|^2_{L^2(\Omega)^{d\times d}} \leq C_K^2 \, \|\nabla_{\mathrm{s}} \boldsymbol{v}\|^2_{L^2(\Omega)^{d\times d}} \,,$$

and substituting this into (6.19) gives

$$(Sq \,,\, q) \leq \frac{(C_P^2+1)\, C_K^2}{2} \, \|B^* q\|^2_{H^{-1}(\Omega)^d} \,. \tag{6.20}$$

Together with (6.18), this yields the desired result (6.14). □

We complete the presentation of spectral equivalence by deriving an upper bound for problems with variable viscosities.

Lemma 6.3.6 (Upper bound, $S \lesssim \tilde{S}_{\text{w-BFBT}}$, for variable μ)**.** *As before in Lemma 6.3.3, let* $\hat{Q} = L^2(\Omega)/\mathbb{R} \cap H^1(\Omega)$*. If the left and right w-BFBT weight functions are equal to*

$$w_l(\boldsymbol{x}) = \mu(\boldsymbol{x})^{-\frac{1}{2}} = w_r(\boldsymbol{x}) \quad \textit{for a.a. } \boldsymbol{x} \in \Omega, \tag{6.21}$$

then the exact Schur complement is bounded by the w-BFBT approximation from above by

$$(Sq \,,\, q) \leq \Big(1 + \frac{1}{4} \, \|\nabla \mu\|^2_{L^\infty(\Omega)^d}\Big)\Big(C^2_{P,\mu} + 1\Big) C^2_{K,\mu} \, \Big(\tilde{S}_{\text{w-BFBT}} \, q \,,\, q\Big) \quad \textit{for all } q \in \hat{Q} \tag{6.22}$$

with constants $C_{P,\mu}, C_{K,\mu} > 0$ *stemming from weighted Poincaré–Friedrichs' and Korn's inequalities, respectively, where* μ *assumes the role of the weight function.*

Proof. Let the weight functions be equal, $w_l \equiv w \equiv w_r$, but (for now) otherwise arbitrary subject to the condition $0 < w_{\min} \leq w(\boldsymbol{x})$ for a.a. $\boldsymbol{x} \in \Omega$. At the end of this proof we will argue the special role of the choice (6.21) for the weight functions. In addition, let $\hat{P} = C^\infty(\Omega)$ and $q \in \hat{Q}$, then due to (6.11) we can write

$$\Big(\tilde{S}_{\text{w-BFBT}} \, q \,,\, q\Big) = \sup_{p \in \hat{P}} \frac{(B^* p \,,\, w B^* q)^2}{\|B^* p\|^2_{H^1(\Omega)^d}} \, \frac{\|B^* p\|^2_{H^1(\Omega)^d}}{(w A_\mu w B^* p \,,\, B^* p)}. \tag{6.23}$$

We begin by estimating the second factor on the right-hand side of (6.23). For an arbitrary $\boldsymbol{v} \in H^1(\Omega)^d$, observe that $\nabla_{\mathrm{s}}(w\boldsymbol{v}) = w \nabla_{\mathrm{s}} \boldsymbol{v} + \nabla w \otimes \boldsymbol{v}$, where "$\otimes$" denotes the outer product of two vectors in $\mathbb{R}^d$, and thus

$$(w A_\mu w \boldsymbol{v} \,,\, \boldsymbol{v}) = 2 \, (\mu \nabla_{\mathrm{s}}(w\boldsymbol{v}) \,,\, \nabla_{\mathrm{s}}(w\boldsymbol{v})) = 2 \, \|\sqrt{\mu} w \nabla_{\mathrm{s}} \boldsymbol{v} + \sqrt{\mu} \nabla w \otimes \boldsymbol{v}\|^2_{L^2(\Omega)^{d\times d}} \,.$$

Applying the triangle inequality and then Hölder's inequality to the resulting terms,

$$\|\sqrt{\mu} w \nabla_{\mathrm{s}} \boldsymbol{v}\|_{L^2(\Omega)^{d\times d}}^2 \leq \|\sqrt{\mu} w\|_{L^\infty(\Omega)}^2 \, \|\nabla_{\mathrm{s}} \boldsymbol{v}\|_{L^2(\Omega)^{d\times d}}^2 \,,$$
$$\|\sqrt{\mu} \nabla w \otimes \boldsymbol{v}\|_{L^2(\Omega)^{d\times d}}^2 \leq \|\sqrt{\mu} \nabla w\|_{L^\infty(\Omega)^d}^2 \, \|\boldsymbol{v}\|_{L^2(\Omega)^d}^2 \,,$$

and thus we obtain the estimate

$$(w A_\mu w \boldsymbol{v} \,,\, \boldsymbol{v}) \leq 2 C_{A,\mu,w} \, \|\boldsymbol{v}\|_{H^1(\Omega)^d}^2 \,,$$

where

$$C_{A,\mu,w} \coloneqq \|\sqrt{\mu} w\|_{L^\infty(\Omega)}^2 + \|\sqrt{\mu} \nabla w\|_{L^\infty(\Omega)^d}^2 \,. \tag{6.24}$$

Similarly to (6.17) and (6.18), we obtain the following estimate for the w-BFBT Schur complement approximation (6.23):

$$\left(\tilde{S}_{\text{w-BFBT}} \, q \,,\, q\right) \geq \frac{1}{2 C_{A,\mu,w}} \, \|w B^* q\|_{H^{-1}(\Omega)^d}^2 \,. \tag{6.25}$$

Proceeding with the exact Schur complement, we obtain from (6.12) in Lemma 6.3.3 that

$$(Sq \,,\, q) = \sup_{\boldsymbol{v}\in V} \frac{\left(w^{-1}\boldsymbol{v} \,,\, w B^* q\right)^2}{(A_\mu \boldsymbol{v} \,,\, \boldsymbol{v})} \leq \sup_{\boldsymbol{v}\in V} \frac{\left\|w^{-1}\boldsymbol{v}\right\|_{H^1(\Omega)^d}^2 \, \|w B^* q\|_{H^{-1}(\Omega)^d}^2}{2 \left\|\sqrt{\mu} \nabla_{\mathrm{s}} \boldsymbol{v}\right\|_{L^2(\Omega)^{d\times d}}^2}. \tag{6.26}$$

We require a weighted Poincaré–Friedrichs' inequality (see Remark 6.3.7 for details),

$$\|\boldsymbol{v}\|_{L^2_{w^{-2}}(\Omega)^d} \leq C_{P,w^{-2}} \, \|\nabla \boldsymbol{v}\|_{L^2_{w^{-2}}(\Omega)^{d\times d}} \,, \tag{6.27}$$

and also a weighted Korn's inequality (see Remark 6.3.7 for more information),

$$\|\nabla \boldsymbol{v}\|_{L^2_\mu(\Omega)^{d\times d}} \leq C_{K,\mu} \, \|\nabla_{\mathrm{s}} \boldsymbol{v}\|_{L^2_\mu(\Omega)^{d\times d}} \,. \tag{6.28}$$

With (6.27) and (6.28), we are able to bound (6.26) from above:

$$(Sq \,,\, q) \leq \frac{\left(C_{P,w^{-2}}^2 + 1\right) C_{K,\mu}^2}{2} \left(\sup_{\boldsymbol{v}\in V} \frac{\|\nabla \boldsymbol{v}\|_{L^2_{w^{-2}}(\Omega)^{d\times d}}}{\|\nabla \boldsymbol{v}\|_{(L^2_\mu(\Omega))^{d\times d}}} \right) \|w B^* q\|_{H^{-1}(\Omega)^d}^2 \,. \tag{6.29}$$

The supremum term in (6.29) and the constant $C_{A,\mu,w}$ in (6.24) motivate the choice for the weight w to be

$$w \coloneqq \mu^{-\frac{1}{2}}.$$

Then the supremum in (6.29) vanishes and (6.24) simplifies to

$$C_{A,\mu,w} = 1 + \frac{1}{4} \, \|\nabla \mu\|_{L^\infty(\Omega)^d}^2 \,.$$

Substituting this into (6.25) together with inequality (6.29) yields the desired result (6.22). □

Remark 6.3.7. In the proof of Lemma 6.3.6 we utilized a weighted Poincaré–Friedrichs' inequality, for which the optimal constant is

$$C_{P,\mu} = \sup_{\boldsymbol{v} \in V} \frac{\|\boldsymbol{v}\|_{L^2_\mu(\Omega)^d}}{\|\nabla \boldsymbol{v}\|_{L^2_\mu(\Omega)^{d\times d}}},$$

where the viscosity takes the role of the weight function. While weighted Poincaré and Friedrichs' inequalities have been investigated in the literature numerous times, usually they are proven by contradiction and scaling arguments, which does not provide information about the constants. If explicit constants are found, they depend, in general, on the weight such that the resulting estimates are too pessimistic, e.g., $C_{P,\mu} = \mathcal{O}(\mathrm{DR}(\mu))$. Knowledge of constants that are robust with respect to weight functions is limited. In the context of a posteriori error estimates for finite elements, weight-independent constants could be found for convex domains and weights that are a positive power of a non-negative concave function [34]. These results were refined for star-shaped domains under certain assumptions for the weights [118]. For another class of weights, namely quasi-monotone piecewise constant weight functions, robust constants were derived in [90].

In addition to weighted Poincaré–Friedrichs', we utilized a weighted Korn's inequality in the proof of Lemma 6.3.6. The optimal constant for this inequality is

$$C_{K,\mu} = \sup_{\boldsymbol{v} \in V} \frac{\|\nabla \boldsymbol{v}\|_{L^2_\mu(\Omega)^{d\times d}}}{\|\nabla_{\mathrm{s}} \boldsymbol{v}\|_{L^2_\mu(\Omega)^{d\times d}}}.$$

As for $C_{P,\mu}$, straightforward estimation results in an overly pessimistic weight-dependent constant, namely $C_{K,\mu} = \mathcal{O}(\mathrm{DR}(\mu))$, [73]. Other work utilizing weighted Korn's inequalities usually aims to derive inequalities for special domain shapes, e.g., [1].

In summary, robust constants for weighted Poincaré–Friedrichs' and Korn's inequalities for general weight functions are difficult to obtain and limitations exist in the form of assumptions on the weights. Further research on this topic could improve the constants for the spectral equivalence of w-BFBT but is beyond the scope of this work.

6.4 Robustness of w-BFBT

In this section, we analyze the robustness properties of the widely used Schur complement approximation $\mathbf{M}_p(1/\mu)$ and the new w-BFBT via numerical experiments. Furthermore, we calculate the spectra for both approaches and thus support the discussion in Section 6.3 about theoretical eigenvalue bounds with numerical results. The comparison of $\mathbf{M}_p(1/\mu)$ and w-BFBT is of particular importance, because of the widespread use of the inverse viscosity-weighted mass matrix. It is therefore of interest to determine when convergence with $\mathbf{M}_p(1/\mu)$ deteriorates and using w-BFBT becomes beneficial. A comparison with diag($\mathbf{A}$)-BFBT was not performed because Section 6.2 already showed that diag($\mathbf{A}$)-BFBT performs similarly or worse than w-BFBT, hence there are no advantages in using diag($\mathbf{A}$)-BFBT over w-BFBT.

Table 6.1: Robustness classification for Schur complement approximations (a) $\mathbf{M}_p(1/\mu)$ and (b) w-BFBT in terms of number of GMRES iterations (10^{-6} residual reduction, GMRES restart every 100 iterations). Number of randomly placed sinkers (#sinkers) is increased across rows, while dynamic ratio (DR(μ)) is increased across columns. Discretization is fixed at $k = 2$, $\ell = 7$.

(a) $\mathbf{M}_p(1/\mu)$

#sinkers \ DR(μ)	10^4	10^6	10^8	10^{10}
S1-rand	29	31	31	29
S4-rand	53	63	71	80
S8-rand	64	79	93	165
S12-rand	70	86	99	180
S16-rand	85	167	231	891
S20-rand	84	167	380	724
S24-rand	117	286	3279	5983
S28-rand	108	499	2472	>10000

(b) w-BFBT

#sinkers \ DR(μ)	10^4	10^6	10^8	10^{10}
S1-rand	29	31	31	29
S4-rand	53	63	71	80
S8-rand	64	79	93	165
S12-rand	70	86	99	180
S16-rand	85	167	231	891
S20-rand	84	167	380	724
S24-rand	117	286	3279	5983
S28-rand	108	499	2472	>10000

For the numerical experiments in this section, we return to the definitions and setup from Section 6.2. To apply the inverse of $\mathbf{M}_p(1/\mu)$, we diagonalize the mass matrix of the discontinuous, modal pressure space by forming its lumped version (6.2). Moreover, to apply the approximate inverse of the viscous block in (6.1) we use the same multigrid method for each of the two Schur approximations; this multigrid method is also used for the inverse operators of w-BFBT in (6.6). The details of the multigrid method are provided in Section 7.2. To compare the robustness, we vary two problem parameters: (i) the number of randomly placed sinkers n and (ii) the dynamic ratio DR(μ). The parameter n influences the geometric complexity of the viscosity μ while DR(μ) controls the magnitude of viscosity gradients.

Tables 6.1a and 6.1b present the number of GMRES iterations for a 10^{-6} residual reduction in the Euclidean norm. Observe that for the S1-rand problem, the iteration count is essentially the same for both $\mathbf{M}_p(1/\mu)$ and w-BFBT, and that it stays stable across all dynamic ratios DR(μ) $= 10^4, \ldots, 10^{10}$. Hence for this simple problem, w-BFBT has no advantages and its additional computational cost makes it less efficient than $\mathbf{M}_p(1/\mu)$. However, the limitations of the $\mathbf{M}_p(1/\mu)$ approach become apparent by increasing the number of randomly positioned sinkers. Two observations for $\mathbf{M}_p(1/\mu)$ can be made from Table 6.1a. First, the number of GMRES iterations rises with increasing number of sinkers (factor $\sim$80 increase for $n = 1, \ldots, 28$, DR(μ) $= 10^8$). Second, in a multi-sinker setup the dependence on DR(μ) becomes more severe (factor $\sim$50 increase for $n = 24$, DR(μ) $= 10^4, \ldots, 10^{10}$). This demonstrates that $\mathbf{M}_p(1/\mu)$ is a poor approximation of the Schur complement for certain classes of problems with highly heterogeneous viscosities.

The advantages in robustness of the w-BFBT preconditioner are demonstrated in Table 6.1b. Compared to $\mathbf{M}_p(1/\mu)$, the number of GMRES iterations is stable and the increase over the whole range of problem parameters is just a factor of 2. Only 60 iterations are needed for the most extreme problem, namely S28-rand, DR(μ) $= 10^{10}$, for which convergence with $\mathbf{M}_p(1/\mu)$ essentially stagnated.

More insight concerning the different convergence behaviors can be gained from the eigenvalues in Figure 6.3. The plots in that figure are for two-dimensional multi-sinker problems, which are

analogous to the three-dimensional benchmark problems from Section 6.2. We discretize the problems on triangular meshes utilizing the FEniCS library [81]. We choose $\mathbb{P}_2^{\text{bubble}} \times \mathbb{P}_1^{\text{disc}}$ finite elements [35,39] because they represent a close analog to the $\mathbb{Q}_2 \times \mathbb{P}_1^{\text{disc}}$ elements, which are employed on (three-dimensional) hexahedral meshes. Each plot shows the eigenvalues of the exact Schur complement and the eigenvalues of the preconditioned Schur complement for the $\mathbf{M}_p(1/\mu)$ and w-BFBT approximations, where all inverse matrices, e.g., the viscous block matrix $\mathbf{A}$ and the pressure Poisson matrices within w-BFBT, are inverted with a direct solver. Effective preconditioners exhibit a strong clustering of eigenvalues, whereas the convergence of Krylov methods deteriorates if the eigenvalues are spread out. We recognize different characteristics in the spectra associated with $\mathbf{M}_p(1/\mu)$ and w-BFBT preconditioning. For $\mathbf{M}_p(1/\mu)$, the dominant eigenvalues are clustered around one while smaller eigenvalues, i.e., eigenvalues $\ll 1$, are spread out. The behavior for w-BFBT is the opposite: the dominant eigenvalues are spread out and the smaller eigenvalues are tightly clustered around one. Now, as the problem difficulty is increased by introducing more viscosity anomalies in the domain, the spreading of smaller eigenvalues associated with $\mathbf{M}_p(1/\mu)$ becomes more severe (compare in Figure 6.3 the *top row* of plots and the *bottom row* of plots). We postulate that this is the property that is responsible for the deteriorating convergence with $\mathbf{M}_p(1/\mu)$ that was observed in Table 6.1a. With w-BFBT on the other hand, the spectrum remains largely unaffected by increased sinker counts. The clustering of smaller eigenvalues around one remains stable, which is likely the reason for the robustness of w-BFBT. The lower bound on the eigenvalues that we observe here numerically supports the theoretical estimates on spectral equivalence in Section 6.3. Therefore we find the lower bound to be sharp and, moreover, to be essential for the robustness of the w-BFBT preconditioner.

Remark 6.4.1. In addition to the $\mathbb{P}_2^{\text{bubble}} \times \mathbb{P}_1^{\text{disc}}$ discretization used for the results in Figure 6.3, we also calculated the spectra using $\mathbb{P}_2 \times \mathbb{P}_1$ Taylor-Hood finite elements. We obtained very similar results for this discretization, which uses continuous elements to approximate the pressure. Therefore, both the efficacy of w-BFBT as a preconditioner and the issues with $\mathbf{M}_p(1/\mu)$ seem to be largely unaffected by the specific type of discretization, at least for the two cases that we tested.

Remark 6.4.2. The convergence of the Stokes solver with the $\mathbf{M}_p(1/\mu)$ preconditioner can be improved by approximating the heterogeneous viscosity $\mu(\boldsymbol{x})$ with elementwise constants, computed by averaging μ over each element [11]. The benefit of faster convergence comes at the cost of slower asymptotic convergence of the discrete finite element solution and an altered constitutive relationship, which might be a less accurate representation of the physics; this is, however, problem-dependent. Moreover, for a nonlinear (e.g., power-law) rheology, elementwise averaging of μ can introduce non-physical, artificial disturbances in the effective viscosity during Newton or Picard-type nonlinear solves. We observed such a behavior in mantle convection simulations, which are governed by a nonlinear power-law rheology. Here, viscosity averaging led to non-physical checkerboard-like patterns upon convergence of the nonlinear Newton solver.

Remark 6.4.3. In practice, the convergence of w-BFBT can be improved for coarse meshes, where the viscosity variations over elements are large. This is achieved by alternative choices for the diagonal

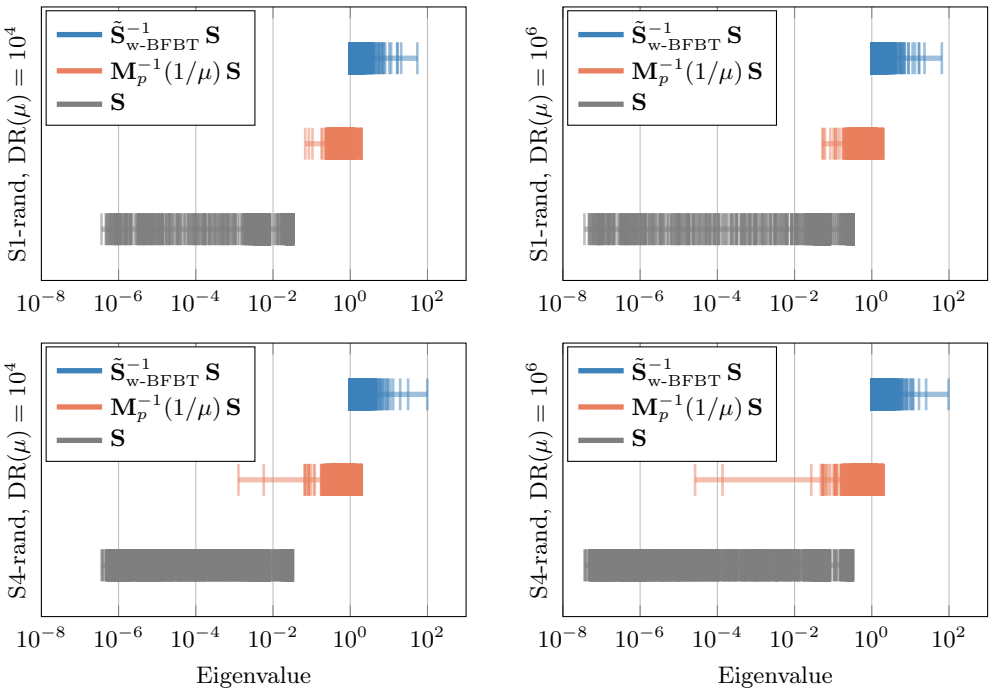


Figure 6.3: Spectra of the Schur complement *(gray)*, $\mathbf{M}_p(1/\mu)$-preconditioned Schur complement *(red)*, and w-BFBT-preconditioned Schur complement *(blue)*; zero eigenvalues corresponding to the null space of the Schur complement matrix are omitted. Results for viscosities with one sinker (S1-rand) are shown in *top row*, and with four sinkers (S4-rand) in the *bottom row* of plots; $\mathrm{DR}(\mu) = 10^4$ in the *left column* and $\mathrm{DR}(\mu) = 10^6$ in the *right column*. The two-dimensional Stokes equations are discretized with $\mathbb{P}_2^{\text{bubble}} \times \mathbb{P}_1^{\text{disc}}$ finite elements on a uniform triangular mesh consisting of 512 triangles using the FEniCS library. As the problem difficulty increases from one to four sinkers, the spreading of small eigenvalues for $\mathbf{M}_p(1/\mu)$ becomes more severe, which is disadvantageous for solver convergence. For w-BFBT, the spectrum remains largely unaffected by increased sinker counts, which contributes to convergence that is robust with respect to viscosity variations.

weighting matrices $\mathbf{C}_{w_l}$ and $\mathbf{D}_{w_r}$ from (6.6) with the weights

$$w_l(\boldsymbol{x}) = \left(\mu^2(\boldsymbol{x}) + |\nabla\mu(\boldsymbol{x})|^2\right)^{\frac{1}{4}} = w_r(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in \Omega,$$

where $|\cdot|$ denotes the Euclidean norm in $\mathbb{R}^d$. These viscosity gradient-based w-BFBT weights have the advantage of performing at least as well as the pure viscosity-based weights, proposed in Section 6.1, but they exhibit superior robustness on coarser meshes. They are, however, challenging to analyze theoretically.

6.5 Modifications for Dirichlet boundary conditions

In Section 6.2, deteriorating approximation properties of w-BFBT for increasing discretization order and mesh refinement level could be observed. The numerical experiments in Figure 6.2, *right*

did show slightly slower convergence when k and ℓ were increased. This can stem from w-BFBT representing a poor approximation to the exact Schur complement at the boundary $\partial\Omega$ in the presence of Dirichlet boundary conditions for the velocity. This section investigates modifications to w-BFBT near a Dirichlet boundary and aims at obtaining mesh independence and only a mild dependence on discretization order in terms of Stokes solver convergence.

Consider the commutator that leads to the w-BFBT formulation in an infinite-dimensional form: $AB^* - B^*X \approx 0$, where A represents the viscous stress operator, B^* the gradient operator and X the sought commuting operator. In case of an unbounded domain $\Omega = \mathbb{R}^d$ and constant viscosity $\mu \equiv 1$, this commutator is exactly satisfied since $(\nabla \cdot \nabla)\nabla - \nabla(\nabla \cdot \nabla) = 0$. For Dirichlet boundary conditions, the commutator does not, in general, vanish at the boundary. Therefore a possible source for deteriorating Schur complement approximation properties of w-BFBT is a commutator mismatch for mesh elements that are touching the boundary $\partial\Omega$. A similar observation was also made in [46, 47]. A possible remedy is to modify the norm in the least-squares minimization problem (6.4), which is represented by the matrix $\mathbf{C}^{-1}$, such that a damping factor is applied to the matrix entries near the boundary. By damping the influence of the boundary in the minimization objective, more emphasis is given to the domain interior, and the w-BFBT approximation is improved.

Damping near Dirichlet boundaries can be incorporated by modifying the matrices $\mathbf{C}_{w_l}^{-1}$ or $\mathbf{D}_{w_r}^{-1}$ of the w-BFBT inverse Schur complement approximation (6.6). A similar idea for BFBT in a Navier–Stokes setting is presented in [47], where a damping to the weighting matrix $\mathbf{D}^{-1}$ in (6.5) is introduced to achieve mesh independence ($\mathbf{C}^{-1}$ is not changed). There, damping affects the normal components of the velocity space inside mesh elements touching $\partial\Omega$ and simply a constant damping factor of $1/10$ is set regardless of mesh refinement ℓ. Also, only the discretization order $k = 2$ was considered (in addition to $\mathbb{Q}_2 \times \mathbb{Q}_1$ and MAC discretizations).

Now, we attempt to enhance our understanding of how modifications at a Dirichlet boundary $\partial\Omega$ influence convergence and therefore the efficacy of w-BFBT as a Schur complement approximation. Let $\Omega_D \coloneqq \bigcup_{e \in D} \Omega_e$, $D \coloneqq \{e \mid \overline{\Omega_e} \cap \partial\Omega \neq \emptyset\}$ be the set of all mesh elements Ω_e touching the Dirichlet boundary. Given values $a_l, a_r \geq 1$, extend the previous definition of the weights $w_l = w_r = \sqrt{\mu}$ (see Section 6.1) to a version with boundary modification:

$$w_l(\boldsymbol{x}) \coloneqq \begin{cases} a_l\sqrt{\mu(\boldsymbol{x})} & \boldsymbol{x} \in \Omega_D, \\ \sqrt{\mu(\boldsymbol{x})} & \boldsymbol{x} \notin \Omega_D, \end{cases} \quad \text{and} \quad w_r(\boldsymbol{x}) \coloneqq \begin{cases} a_r\sqrt{\mu(\boldsymbol{x})} & \boldsymbol{x} \in \Omega_D, \\ \sqrt{\mu(\boldsymbol{x})} & \boldsymbol{x} \notin \Omega_D. \end{cases} \tag{6.30}$$

We obtain matrices $\mathbf{C}_{w_l} = \tilde{\mathbf{M}}_{\boldsymbol{u}}(w_l)$ and $\mathbf{D}_{w_r} = \tilde{\mathbf{M}}_{\boldsymbol{u}}(w_r)$ in (6.6) that may differ at boundary elements in Ω_D due to possibly different values for a_l and a_r. Note that amplifying the weight functions w_l, w_r at the boundary is similar to damping at the boundary after taking the inverses $\mathbf{C}_{w_l}^{-1}$, $\mathbf{D}_{w_r}^{-1}$.

The Stokes solver convergence under the influence of boundary amplifications a_l, a_r is summarized in Table 6.2. The table shows that the boundary amplification is most effective when performed non-symmetrically, i.e., either $a_l > 1$ or $a_r > 1$ but not both. Further, we deduce that with higher mesh refinement level ℓ, the boundary amplification should increase roughly proportional to 2^ℓ (or

Table 6.2: Influence of boundary modification factors a_l, a_r on the Stokes solver convergence with w-BFBT for discretizations: $k = 2$, $\ell = 5, \ldots, 7$ (see (a), (c), (e)) and $k = 2, \ldots, 5$, $\ell = 5$ (see (a), (b), (d), (f)). Reported are the number of GMRES iterations for 10^{-6} residual reduction for the problem S16-rand, $\mathrm{DR}(\mu) = 10^6$. *Colors* highlight solves within ∼5% of iterations above the lowest iteration count. Increase of mesh refinement level ℓ or discretization order k demands larger boundary amplification a_r or a_l to maintain fast convergence.

(a) $k = 2$, $\ell = 5$

$a_l \setminus a_r$	1	2	4	8	16	32
1	33	33	34	34	34	35
2	33	33	34	34	34	34
4	33	34	34	36	38	39
8	34	34	36	39	43	44
16	34	34	38	43	46	49
32	34	34	39	44	49	53

(b) $k = 3$, $\ell = 5$

$a_l \setminus a_r$	1	2	4	8	16	32
1	41	38	37	37	37	37
2	38	37	38	38	39	39
4	37	38	40	42	44	46
8	36	38	42	47	50	51
16	37	39	44	50	53	56
32	37	39	45	51	56	59

(c) $k = 2$, $\ell = 6$

$a_l \setminus a_r$	1	2	4	8	16	32
1	37	34	33	34	34	34
2	34	34	34	34	34	34
4	33	33	34	35	36	37
8	34	34	35	38	39	39
16	34	34	36	39	40	41
32	34	34	37	39	41	42

(d) $k = 4$, $\ell = 5$

$a_l \setminus a_r$	1	2	4	8	16	32
1	44	39	36	36	36	36
2	39	39	39	40	41	41
4	36	39	43	47	49	51
8	36	40	47	52	56	58
16	36	41	49	56	60	63
32	36	41	50	58	63	66

(e) $k = 2$, $\ell = 7$

$a_l \setminus a_r$	1	2	4	8	16	32
1	45	37	34	34	34	34
2	37	36	35	36	36	36
4	34	36	38	39	40	41
8	34	36	39	42	44	44
16	34	36	40	44	45	46
32	34	36	41	44	46	47

(f) $k = 5$, $\ell = 5$

$a_l \setminus a_r$	1	2	4	8	16	32
1	63	53	46	43	43	44
2	53	51	51	51	52	53
4	47	51	55	59	62	64
8	44	51	59	65	69	72
16	43	52	62	69	75	78
32	44	53	64	72	78	82

proportional to the reciprocal element size, here $h^{-1} = 2^\ell$). Similar observations can be made for the discretization order k, i.e., amplification needs to increase for larger k to avoid higher iteration counts. These implications were made based on extensive numerical experiments for which Table 6.2 serves as a representative summary.

Remark 6.5.1. The theoretical derivations of spectral equivalence from Section 6.3 and the necessity for damping at Dirichlet boundaries appear inconsistent. However, spectral equivalence was shown in infinite dimensions whereas boundary damping is applied to the discretized problem. Therefore, we believe that the necessity for damping is introduced through the discretization. It still remains an open question what might be causing the slowdown in convergence that is avoided by damping.

7

Multigrid Preconditioning with HMG

For approximative inverses of the elliptic differential operators that arise within our Schur complement-based Stokes preconditioner, we develop multigrid methods that exhibit extreme parallel scalability and retain nearly optimal algorithmic scalability, which is demonstrated in Chapter 8 and also documented in [96, 97, 113]. Generally, multigrid methods are based on the fundamental idea that effective and computationally efficient solvers can be constructed for a relatively narrow spectrum of the solution space. A multigrid method then combines these sub-solvers, which typically operate on a hierarchy of grids, in a clever way, resulting in a powerful solver for the full solution space.

We use multigrid V-cycles in our Stokes preconditioner to approximate the inverse viscous block required in (4.7), i.e., computing $\tilde{\mathbf{A}}^{-1}$, and to approximate the inverses of the discrete pressure Poisson operators in (6.6),

$$\tilde{\mathbf{K}}_l^{-1} \approx \mathbf{K}_l^{-1} \coloneqq (\mathbf{B}\mathbf{C}_{w_l}^{-1}\mathbf{B}^{\mathsf{T}})^{-1} \quad \text{and} \quad \tilde{\mathbf{K}}_r^{-1} \approx \mathbf{K}_r^{-1} \coloneqq (\mathbf{B}\mathbf{D}_{w_r}^{-1}\mathbf{B}^{\mathsf{T}})^{-1}. \tag{7.1}$$

These approximations are crucial for overall Stokes solver performance and scalability and are addressed in this chapter. For brevity, we limit our discussion to $\tilde{\mathbf{K}}^{-1} \coloneqq \tilde{\mathbf{K}}_r^{-1}$ since the results also hold for $\tilde{\mathbf{K}}_l^{-1}$.

The state of the art in extreme-scale multilevel solvers is exemplified by the Hybrid Hierarchical Grids (HHG) geometric multigrid (GMG) method [55], the GMG solver underlying the UG package [94], the algebraic multigrid (AMG) solver BoomerAMG from the hypre library [9], the multilevel balancing domain decomposition solver in FEMPAR [8], and the AMG solver for heterogeneous coefficients from the DUNE project [69]. These multigrid solvers have all been demonstrated to scale up to several hundred thousand cores (458K cores in some cases), but only for constant coefficient linear operators, uniformly-refined meshes, and low-order discretizations (with the exception of the DUNE solver, which has demonstrated scalability on a problem with heterogeneous coefficients but otherwise with uniform and low order grids). The complex PDE problems we target—characterized by advanced high-order discretizations, highly-locally adapted meshes, extreme (six orders-of-magnitude variation) heterogeneities, anisotropies, and severely nonlinear rheology—are significantly more difficult.

7.1 An abstract multigrid method

This section introduces fundamental ideas and concepts behind multigrid methods in an abstract framework. It will arrive at an error operator describing the discrepancy between exact solution and a multigrid approximation of the solution. This will guide the development of our multigrid methods in a subsequent section. The derivations presented here are based on the extensive literature around the topic of multigrid and multilevel solvers, notably [17, 20], which build on [18, 19, 21, 120], and the references therein. Other valuable sources for introductions to multigrid, in-depth treatment, and applications are [22, 58, 115].

General iterative processes

Let V be a Hilbert space of functions with the inner product denoted as $(\cdot\,,\cdot) \equiv (\cdot\,,\cdot)_V$ and let $V' = B(V, \mathbb{R})$ be the dual space to V, where $B(V, \mathbb{R})$ denotes the set of all linear bounded functionals from V to $\mathbb{R}$. On these spaces we define the operator

$$A : V \to V'$$

to be symmetric,

$$\langle Au\,, v\rangle_{V'\times V} = \langle u\,, Av\rangle_{V\times V'} \quad \text{for all } u, v \in V,$$

and positive definite,

$$0 < \langle Av\,, v\rangle_{V'\times V} \quad \text{for all } v \in V \setminus \{0\}.$$

We define the bilinear form associated to A by

$$A(u\,, v) \coloneqq \langle Au\,, v\rangle_{V'\times V} \quad \text{for } u, v \in V$$

and, since A is symmetric and positive definite, it induces an inner product with an associated norm:

$$\|u\|_A \coloneqq A(u\,, u)^{1/2} \quad \text{for } u \in V.$$

For any right-hand side $f \in V'$, the goal is to find a solution $u \in V$ of the linear system of equations

$$Au = f.$$

Equivalent ways to formulate the problem are

$$\langle Au\,, \phi\rangle_{V'\times V} = \langle f\,, \phi\rangle_{V'\times V} \quad \text{and} \quad A(u\,, \phi) = \langle f\,, \phi\rangle_{V'\times V}$$

for all $\phi \in V$. Furthermore, we define an approximate inverse operator to A by

$$\tilde{A}^{-1} : V' \to V$$

and its dual operator,

$$(\tilde{A}^{-1})' : V' \to V,$$

is defined via the relationship

$$\left\langle (\tilde{A}^{-1})' f \, , g \right\rangle_{V \times V'} = \left\langle f \, , \tilde{A}^{-1} g \right\rangle_{V' \times V} \quad \text{for all } f, g \in V'.$$

Next, we introduce the general concepts of a Picard iteration, error propagation operators, and iterative processes and link them together in Lemma 7.1.4.

Definition 7.1.1 (Preconditioned Picard iteration)**.** Given operator A defined as above, a right-hand side $f \in V'$, and an initial guess u^0, a preconditioned Picard fixed-point iteration seeks to approximate the solution of the preconditioned linear system

$$\tilde{A}^{-1} A u = \tilde{A}^{-1} f$$

via the iteration process

$$u^{n+1} \leftarrow u^n - \tilde{A}^{-1}(A u^n - f) = (I - \tilde{A}^{-1} A)\, u^n + \tilde{A}^{-1} f,$$

where I denotes the identity operator.

Definition 7.1.2 (Error operator)**.** If $u^* \in V$ denotes the exact solution of $Au = f$, then we can define the error at the n-th iteration by

$$e^n \coloneqq u^n - u^* \in V$$

and we call a linear map

$$\mathcal{E} : V \to V \quad \text{s.t.} \quad e^n = \mathcal{E}\, e^{n-1}$$

that propagates the error from one iteration to the next the error operator.

Definition 7.1.3 (Iterative process)**.** Given a right-hand side $f \in V'$ and an initial guess u^0, define an iterative process for solving for $u \in V$ in $Au = f$ by the map

$$\mathcal{I} : V \times V' \to V \quad \text{s.t.} \quad u^n = \mathcal{I}\left(u^{n-1} \, , f\right), \quad 0 < n.$$

$\mathcal{I}(\cdot\,,\cdot)$ is called consistent with $Au = f$ if the exact solution u^* is a fixed point,

$$\mathcal{I}(u^* \, , f) = u^*.$$

$\mathcal{I}(\cdot\,,\cdot)$ is linear if

$$\mathcal{I}(u + v \, , f + g) = \mathcal{I}(u \, , f) + \mathcal{I}(v \, , g) \quad \text{for all } u, v \in V,\ f, g \in V'$$

and

$$\mathcal{I}(\mu u \, , \mu f) = \mu\, \mathcal{I}(u \, , f) \quad \text{for all } u \in V,\ f \in V',\ \mu \in \mathbb{R}.$$

Lemma 7.1.4 (Iterative process equivalence)**.** *Let $\mathcal{I}(\cdot\,,\cdot)$ be an iterative process as in Definition 7.1.3. The following statements are equivalent:*

(i) $\mathcal{I}(\cdot\,,\cdot)$ is linear and consistent.

(ii) The error $e^n \in V$ at iteration n is connected to the error at the previous iteration via an error operator

$$\mathcal{E} : V \to V \quad s.t. \quad e^n = \mathcal{E}\, e^{n-1}$$

(iii) The map $\mathcal{I}(\cdot\,,\cdot)$ is given by a preconditioned Picard iteration

$$\mathcal{I}(u^n\,, f) = (I - \tilde{A}^{-1}A)\, u^n + \tilde{A}^{-1} f$$

Proof. See [20, p. 183]. □

The following two corollaries analyze properties of the error operator and provide estimates for the error. The result of the first corollary follows directly from the definitions above.

Corollary 7.1.5. *The error operator $\mathcal{E}$ of a linear and consistent iterative process $\mathcal{I}(\cdot\,,\cdot)$ has the form*

$$\mathcal{E}\,(\cdot) = \mathcal{I}(\cdot\,, 0) = (I - \tilde{A}^{-1}A)\,(\cdot).$$

Corollary 7.1.6 (Error reduction estimates)**.** *Let $\tilde{A}^{-1}$ be an approximation of the inverse of A and let $\mathcal{E} = I - \tilde{A}^{-1}A$ be the error operator of the iterative process induced by $\tilde{A}^{-1}$.*

(i) For general $\tilde{A}^{-1}$, the following error reduction estimate is satisfied

$$\|e^n\|_A \leq \|\mathcal{E}\|_A^n \left\|e^0\right\|_A = \left\|I - \tilde{A}^{-1}A\right\|_A^n \left\|e^0\right\|_A . \tag{7.2}$$

(ii) If $\tilde{A}^{-1}$ is symmetric the estimate becomes

$$\|e^n\|_A \leq \sup_{\lambda \in \sigma(\tilde{A}^{-1}A)} |1 - \lambda|^n \left\|e^0\right\|_A , \tag{7.3}$$

where $\sigma(\tilde{A}^{-1}A)$ is the spectrum of $\tilde{A}^{-1}A$. For convergence, it is essentially required that $\sup_{\lambda \in \sigma(\tilde{A}^{-1}A)} |\lambda|$ is bounded by a constant, and if $\tilde{A}^{-1}A$ is positive definite we can omit taking the absolute value. If this constant exists, we simply rescale $\tilde{A}^{-1}A$ such that $\sup_{\lambda \in \sigma(\tilde{A}^{-1}A)} |1 - \lambda| < 1$.

(iii) The error reduction can be stated in the form

$$\|e^n\|_A \leq \rho^n \left\|e^0\right\|_A , \tag{7.4}$$

where $\rho > 0$ is found through the estimate

$$A(\mathcal{E}v\,, v) \leq \rho^2 A(v\,, v) \quad \textit{for all } v \in V.$$

Therefore, the iterative process is convergent if $\rho < 1$.

(iv) Given a tolerance $\epsilon > 0$, the number of iterations n for a convergent (i.e., $\rho < 1$) iterative process to reduce the error relative to the initial error by

$$\frac{\|e^n\|_A}{\|e^0\|_A} \leq \epsilon$$

can be estimated by

$$n \leq \frac{\log(1/\epsilon)}{\log(1/\rho)}. \tag{7.5}$$

Proof. Note that estimate (7.4) is equivalent to (7.2). To prove (7.2), consider

$$\|e^n\|_A \leq \left\|I - \tilde{A}^{-1}A\right\|_A \|e^{n-1}\|_A,$$

where

$$\left\|I - \tilde{A}^{-1}A\right\|_A := \sup_{v \in V \setminus \{0\}} \frac{\left\|(I - \tilde{A}^{-1}A)\, v\right\|_A}{\|v\|_A}.$$

Then Equation (7.2) follows by applying the upper estimate multiple times:

$$\|e^n\|_A \leq \left\|I - \tilde{A}^{-1}A\right\|_A \|e^{n-1}\|_A \leq \ldots \leq \left\|I - \tilde{A}^{-1}A\right\|_A^n \|e^0\|_A.$$

If $\tilde{A}^{-1}$ is symmetric, then $\tilde{A}^{-1}A$ is symmetric with respect to the $A(\cdot\,,\cdot)$ inner product and we obtain

$$\left\|I - \tilde{A}^{-1}A\right\|_A^2 = \sup_{v \in V \setminus \{0\}} \frac{\left|A\left((I - \tilde{A}^{-1}A)^2 v\,,\, v\right)\right|}{A(v\,,\,v)} = \sup_{\lambda \in \sigma(\tilde{A}^{-1}A)} |1 - \lambda|^2$$

giving result (7.3).

For the iterations count estimate (7.5), let $\epsilon > 0$, then seek n such that

$$\frac{\|e^n\|_A}{\|e^0\|_A} \leq \rho^n \leq \epsilon.$$

Taking the logarithm leads to

$$n \leq \log_\rho(\epsilon) = \frac{\log(\epsilon)}{\log(\rho)} = \frac{\log(1/\epsilon)}{\log(1/\rho)},$$

which shows (7.5). □

Many iterative methods—including multigrid—successively apply several different iterative processes. The error operator of these composites is considered in the following lemma.

Lemma 7.1.7 (Composition of iterative processes)**.** *Let $\mathcal{I}_1$ and $\mathcal{I}_2$ be linear and consistent iterative processes with associated error operators $\mathcal{E}_1$ and $\mathcal{E}_2$. Then the iterative process*

$$\mathcal{I} := \mathcal{I}_2 \circ \mathcal{I}_1$$

is linear and consistent and its error operator satisfies

$$\mathcal{E} = \mathcal{E}_2\, \mathcal{E}_1.$$

Proof. By Lemma 7.1.4 we can write the linear, consistent iterative processes in the form

$$\mathcal{I}_i(u\,,f) = (I - \tilde{A}_i^{-1}A)\,u + \tilde{A}_i^{-1}f \quad \text{for } i = 1,2$$

yielding

$$\begin{aligned}\mathcal{I}(u\,,f) = \mathcal{I}_2 \circ \mathcal{I}_1(u\,,f) &= (I - \tilde{A}_2^{-1}A)\left((I - \tilde{A}_1^{-1}A)\,u + \tilde{A}_1^{-1}f\right) + \tilde{A}_2^{-1}f \\ &= (I - \tilde{A}_2^{-1}A)\,(I - \tilde{A}_1^{-1}A)\,u + (\tilde{A}_1^{-1} + \tilde{A}_2^{-1} - \tilde{A}_2^{-1}A\tilde{A}_1^{-1})\,f.\end{aligned}$$

This implies that $\mathcal{I}(\cdot\,,\cdot)$ is linear and consistent and we find its error operator by evaluating

$$\mathcal{E} = \mathcal{I}_2 \circ \mathcal{I}_1(\cdot\,,0) = (I - \tilde{A}_2^{-1}A)\,(I - \tilde{A}_1^{-1}A) = \mathcal{E}_2\,\mathcal{E}_1.$$

□

Framework for multigrid methods

This section introduces essential components of multigrid methods and shows their properties. These components constitute a framework that is later used to define specific multigrid methods. Two fundamental components of multigrid methods are a sequence of grids, typically a hierarchy of subspaces, and a corresponding sequence of operators defined on these grids.

Definition 7.1.8 (Multigrid hierarchy)**.** The multigrid hierarchy is composed of a nested sequence of finite dimensional subspaces V_k, $1 \le k \le J$, such that

$$V_1 \subset V_2 \subset \cdots \subset V_J \subset V.$$

Definition 7.1.9 (Auxiliary operators)**.** For $1 \le k \le J$ and $v \in V$ define the auxiliary operators to be

$$A_k : V_k \to V_k' \quad \text{s.t.} \quad \langle A_k v\,,\phi\rangle_{V'\times V} = \langle Av\,,\phi\rangle_{V'\times V} = A(v\,,\phi) \quad \text{for all } \phi \in V_k.$$

In order to formulate iterative processes at each subspace, or level, of the multigrid hierarchy, each auxiliary operator is accompanied by an approximative inverse, called smoothing operator.

Definition 7.1.10 (Smoothing operator)**.** A smoothing operator is a linear map

$$R_k : V_k' \to V_k$$

that is an approximation to the inverse of A_k. Therefore, it induces an iterative process

$$u^{n+1} \leftarrow (I - R_k A)\,u^n + R_k f,$$

which is called smoothing or relaxation (hence the letter R). We automatically obtain the dual of the smoothing operator

$$R_k' : V_k' \to V_k \quad \text{s.t.} \quad \left\langle R_k' f\,,g\right\rangle_{V\times V'} = \langle f\,,R_k g\rangle_{V'\times V} \quad \text{for all } f,g \in V_k'.$$

Transitions in between subsequent levels of the multigrid hierarchy are facilitated by projection operators, which are described by the following definitions.

Definition 7.1.11 ($A(\cdot\,,\cdot)$-orthogonal projector)**.** For $1 \leq k \leq J$, the linear map

$$\Pi_k : V \to V_k \quad \text{s.t.} \quad A(\Pi_k v\,,\phi) = A(v\,,\phi) \quad \text{for all } \phi \in V_k$$

projects $v \in V$ onto V_k orthogonally to AV_k. This projector is also called *Galerkin projector*. Equivalently to the definition we can also write

$$\Pi_k : V \to V_k \quad \text{s.t.} \quad (I - \Pi_k) \perp_{A(\cdot,\cdot)} V_k.$$

The dual operator is

$$\Pi_k' : V_k' \to V' \quad \text{s.t.} \quad \left\langle \Pi_k' A v\,,\phi \right\rangle_{V'\times V} = \langle Av\,,\phi\rangle_{V'\times V} \quad \text{for all } \phi \in V,$$

where $v \in V_k$.

Definition 7.1.12 ($(\cdot\,,\cdot)$-orthogonal projector)**.** For $1 \leq k \leq J$, the linear map

$$P_k : V_k \to V \quad \text{s.t.} \quad (P_k v\,,\phi) = (v\,,\phi) \quad \text{for all } \phi \in V$$

projects $v \in V_k$ onto V orthogonally to V. It projects from a lower- to a higher-dimensional space and is therefore called *interpolation* or *prolongation* (hence the letter P). Equivalently to the definition we can also write

$$P_k : V_k \to V \quad \text{s.t.} \quad (I - P_k) \perp_{(\cdot,\cdot)} V.$$

This implies the dual operator

$$P_k' : V' \to V_k' \quad \text{s.t.} \quad \left\langle P_k' g\,,\phi \right\rangle_{V'\times V} = \langle g\,,\phi\rangle_{V'\times V} \quad \text{for all } \phi \in V_k,$$

where $g \in V$. The dual projector is also called *restriction*.

To project between more than one level of the multigrid hierarchy, the projection operators from above have to be applied consecutively. Properties of these compositions are discussed next.

Lemma 7.1.13 (Composition of projectors)**.** *Let Π_k and P_k be projectors as in Definitions 7.1.11 and 7.1.12. Then*

(i) "Prolongation then Galerkin projection" gives the identity operator,

$$\Pi_k P_k = I : V_k \to V_k. \tag{7.6}$$

(ii) "Galerkin projection then prolongation,"

$$P_k \Pi_k : V \to V_k \tag{7.7}$$

gives a projector that is $A(\cdot\,,\cdot)$-orthogonal.

(iii) Its perpendicular projector

$$I - P_k \Pi_k : V \to V_k^{\perp_{A(\cdot,\cdot)}} \tag{7.8}$$

gives a projector that is $A(\cdot\,,\cdot)$-orthogonal, where

$$V_k^{\perp_{A(\cdot,\cdot)}} \coloneqq \{v \in V \mid A(v\,,\phi) = 0 \text{ for all } \phi \in V_k\}\,,$$

Proof. First, let $v, \phi \in V_k$ and derive using the definition of the Galerkin projector

$$A(\Pi_k P_k v\,,\phi) = A(P_k v\,,\phi) = \langle A P_k v\,,\phi\rangle_{V' \times V}\,.$$

This yields, with the definition of the dual to the prolongation,

$$\langle A P_k v\,,\phi\rangle_{V' \times V} = \left\langle v\,, P_k' A \phi\right\rangle_{V \times V'} = \langle v\,, A\phi\rangle_{V \times V'} = A(v\,,\phi)\,,$$

which proves (7.6).

Next, we can show that $P_k \Pi_k$ is a projector,

$$(P_k \Pi_k)^2 = P_k \Pi_k P_k \Pi_k = P_k (\Pi_k P_k) \Pi_k = P_k \Pi_k,$$

since it is idempotent and using result (7.6). The range of $P_k \Pi_k$ satisfies $\mathrm{Ran}(P_k \Pi_k) = V_k$ since, for $v \in V$,

$$u \coloneqq \Pi_k v \in V_k \quad \Rightarrow \quad P_k u = u \in V_k.$$

Further, $\mathrm{Ker}(P_k \Pi_k) = V_k^{\perp_{A(\cdot,\cdot)}}$ because the kernel of the $A(\cdot\,,\cdot)$-orthogonal projector Π_k satisfies $\mathrm{Ker}(\Pi_k) = V_k^{\perp_{A(\cdot,\cdot)}}$ by definition. Then we have that

$$\mathrm{Ran}(P_k \Pi_k) = \mathrm{Ker}(P_k \Pi_k)^{\perp_{A(\cdot,\cdot)}},$$

since V_k is a closed subspace of V, which proves that $P_k \Pi_k$ in (7.7) is $A(\cdot\,,\cdot)$-orthogonal.

The result for the perpendicular projector (7.8) is obtained by fundamental properties of projectors. □

We conclude this section by discussing relationships between projections and differentiation operators.

Lemma 7.1.14 (Commutator relationship between projection and differentiation)**.** *Let*

$$1 \le i < j \le J \quad \text{or} \quad 1 \le i \le J \text{ and } j = \infty.$$

Then

$$P_i' A_j = A_i \Pi_i : V_j \to V_i',$$

where in case of $j = \infty$, $A_j \equiv A$ and $V_j \equiv V$.

Proof. Let $v \in V_j$, $\phi \in V_i$ and consider

$$\left\langle P_i' A_j v \,,\, \phi \right\rangle_{V' \times V} = \langle A_j v \,,\, \phi \rangle_{V' \times V} = A(v \,,\, \phi) = A(\Pi_i v \,,\, \phi) = \langle A_i \Pi_i v \,,\, \phi \rangle_{V' \times V} \,,$$

where we used Definitions 7.1.11 and 7.1.12. □

Corollary 7.1.15 (Auxiliary operator projection)**.** *For* $1 \leq k \leq J$,

$$P_k' A P_k = A_k : V_k \to V_k'.$$

Proof. Result follows from Lemma 7.1.14 and (7.6). □

A multigrid method with correction scheme

The multigrid V-cycle is a recursive iterative process. Algorithm 7.1.1 and Definition 7.1.16 state this V-cycle with one smoothing iteration before and one after the coarse grid correction. In practice it is common to perform multiple smoothing iterations, in which case it is straightforward to extend Algorithm 7.1.1. However, for the analysis of multigrid in this section we fix the number of smoothing iterations to one.

Algorithm 7.1.1 Multigrid V-cycle, $MG_k : V_k' \to V_k$

input $g \in V_k'$
if $k = 1$ **then**
  $v \leftarrow MG_1 g \coloneqq A_1^{-1} g$ ▷ direct solve
else
  $v \leftarrow 0$
  $v \leftarrow v - R_k'(A_k v - g)$ ▷ smoothing iteration
  $v \leftarrow v - P_{k-1} MG_{k-1} P_{k-1}'(A_k v - g)$ ▷ coarse correction
  $v \leftarrow v - R_k(A_k v - g)$ ▷ smoothing iteration
end if
return $v \in V_k$

Definition 7.1.16 (Multigrid V-cycle operator and iterative process)**.** From Algorithm 7.1.1 we can deduce the multigrid V-cycle operator

$$\begin{aligned} MG_k &= R_k + R_k' - R_k A_k R_k' + (I - R_k A_k)\, P_{k-1} MG_{k-1} P_{k-1}' \,(I - R_k' A_k) \\ &= R_k + R_k' - R_k A_k R_k' + (I - R_k A_k)\, C_{k-1} \,(I - R_k' A_k), \end{aligned}$$

where we defined the coarse correction operator

$$C_{k-1} \coloneqq P_{k-1} MG_{k-1} P_{k-1}'.$$

These lead to the definition of the multigrid V-cycle iterative process

$$\mathcal{I}_{MG_k}(v \,,\, g) = v - MG_k(A_k v - g) = (I - MG_k A_k)\, v + MG_k g. \tag{7.9}$$

The next lemma and corollary discuss the error of the V-cycle and allow us to deduce practical guidelines for designing multigrid methods.

Lemma 7.1.17 (Multigrid V-cycle error operator)**.** *The error operator for the multigrid V-cycle iterative process is*

$$\mathcal{E}_k = \mathcal{E}_{R_k}\,\mathcal{E}_{C_{k-1}}\,\mathcal{E}_{R'_k} = (I - R_k A_k)\,(I - P_{k-1} M G_{k-1} P'_{k-1} A_k)\,(I - R'_k A_k) \tag{7.10}$$

with the definitions

$$\mathcal{E}_{R_k} \coloneqq I - R_k A_k \quad \textit{and} \quad \mathcal{E}_{R'_k} \coloneqq I - R'_k A_k$$

and

$$\mathcal{E}_{C_{k-1}} \coloneqq I - P_{k-1} M G_{k-1} P'_{k-1} A_k. \tag{7.11}$$

Proof. This is a direct consequence of (7.9) and Lemma 7.1.7. □

Corollary 7.1.18 (Multigrid V-cycle error recursion)**.** *The error operator for the multigrid V-cycle satisfies the recursion relationship*

$$\mathcal{E}_k = \mathcal{E}_{R_k}\,(I - P_{k-1}\Pi_{k-1} + P_{k-1}\mathcal{E}_{k-1}\Pi_{k-1})\,\mathcal{E}_{R'_k}. \tag{7.12}$$

Proof. Consider the coarse correction error operator (7.11) and apply the commutator relationship between projection and differentiation from Lemma 7.1.14

$$\mathcal{E}_{C_{k-1}} = I - P_{k-1} M G_{k-1} P'_{k-1} A_k = I - P_{k-1} M G_{k-1} A_{k-1} \Pi_{k-1}.$$

Then by including a zero term, we get

$$\begin{aligned}\mathcal{E}_{C_{k-1}} &= I - P_{k-1}\Pi_{k-1} + P_{k-1}\,(I - M G_{k-1} A_{k-1})\,\Pi_{k-1} \\ &= I - P_{k-1}\Pi_{k-1} + P_{k-1}\mathcal{E}_{k-1}\Pi_{k-1},\end{aligned}$$

which shows the result after substitution into Equation (7.10). □

The error operator of the multigrid V-cycle in Equation (7.12) is an important guide for implementing and applying multigrid methods in practice. It shows how errors from smoothing iterations, $\mathcal{E}_{R_k}$ and $\mathcal{E}_{R'_k}$, and coarse correction, $\mathcal{E}_{C_{k-1}}$, interact and the role of the projection operators, P_{k-1} and Π_{k-1}, when these errors are combined. Initially, the smoothing iterations reduce the error for a relatively narrow high-frequency part of the spectrum within the space V_k, where spectrum refers to modes or eigenvalues of the differential operator A_k. Afterwards, the coarse correction reduces the error in the remaining lower-frequency part of the spectrum, which acts within the space V_{k-1}. For the projection operators, accuracy and filtering properties between spaces V_k and V_{k-1} are important. It is desired that Π_{k-1} separates low frequencies as well as possible when restricting the residual onto V_{k-1} and, in turn, P_{k-1} introduces as little high-frequency pollution as possible when interpolating the correction into the space V_k. The accuracy of the projections contributes to the error in Equation (7.12) by

means of the term $(I - P_{k-1}\Pi_{k-1})$. The final smoothing iteration after the coarse correction accounts for inaccuracies in the projections by reducing high-frequency errors in the interpolated correction.

When designing a multigrid method, the efficacy of the preconditioner, which we can express by the error (7.12), as well as the computational complexity are important. Typically, improving efficacy increases computational cost and vice versa, hence, a feasible balance needs to be found. With this discussion we want to demonstrate that the interaction between smoothers and projection operators plays a central part in the design of a multigrid method rather than its individual components in isolation. Hence, we aim for balancing the trade-offs between efficacy and computational cost. An additional and significant challenge to designing an overall efficient method is added through large-scale parallelism. The next section presents a multigrid method with a hierarchy that is generated by spectral, geometric, and algebraic coarsening. It operates on locally adaptively refined, high-order finite element discretizations and is capable of effective preconditioning in the presence of highly varying coefficients. Furthermore, we demonstrate parallel scalability to extreme scales.

7.2 Hybrid spectral–geometric–algebraic multigrid (HMG)

In this section, we develop a hybrid spectral–geometric–algebraic multigrid method, which exhibits extreme parallel scalability and retains nearly optimal algorithmic scalability (see Chapter 8 for scalability results).

While traversing the HMG hierarchy shown in Figure 7.1, HMG initially reduces the discretization order (spectral multigrid); after arriving at order one, it continues by coarsening mesh elements (geometric multigrid); once the degrees of freedom fall below a threshold, algebraic multigrid (AMG) carries out further coarsening until a direct solve can be computed efficiently. During parallel geometric coarsening, the number of compute cores and the size of the MPI communicator is reduced successively to minimize communication. Re-discretization of the differential equations is performed on each coarser spectral and geometric level. The viscosity values in each element are stored at the quadrature points of the velocity discretization, and are thus local to each element. The viscosity coarsening is done level-by-level during the setup phase. The coarsening operator is the adjoint of the refinement operator, which performs element-wise interpolation. This adjoint is computed with respect to the L^2-inner products, and since viscosity values are not shared amongst elements, this does not require (an approximation of) a global mass matrix solve. The transition from geometric to algebraic multigrid is done at a sufficiently small core count and small MPI communicator. AMG continues to further reduce problem size (via Galerkin coarse grid projection) and the number of cores down to a single core for the direct solver.

The approximation of the inverse viscous block $\tilde{\mathbf{A}}^{-1}$ is well suited for multigrid V-cycles. The operator $\mathbf{K} = \mathbf{K}_r$ (see Equation (7.1)) is regarded as a discrete, variable-coefficient Poisson operator on the discontinuous pressure space $\mathbb{P}_{k-1}^{\text{disc}}$ with Neumann boundary conditions. Therefore, multigrid V-cycles can also be employed to approximate the inverse $\tilde{\mathbf{K}}^{-1}$. However, it turned out to be problematic

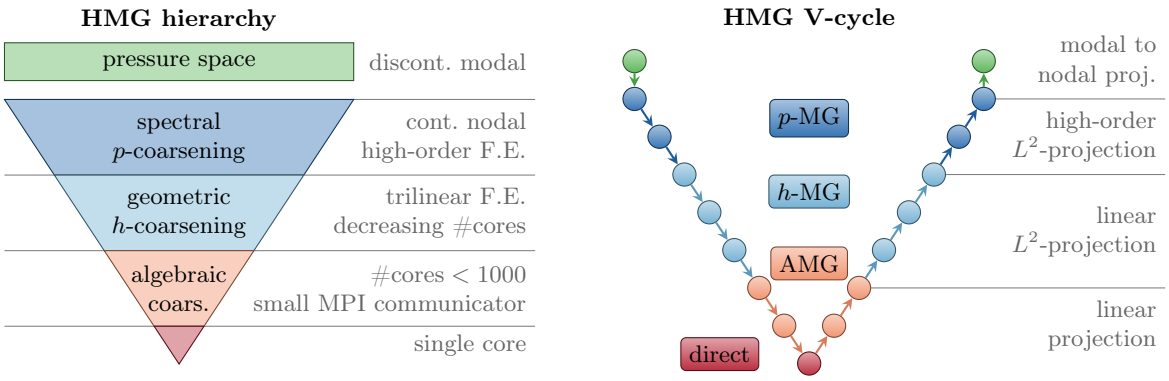


Figure 7.1: Hybrid spectral–geometric–algebraic multigrid (HMG). *Left*: Illustration of multigrid hierarchy. From top to bottom, first, the multigrid levels are obtained by spectral coarsening *(dark blue)*. Next, the mesh is geometrically coarsened and repartitioned on successively fewer cores to minimize communication *(light blue)*. Finally, AMG further reduces problem size and core count *(light red)*. The multigrid hierarchy for the pressure Poisson operator $\mathbf{K}$ additionally involves smoothing in the discontinuous, modal pressure space *(green)*. The projection from the discontinuous, modal to a continuous finite element nodal basis uses a lumped mass matrix in the nodal space to avoid the global mass matrix system solve. *Right*: The multigrid V-cycle consists of smoothing at each level of the hierarchy *(circles)* and intergrid transfer operators (*arrows downward* for restriction and *arrows upward* for interpolation). To enhance efficacy of the the V-cycle as a preconditioner, different types of projection operators are employed for these operators depending on the phase within the V-cycle.

to apply multigrid coarsening directly due to the discontinuous, modal discretization of the pressure. We take a novel approach (see also [97]) by considering the underlying infinite-dimensional, variable-coefficient Poisson operator, where the coefficient is derived from the diagonal weighting matrix (here, $\mathbf{C}_{w_l}^{-1}$ or $\mathbf{D}_{w_r}^{-1}$). Then we re-discretize with continuous, nodal high-order finite elements in $\mathbb{Q}_k$. An alternative would be to use $\mathbb{Q}_{k-1}$, but we prefer to use $\mathbb{Q}_k$ since the corresponding data structures are readily available from the discretization of the velocity. Hence, this choice avoids the setup cost related to discretization-specific parameters and their storage. Additionally, the HMG hierarchy of the preconditioner acting on the velocity can be partially reused, again saving setup time and memory. This continuous, nodal discretization of the Poisson operator is then approximately inverted with an HMG V-cycle that is similar to the one described above for the inverse viscous block approximation $\tilde{\mathbf{A}}^{-1}$. Additional smoothing is applied in the discontinuous pressure space (Figure 7.1, *green level*) to account for high frequency modes in residuals that are introduced through projections between $\mathbb{Q}_k$ and $\mathbb{P}_{k-1}^{\text{disc}}$. Moreover, when mapping residuals and updates between the continuous, nodal and discontinuous, modal spaces, the null space of constant mean pressure is enforced via projections as described in Section 4.2.

Our hybrid multigrid method combines high-order L^2-restriction and interpolation operators and employs Chebyshev-accelerated point-Jacobi smoothers. The choice of intergrid projections with high accuracy that matches the polynomial order of the finite element shape functions is motivated by the discussion of design guidelines in Section 7.1 and the observations regarding the multigrid error (7.12).

It enables a computationally more efficient and simpler smoother. While the implementation and parallelization of the point-Jacobi smoother is straightforward, we carefully designed our parallel intergrid projections and carried out numerous optimizations of the implementation to reduce computational costs and communication overheads (see Section 7.3). This results in optimal or nearly optimal algorithmic multigrid performance (see Section 8.1), i.e., iteration numbers are independent of mesh size and only mildly dependent on discretization order, while maintaining robustness with respect to highly heterogeneous coefficients. In addition, the efficacy of the HMG preconditioner does not deteriorate with increasing core counts, because the spectral and geometric multigrid is by construction independent of the number of cores and AMG is invoked for prescribed small problem sizes on essentially fixed small core counts.

For all numerical experiments presented here and in Chapter 8, three pre- and post-smoothing iterations with a Chebyshev accelerated point-Jacobi smoother are performed. PETSc's [10] implementations of Chebyshev acceleration, direct solver, AMG (called GAMG), and GMRES are used.

7.3 Implementation and optimization

The construction of the HMG hierarchy requires parallel geometric coarsening. Recall that we discretize Earth's mantle using locally adaptively refined hexahedral meshes. Extreme local refinement is critical to resolve plate boundaries down to a few hundred meters, while away from these regions significantly coarser meshes can be used that still capture global-scale behavior.

Parallel adaptive forest-of-octrees algorithms, implemented in the p4est parallel AMR library, are used for efficient parallel mesh refinement and coarsening, mesh balancing, and repartitioning [26,30,70]. The representation of the mesh as an octree topology (see *top left graph* in Figure 7.2) enables fast refinement and coarsening, which is performed locally on each processor core without communication. Moreover, the octree structure allows for efficient 2:1 mesh balancing, for which communication is necessary. Space filling curves transform elements of a two- or three-dimensional space into a (one-dimensional) sequence (see *top right mesh* in Figure 7.2). This sequence is used for an efficient partitioning of mesh elements in parallel with the desirable property of clustering neighboring elements in the sequence.

During parallel geometric coarsening, repartitioning of locally adapted meshes across compute cores has to be performed for load balancing. As the number of elements in the mesh is decreasing, the number of cores that contain elements is reduced successively to minimize communication during multigrid cycles. Simultaneously, we reduce the MPI communicator such that it contains only non-empty cores. We refer to this procedure as "core-thinning." To ensure coarsening across core boundaries, we partition a family of elements that can be coarsened in the next sweep on the same core. The geometric coarsening is visualized in Figure 7.3.

From a high-level perspective, the challenge of a parallel multigrid implementation is to balance

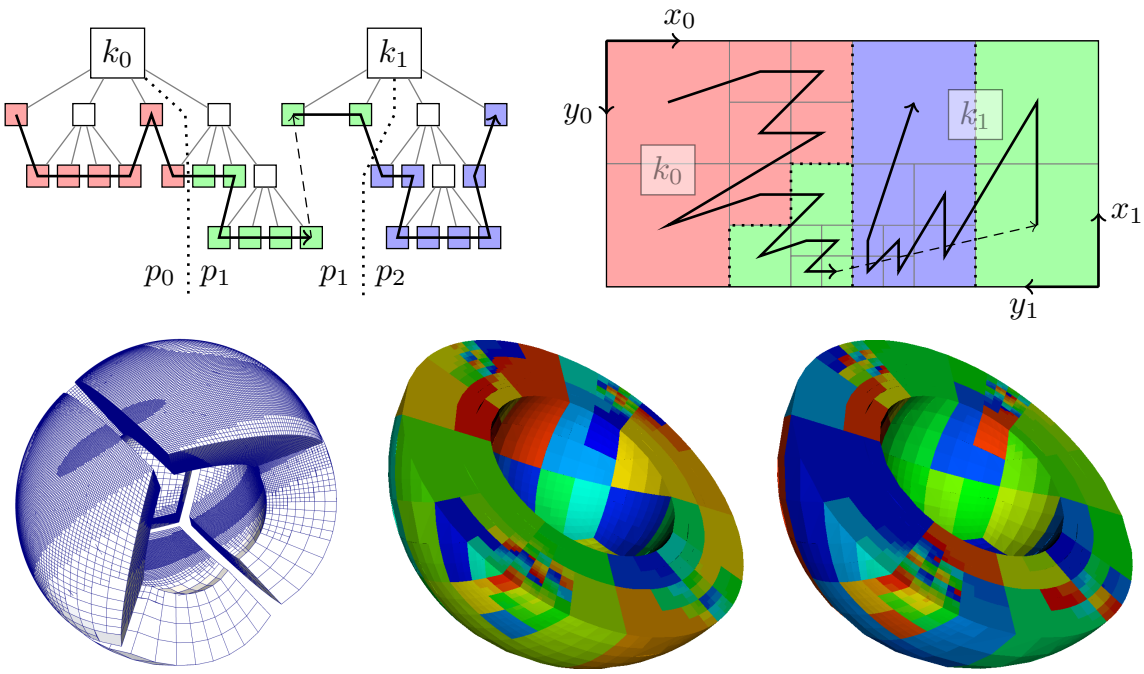


Figure 7.2: *Top:* A forest-of-octree topology with two octrees, k_0 and k_1, is shown in the *top left graph* including a space filling curve connecting the leaves of the octrees. The corresponding mesh with space filling curve is depicted in the *top right*. The three *colors* label different processor cores, p_0, p_1, p_2. *Bottom:* A three-dimensional mesh wireframe and two example partitions *(colors)*. (Credit: Burstedde, et al.)

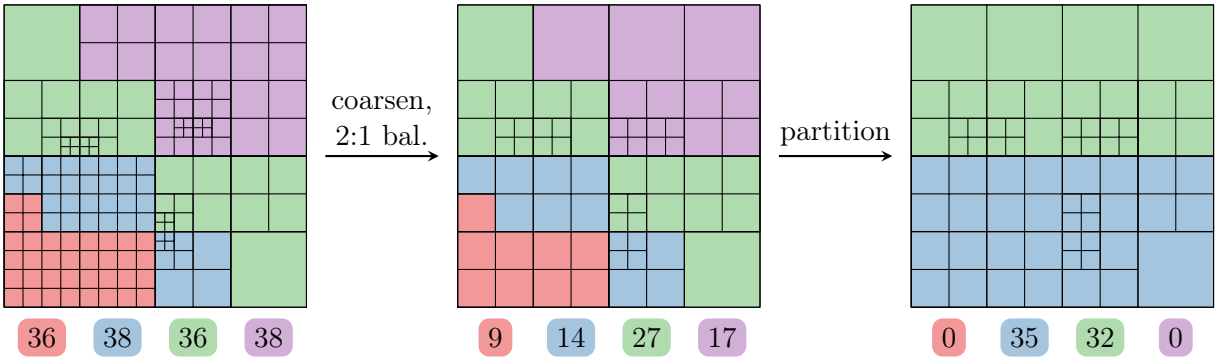


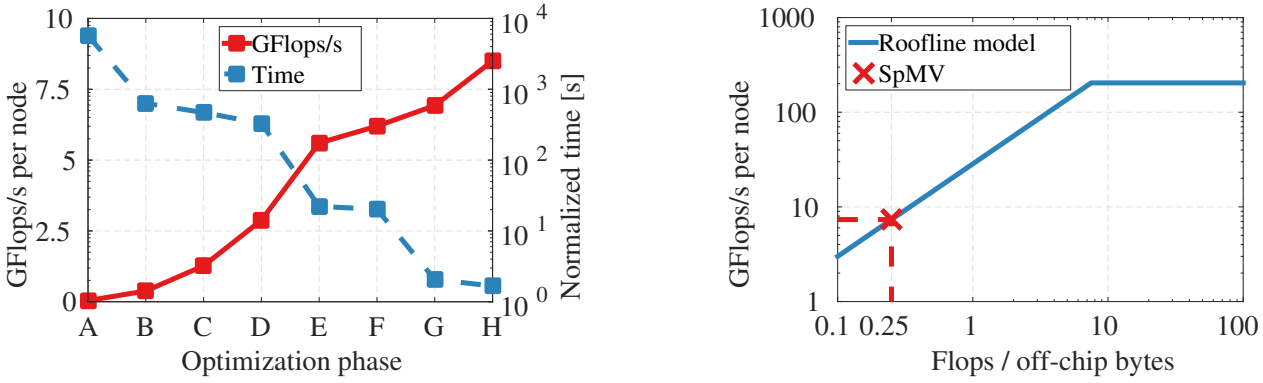
Figure 7.3: HMG geometric coarsening with repartitioning and core-thinning. *Colors* depict four different processor cores, *numbers* indicate element count on each core. In the first step, an evenly distributed locally refined mesh is coarsened and balanced on each processor core without communication, resulting in varying numbers of elements. In the second step, the mesh elements are partitioned evenly across processor cores leaving two of the cores empty due to the coarseness of the mesh.

the performance of two critical components: *(i)* application of differential operators during smoothing, commonly referred to as MatVecs, and *(ii)* intergrid transfer operators that perform restriction and interpolation between multigrid levels. Further, this balance has to be maintained as the number of cores grows to extreme scales. Both MatVecs and intergrid operators rely on point-to-point communication such that optimizing the runtime of one deteriorates the performance of the other: MatVecs benefit from even parallel distribution and clustering of local degrees of freedom, while intergrid operators benefit from bundling degrees of freedom on fewer cores to reduce the amount of communication. In the case of our complex mantle flow solver, we deal with four different kinds of MatVecs (viscous stress $\mathbf{A}$, divergence/gradient $\mathbf{B}/\mathbf{B}^\mathsf{T}$, continuous nodal Poisson operator $\mathbf{K}$, and Stokes operator) and six different intergrid operators (restriction and interpolation for each of: modal to nodal projection, p-projection in spectral multigrid, and h-projection in geometric multigrid). Optimization efforts have to target all of these operators to be successful. Additionally, this task becomes even more complex because the HMG V-cycle has to be performed on unstructured, highly locally-adapted meshes.

In order to obtain optimal load balance for MatVecs during the V-cycle, we repartition the coarser multigrid levels uniformly across the cores and gradually reduce the size of the MPI communicators as we progress through the coarser levels. The reduction of the MPI size is done such that neither MatVecs nor intergrid transfer operations become a bottleneck at large scale. Moreover, point-to-point communication is overlapped with computations for optimal scalability. No collective communication is used in the V-cycle. The HMG setup cost is minimized with a matrix-free approach for differential and intergrid operators, which additionally produces a lightweight memory footprint.

These key principles were at the foundation of our extreme-scale multigrid implementation. Further improvements of time-to-solution and performance were carried out in a number of successive optimization steps (see Figure 7.4a). Overall, we decreased the time to solution for the BG/Q hardware architecture (see Section 8.2) by a factor of over 1000 and increased performance on a BG/Q compute node by a factor of $\sim$200. Therefore, our complex mantle flow solver as a whole, including spectral, geometric and algebraic multigrid phases on highly adaptively refined meshes, performs similarly to a routine for sparse matrix-vector multiplications. This is supported by the roofline model analysis [95], from which we obtain an optimal performance of approximately 8 GFlops for sparse MatVecs (Figure 7.4b). Note that implicit solvers for PDEs inherently exhibit a sparsity structure and hence performance will always be memory-bound, which suggests that our memory-bound solver's computational performance is close to optimal. This is further supported by numerical results in Chapter 8.

Further optimizations regarding the multigrid hierarchy setup at scale were performed by means of changing the distribution of mesh elements at coarser levels across cores. In particular, we focused on the runtime of mesh coarsening and 2:1 balancing algorithms followed by repartitioning (see Figure 7.3). Among these, the 2:1 balancing turned out to be a bottleneck. Mesh balancing requires information from neighboring elements to adjust the level of mesh refinement such that it differs by mostly one.



(a) (b)

Figure 7.4: (a) Performance improvement and time-to-solution reduction over a sequence of optimization steps (time is normalized by GMRES iterations per 1024 BG/Q nodes per billion DOF). *Pt. A* is base performance before optimization. *Pt. B:* reduction of blocking MPI communication. *Pt. C:* minimizing integer operations in inner MatVec for-loops and reducing the number of cache misses. *Pt. D:* computation of derivatives by applying precomputed CSR-matrices at the element level and SIMD vectorization. *Pt. E:* OpenMP threading of major loops in MatVecs. *Pt. F:* MPI communication reduction, overlapping with computations, and OpenMP threading in intergrid operators. *Pt. G:* low-level optimization of finite element kernels via improving flop-byte ratio and consecutive memory access, and better pipelining of floating point operations. *Pt. H:* various low-level optimizations including enforcement of boundary conditions and interpolation of hanging finite element nodes. (b) BG/Q node roofline model (theoretical peak performance) and SpMV performance with max flop-byte ratio of 0.25 [75].

Table 7.1: HMG's geometric hierarchy setup runtimes (in seconds) of mesh coarsening and 2:1 balancing algorithms followed by repartitioning. The threshold of elements per core that triggers repartitioning of coarse multigrid levels is lower in the *left half of columns* and higher in the *right half of columns*. Note that runtimes correspond to element counts one row lower, however, the number of cores involved in coarsening, 2:1 balancing, and partitioning are found in the same row as the times.

	Low elements/core threshold					**Improved elements/core threshold**				
h-MG level	#elems [$\times 10^6$]	#cores w/ elems	**elements per core**	coarsen, 2:1 bal.	partition	#elems [$\times 10^6$]	#cores w/ elems	**elements per core**	coarsen, 2:1 bal.	partition
0	114.59	32,768	**3497**	0.21	0.05	114.59	32,768	**3497**	0.22	0.05
1	25.47	32,768	**777**	0.19	0.05	25.47	32,768	**777**	0.28	0.44
2	5.81	32,768	**177**	10.56	0.15	5.81	23,230	**177**	6.95	0.99
3	1.49	11,624	**45**	0.80	0.13	1.49	5,952	**64**	0.29	0.11
4	.36	2,853	**31**	0.13	0.02	.37	1,461	**61**	0.07	0.01
5	.12	931	**42**	–	–	.12	477	**81**	–	–
0	231.26	65,536	**3529**	0.32	0.10	240.81	65,536	**3674**	0.22	0.09
1	64.89	65,536	**990**	0.73	0.10	30.10	65,536	**459**	0.28	0.56
2	17.95	65,536	**274**	56.86	0.10	6.67	26,672	**102**	1.00	1.08
3	4.55	65,536	**69**	42.70	0.14	1.69	6,745	**63**	0.22	0.13
4	1.15	8,963	**18**	0.98	0.08	.44	1,780	**66**	0.05	0.01
5	.33	2,548	**36**	0.09	0.02	.12	498	**70**	–	–
6	.11	866	**43**	–	–					

As the mesh gets coarser, the neighboring elements' information has to be communicated between increasing numbers of cores. This communication, when performed at large scales, eventually leads to congestions in the network. Our experiments in Table 7.1 report two problem setups *(top and bottom set of rows)* differing by the number of elements at the fine multigrid level (∼115M and ∼240M) and the number of BG/Q cores (32K and 64K). In addition, we change the threshold of elements per core that triggers repartitioning at coarse multigrid levels: a low threshold (i.e., fewer elements/core) in the *left half of columns* and a higher threshold (i.e., more elements/core) in the *right half of columns*. The experiments show increasing runtimes (in seconds) for balancing at coarser multigrid levels highlighted in *red*. The increase becomes more pronounced in the 64K core run, taking up to ∼50 s. At the same time the average number of elements/core at the time of balancing is below 50 and reaching only 18 at its lowest.[1] Increasing those thresholds means that more elements are assigned to reside on fewer cores, which at the same time keeps more cores idle. Its effect on balancing times are observed in the *right half of columns* of Table 7.1, where the previous ∼50 s on 64K cores are reduced[2] to ∼1 s (highlighted in *blue*).

7.4 HMG convergence rate and time-to-accuracy

We evaluate convergence properties of HMG when it is used as solver for Laplace's equation. Even though the Laplace operator is a simple problem relative to mantle convection, convergence for it is well understood and documented in the literature. It will serve as a verification of our HMG algorithms and implementation. The goal is to measure HMG's convergence rate and its computational cost for increasing resolutions in the discretization by means of both mesh refinement and higher finite element orders.

The model problem we utilize in this section's numerical experiments is defined on the open unit cube domain $\Omega = (0,1)^3$, which is discretized with continuous, nodal finite elements of polynomial order k on a uniform mesh of refinement level ℓ. We consider the boundary value problem consisting of the Laplace differential operator with homogeneous Dirichlet boundary conditions:

$$-\Delta u = f \quad \text{in } \Omega, \tag{7.13a}$$

$$u = 0 \quad \text{on } \partial\Omega. \tag{7.13b}$$

Since the Laplace operator is symmetric with respect to the L^2-inner product and positive definite, it induces the following inner product and norm

$$(u\,,u)_A \coloneqq (\nabla u\,,\nabla u)_{L^2(\Omega)} \quad \text{and} \quad \|u\|_A \coloneqq \sqrt{(u\,,u)_A}\,,$$

[1] Note that runtimes for coarsening, 2:1 balancing, and partitioning correspond to the mesh sizes one row lower, however, the number of cores involved are found in the same row as the times.

[2] Note that since also the mesh size is different in the slower performing result, there can be other factors at play too. Still, throughout all our numerical experiments the higher elements/core threshold showed better performance for multigrid hierarchy setups, which resulted in the extreme scalability presented in Section 8.4.

where $\|\cdot\|_A$ is referred to as energy norm.

Given integers $i, j, k \in \mathbb{N}$, we define an analytic solution for problem (7.13) by

$$u_{\text{anl}}(x, y, z) \coloneqq \sin(\pi i x) \sin(\pi j y) \sin(\pi k z) \quad \text{for } [x, y, z] \in \Omega$$

and the corresponding manufactured right-hand side

$$f_{\text{anl}}(x, y, z) = -\Delta u_{\text{anl}}(x, y, z) = \pi^2 \left(i^2 + j^2 + k^2\right) \sin(\pi i x) \sin(\pi j y) \sin(\pi k z) \quad \text{for } [x, y, z] \in \Omega.$$

In order to to measure the error of a numerical solution in the energy norm $\|\cdot\|_A$, as it is common (e.g., see [46]), we calculate the exact energy of the analytic solution, E_{anl}, using integration by parts

$$E_{\text{anl}} \coloneqq \|u_{\text{anl}}\|_A^2 = \int_\Omega \nabla u_{\text{anl}} \cdot \nabla u_{\text{anl}} = \int_\Omega f_{\text{anl}}\, u_{\text{anl}} = \frac{\pi^2}{8} \left(i^2 + j^2 + k^2\right).$$

The first numerical error that influences the accuracy of a computed solution is the discretization error:

$$\varepsilon_{\text{discr}} \coloneqq \frac{\sqrt{\|u_{\text{discr}}\|_A^2 - \|u_{\text{anl}}\|_A^2}}{\|u_{\text{anl}}\|_A} = \left(\frac{\|u_{\text{discr}}\|_A^2 - E_{\text{anl}}}{E_{\text{anl}}}\right)^{1/2}, \tag{7.14}$$

which measures the discrepancy between the analytic solution u_{anl} and the (nearly) best possible solution u_{discr} within the space spanned by a finite element basis.[3] Here, the discretization error $\varepsilon_{\text{discr}}$ depends on the polynomial order k of the finite element shape functions and the level of refinement ℓ of the uniform mesh. Related to the discretization error is the rate of error reduction between coarser and finer meshes with element side lengths H and h, respectively,

$$\frac{\log\left(\varepsilon_{\text{discr},H} / \varepsilon_{\text{discr},h}\right)}{\log\left(H/h\right)} \quad \text{for } h < H,$$

which is the discretization's convergence rate. Due to the uniform meshes we are considering in this section, the element sizes are $H = 2^{-L}$ for the coarse mesh with level L and $h = 2^{-\ell}$ for the fine mesh with level $\ell > L$.

The second numerical error contribution stems from the truncation error:

$$\varepsilon_{\text{trunc}} \coloneqq \frac{\|u_{\text{trunc}} - u_{\text{discr}}\|_A}{\|u_{\text{discr}}\|_A}, \tag{7.15}$$

which measures the discrepancy between the discrete solution u_{discr} and the computed (or truncated) solution u_{trunc}, which we solve for using an iterative method down to a relative tolerance that is driven by the discretization error (7.14). If the number of iterations until reaching that relative tolerance is n, then the solver's convergence rate is defined by

$$\varepsilon_{\text{trunc}}^{1/n}.$$

[3] In practice, we compute an approximation to u_{discr} by solving problem (7.13) with an excessive amount of iterations.

Table 7.2: Convergence rates, runtimes, and time-to-accuracy for HMG V-cycles (damped Jacobi (1,1)-smoother), which are used as an iterative solver for problem (7.13). Four discretization orders $k = 1, 2, 3, 4$ shown in four sets of rows, where each row differs by the mesh refinement level ℓ. HMG solver converges largely independent of level ℓ and order k. This results in improved time-to-accuracy for higher k due to increased accuracy per DOF.

Mesh		**DOF**	**Discr. error**	**Discr.**	**Trunc. error**	**HMG solver**		**Solve time**	**Time to**
k	l	[$\times 10^3$]	(energy norm)	**Conv. rate**	(energy norm)	**It.**	**Conv. rate**	[sec]	**accuracy**
1	2	.1	5.49e-1	–	3.89e-2	2	1.97e-1	0.00067	3.70e-4
⋮	3	.7	2.93e-1	0.91	1.78e-2	3	2.61e-1	0.00328	9.63e-4
	4	4.9	1.49e-1	0.98	5.92e-3	4	2.77e-1	0.02454	3.66e-3
	5	35.9	7.49e-2	0.99	6.27e-3	4	2.81e-1	0.19065	1.43e-2
	6	274.6	3.75e-2	1.00	1.80e-3	5	2.82e-1	1.8607	6.98e-2
	7	2146.7	1.87e-2	1.00	1.80e-3	5	2.83e-1	14.802	2.79e-1
2	1	.1	2.03e-1	–	1.36e-2	2	1.17e-1	0.00061	1.24e-4
⋮	2	.7	5.32e-2	1.93	1.55e-3	4	1.99e-1	0.00329	1.75e-4
	3	4.9	1.34e-2	1.99	1.16e-3	5	2.59e-1	0.02167	2.92e-4
	4	35.9	3.37e-3	2.00	1.24e-4	7	2.77e-1	0.21500	7.24e-4
	5	274.6	8.42e-4	2.00	3.91e-5	8	2.81e-1	1.9410	1.64e-3
	6	2146.7	2.11e-4	2.00	1.14e-5	9	2.82e-1	17.387	3.67e-3
3	1	.3	2.63e-2	–	4.75e-3	2	6.89e-2	0.00118	3.14e-5
⋮	2	2.2	3.38e-3	2.96	2.20e-4	5	1.86e-1	0.01553	5.26e-5
	3	15.6	4.26e-4	2.99	1.77e-5	8	2.55e-1	0.16650	7.09e-5
	4	117.6	5.33e-5	3.00	2.52e-6	10	2.76e-1	1.6232	8.66e-5
	5	912.7	6.66e-6	3.00	2.42e-7	12	2.81e-1	15.457	1.03e-4
4	1	.7	2.55e-3	–	9.10e-4	5	2.46e-1	0.00398	1.08e-5
⋮	2	4.9	1.63e-4	3.97	8.13e-5	8	3.08e-1	0.04315	7.87e-6
	3	35.9	1.03e-5	3.99	5.78e-6	11	3.34e-1	0.43819	5.16e-6
	4	274.6	6.40e-7	4.00	5.35e-7	13	3.29e-1	4.1636	3.47e-6
	5	2146.7	6.83e-8	3.23	4.29e-8	15	3.23e-1	38.362	3.09e-6

Finally, combining discretization (7.14) and truncation errors (7.15), we define the (total) error of the numerical solution by

$$\varepsilon_{\text{total}} \coloneqq \sqrt{\varepsilon_{\text{trunc}}^2 + \varepsilon_{\text{discr}}^2}\,. \tag{7.16}$$

The quantity of interest to evaluate the computational cost of a numerical solver is typically the *time to solution*. In the context of solvers for high-order discretizations, however, reporting only the runtime of the solver, t_{solve}, is not satisfactory since increasing order k delivers higher accuracy per DOF. Therefore, we choose to report a quantity that we regard as more suitable to describe computational cost, which we call the *time to accuracy* and define as follows:

$$t_{\text{solve}}\,\varepsilon_{\text{total}}.$$

The iterative solver in this section is designed to focus on the properties of HMG V-cycles. Thus, an iteration of the solver consists of applying one HMG V-cycle to the current residual, where on each level of the HMG hierarchy pre- and post-smoothing is performed via one iteration of damped Jacobi with damping set to 0.75. Cholesky factorization is used on the coarsest level in the hierarchy as the direct solver. The computations were performed on a single core of an Intel Westmere CPU (Xeon

E5620) and the results are summarized in Table 7.2. There, for each discretization order $k = 1, 2, 3, 4$ the mesh refinement level ℓ is varied. We first observe that the convergence rate of the discretization approximates the desired value of order k, which is in accordance with asymptotic finite element convergence results from the literature [46]. Further, comparing the discretization errors of elements with lower and higher orders confirms that the same amount of DOF yields smaller errors if k is higher, as mentioned above. Note that the mesh with $k = 4$, $\ell = 5$ does not attain the theoretical convergence rate of 4 because of the double precision arithmetic of our implementation together with Equation (7.14) for the discretization error. Namely, the roundoff error of the difference in (7.14) at double precision is $\sim 10^{-16}$, hence taking the square root yields $\sim 10^{-8}$ for the least possible value for $\varepsilon_{\text{discr}}$. Truncation errors are shown to be of the same orders of accuracy as discretization errors, which follows from our choice of relative tolerance to terminate the iterative solver. Since this tolerance is reduced as the mesh is refined and order k increases, the numbers of iterations (column *It.* in Table 7.2) are growing. The convergence rate demonstrates a fast and stable performance of the HMG solver throughout varying levels ℓ and orders k. This shows that the residual reduction per HMG V-cycle is largely independent of mesh refinement and discretization order, demonstrating the (nearly) ideal algorithmic convergence properties of HMG in the context of problem (7.13). The last column of Table 7.2, titled *Time to accuracy*, states the computational cost of the solver. It clearly shows the advantage of higher order k if combined with an effective solver as HMG. While for linear order, $k = 1$, the time to accuracy is decreasing as ℓ grows, this trend reduces for $k = 2, 3$ and eventually inverts for $k = 4$. This indicates that high-order discretizations paired with capable solvers are reducing the overall computational cost to approximate a solution with specified accuracy.

7.5 Robustness of HMG-based Stokes solver for mantle flow

The important physical parameter that determines the difficulty of the problem is the viscosity field (see Equation (4.3)). We extend our previous demonstration of Stokes solver robustness in Section 6.4. In this section, we generate a physically realistic representation of the mantle's viscosity using real Earth data. The viscosity varies over six orders of magnitude globally. However, what makes the poor conditioning of realistic mantle flow problems even more severe is the extremely thin layer in which this contrast develops. The viscosity drops by six orders of magnitude within a thin layer between two plates (the plate boundary, see Section 2.5). To assess the robustness of HMG and the Stokes solver as a whole, we generate plate boundaries down to a width of 5 km and a factor of 10^6 viscosity drop over 7 km as shown in Figure 7.5.

The robustness of our preconditioners for the Stokes system (4.6) is assessed in Table 7.3 by observing the number of GMRES iterations required for convergence, while decreasing the width of the plate boundaries (Figure 7.5, *bottom right images*). Our mesh refinement algorithm, which is based on the norm of viscosity gradients, locally refines the mesh to resolve the extreme viscosity variations. This results in an overall increase in the number of DOF. The third column in Table 7.3 demonstrates

the robustness of the solver for the (1,1) Stokes block, $\mathbf{Au} = \mathbf{f}$, therefore providing numerical evidence for the efficacy of our HMG preconditioner. The fourth column in Table 7.3 shows robustness of the complete Stokes solver with HMG for the (1,1) block and using HMG-based diag($\mathbf{A}$)-BFBT to approximate the inverse Schur complement (see Chapter 6). The GMRES iterations are seen to scale independently of the plate boundary width and thus viscosity gradient.

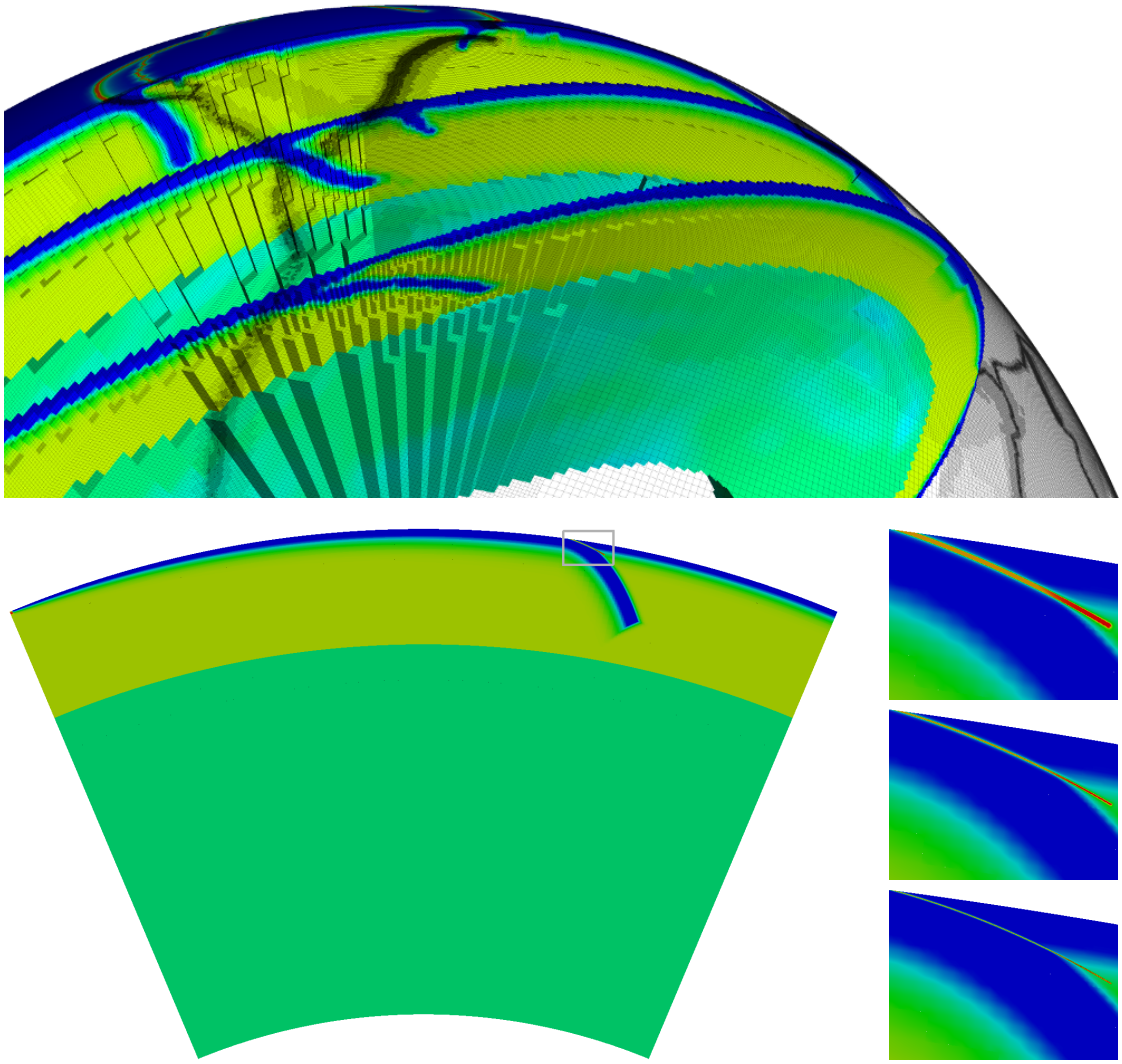


Figure 7.5: *Top image:* Viscosity field *(colors)* in cross sections of the mantle showing a subducting plate (i.e., high viscosity in *blue*). Local refinement of mesh elements resolves high gradients in viscosity. *Bottom images:* Model viscosity of a subducting plate on a thin cross section of Earth's spherical domain. The boundary width between subducting and overriding plate is decreased (15 km, 10 km, 5 km in *right three images*) to demonstrate solver robustness in Table 7.3.

Table 7.3: Robustness w.r.t. plate boundary width of HMG-preconditioned GMRES solver for the (1,1) block of Stokes, $\mathbf{Au} = \mathbf{f}$, and the linear (full) Stokes solver (diag($\mathbf{A}$)-BFBT Schur preconditioner, see Chapter 6). GMRES iterations to reduce the residual by 10^{-6} are reported.

Plate boundary width [km]	**DOF** [$\times 10^9$]	**GMRES iterations to solve $\mathbf{Au} = \mathbf{f}$**	**GMRES iterations to solve Stokes**
15	1.16	115	461
10	1.41	129	488
5	3.01	123	445

8

Computational Performance and Algorithmic & Parallel Scalability

After establishing the robustness of the Stokes solver with w-BFBT preconditioning in theory and numerically (Chapter 6) and introducing an effective and scalable multigrid method (Chapter 7), this chapter finally studies the scalability of the Stokes solver building on HMG+w-BFBT. One aspect of scalability is algorithmic scalability, i.e., the dependence of Newton and Krylov iterations on the mesh resolution and the discretization order. The second aspect is parallel scalability of the implementation, i.e., runtime measured on increasing numbers of compute cores. Studying both aspects is required to fully assess the performance of a solver at scale.

The cost of solving a nonlinear Earth mantle flow problem is dominated by the cost of the combined linear solves across Newton steps (see Section 5.1). The cost of a linear solve is determined by the number of MatVecs and HMG intergrid operations. MatVecs are encountered in the Krylov method and in the HMG smoothers. In all subsequently reported performance results, we use three Chebyshev-accelerated smoothing iterations for pre- and post-smoothing within the HMG V-cycle for both the viscous block $\tilde{\mathbf{A}}^{-1}$ (see Section 4.2) and in the Schur complement $\tilde{\mathbf{S}}^{-1}$ (see Section 6.1), which amounts to three V-cycles per application of the Stokes preconditioner. Therefore each Krylov iteration has the same cost and it is sufficient to compare the number of Krylov iterations. GMRES is used as the Krylov method throughout this chapter.

8.1 Algorithmic scalability

This section presents the algorithmic scalability of our linear and nonlinear Stokes solvers by means of increasing spatial resolution, which is performed by both refinement of mesh elements and increasing the polynomial order of the finite element discretization. Recall that we can study algorithmic scalability independently of parallel scalability, because the parallel distribution of a Stokes problem across cores does not alter the efficacy of the solver.

Our main goal is to achieve the best possible scalability for the whole Stokes solver, which combines

our multigrid and Schur complement preconditioners, but we also want to assess the scalability of HMG alone. Therefore we present results for sub-systems of the Stokes system by reporting iteration numbers for solving only for the velocity vector field $\mathbf{u}$ in $\mathbf{Au} = \mathbf{f}$ and for computing only the (scalar) pressure solution $\mathbf{p}$ in $\mathbf{Kp} = \mathbf{g}$, where $\mathbf{K} = \mathbf{K}_r = \left(\mathbf{BD}^{-1}\mathbf{B}^{\mathsf{T}}\right)$ denotes a pressure Poisson matrix as introduced in Section 6.1. Studying these individual components allows us to observe HMG performance in isolation and to compare it to the scalability of the full Stokes system, which is indicative of the quality of the BFBT Schur complement approximation.

Algorithmic scalability of HMG+w-BFBT linear Stokes preconditioners

We consider the model problem with multiple randomly distributed sinkers from Section 6.2 and the w-BFBT approximation (6.6) for the inverse Schur complement. The algorithmic scalability in Table 8.1 shows results for the Stokes solver as well as its individual components $\mathbf{Au} = \mathbf{f}$ and $\mathbf{Kp} = \mathbf{g}$. All systems are solved with preconditioned GMRES down to a relative tolerance of 10^{-6}. The preconditioners for $\mathbf{A}$ and $\mathbf{K}$ are HMG-V-cycles as described in Section 7.2. For the w-BFBT preconditioner, we set a constant left boundary amplification $a_l = 1$ and vary the right boundary amplification a_r according to results from Section 6.5. The iteration counts in Table 8.1a show textbook mesh independence when increasing the level of refinement ℓ of the uniform mesh. This holds for each component, $\mathbf{A}$ and $\mathbf{K}$, and also the whole Stokes solver. Hence we conclude that the approximation of the of inverses of $\mathbf{A}$ and $\mathbf{K}$ via HMG and the Schur complement approximation by w-BFBT are mesh-independent. When the discretization order k is increased, the iteration counts presented in Table 8.1b increase mildly. The convergence of both components $\mathbf{A}$ and $\mathbf{K}$ exhibits a moderate dependence on k. Since the increase in number of iterations is sightly larger for the full Stokes solve than for $\mathbf{A}$ and $\mathbf{K}$, we suspect a mild deterioration of w-BFBT as a Schur complement approximation.

Algorithmic scalability for nonlinear mantle flow

To study algorithmic scalability in a realistic mantle flow setting, we consider a nonlinear problem with one subducting slab as in Section 7.5, discretize velocity and pressure fields with $\mathbb{Q}_2 \times \mathbb{P}_1^{\text{disc}}$ finite elements, and use the diag($\mathbf{A}$)-BFBT approximation for the inverse Schur complement (see Section 6.1). The plate boundary region between the subducting plate and the overriding plate has a width of 5 km. We refine the mesh locally in the regions of highest viscosity variations by tightening the refinement criterion, which is based on the viscosity gradient. Thus the total number of degrees of freedom grows slowly, though significantly greater resolution is obtained in these regions. The required numbers of linear and nonlinear iterations are shown in Table 8.2, where the cost of the nonlinear solver is measured by the total number of GMRES iterations across nonlinear iterations. The linear solver with diag($\mathbf{A}$)-BFBT Schur preconditioner requires a number of iterations that is largely independent of the resolution. In the numerical experiments for this cross-sectional model problem, we employed a standard Newton linearization. For mantle flow problems of global scale, the

Table 8.1: Algorithmic scalability of Stokes solver with HMG+w-BFBT preconditioning while (a) varying mesh refinement level ℓ and (b) varying discretization order k (problem S16-rand, DR(μ) = 10^6 from Section 6.2). Computational cost is expressed in number of GMRES iterations (abbreviated by *It.*) for 10^{-6} residual reduction. Left boundary amplification for $\mathbf{C}_{w_l}$ is fixed to $a_l = 1$; right boundary amplification a_r for $\mathbf{D}_{w_r}$ varies. Additionally, the numbers of GMRES iterations for solving only the sub-systems $\mathbf{Au} = \mathbf{f}$ and $\mathbf{Kp} = \mathbf{g}$ are given for demonstration of HMG efficacy (here, $\mathbf{f}$ is the right-hand side of the momentum equation and $\mathbf{g}$ is the discreet representation of $\nabla \cdot \boldsymbol{f}$; however, random right-hand sides would give similar convergence results).

(a) Algorithmic scalability (fixed order $k = 2$)

ℓ	$\boldsymbol{u}$-DOF [$\times 10^6$]	It. **A**	p-DOF [$\times 10^6$]	It. **K**	DOF [$\times 10^6$]	It. **Stokes**
4	0.11	18	0.02	8	0.12	40
5	0.82	18	0.13	7	0.95	33
6	6.44	18	1.05	6	7.49	33
7	50.92	18	8.39	6	59.31	34
8	405.02	18	67.11	6	472.12	34
9	3230.67	18	536.87	6	3767.54	34
10	25807.57	18	4294.97	6	30102.53	34

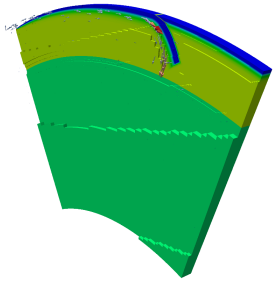
(b) Algorithmic scalability (fixed level $\ell = 5$)

k	$\boldsymbol{u}$-DOF [$\times 10^6$]	It. **A**	p-DOF [$\times 10^6$]	It. **K**	DOF [$\times 10^6$]	It. **Stokes**
2	0.82	18	0.13	7	0.95	33
3	2.74	20	0.32	8	3.07	37
4	6.44	20	0.66	7	7.10	36
5	12.52	23	1.15	12	13.67	43
6	21.56	23	1.84	12	23.40	50
7	34.17	22	2.75	10	36.92	54
8	50.92	22	3.93	10	54.86	67

nonlinearities of the mantle's rheology become significantly more challenging mainly due to plastic yielding. In this case, the perturbed Newton linearization proposed in Chapter 5 enables global mantle convection simulations at a feasible computational cost.

We have demonstrated how the combination of our preconditioner and linear and nonlinear solvers yields an implicit method whose number of iterations scales (largely) independent of model fidelity. Here, fidelity is understood as the resolution of the mesh with finite element discretization and the size of the smallest-scale features, which are the plate boundary regions. This results in an algorithmically optimal method, despite the severely nonlinear rheology, high viscosity gradients, effective anisotropy, and large heterogeneities. Moreover, the cost of the solver is reduced by adaptive mesh refinement,

Table 8.2: Algorithmic scalability of inexact Newton–Krylov method for solving a nonlinear mantle flow problem with one subducting slab and 5 km plate boundary. Simulation cost expressed in total number of GMRES iterations is largely independent of the maximal resolution of the adaptive mesh (10^{-7} Newton residual reduction used as stopping criterion). A two times higher resolution increases the DOF of the adaptively refined mesh only by about a factor of 2–3. In contrast, the factor would be eight with uniform refinement.



Max level of refinement	**Finest resolution** [m]	**DOF** [$\times 10^6$]	**Newton iterations**	**GMRES iterations**
10	2443	0.96	14	1408
11	1222	2.67	18	1160
12	611	5.58	21	1185
13	305	11.82	21	1368
14	153	36.35	27	1527

Table 8.3: Texas Advanced Computing Center's supercomputers.

System name	**Compute node**	**Nodes**	**Cores**	**Peak** [PFlops/s]
Stampede (CPU)	2x Intel Xeon (Sandy Bridge) 8-core CPU	6400	102,400	2+
Lonestar 5	2x Intel Xeon (Haswell) 12-core CPU	1252	30,048	1+
Stampede 2 (KNL)	1x Intel Xeon Phi (Knights Landing), 68 cores	4200	285,600	18
Stampede 2 (SKX)	2x Intel Xeon (Skylake) 24-core CPU	1736	83,328	N/A

which reduces the number of DOF—in this case by four orders of magnitude, from the $\mathcal{O}(10^{13})$ needed for a uniform mesh of Earth's mantle, to about $\mathcal{O}(10^{9})$ required here using aggressive refinement. Further reducing the number of DOF are the third-order accurate finite elements $\mathbb{Q}_2$ employed here, along with a mass-conserving discretization $\mathbb{P}_1^{\text{disc}}$. The algorithmic scalability and the greatly-reduced number of DOF exhibited by our solver are critical for the overall goal of reducing time-to-solution (for a given accuracy). The remaining component is parallel scalability, which we study next.

8.2 Parallel systems and architectures

The systems and architectures used in this dissertation for large-scale simulations and to assess parallel scalability are Intel-based and IBM BlueGene-based. The supercomputers using Intel processors and accelerators are (or were) housed at the Texas Advanced Computing Center (TACC) and are summarized in Table 8.3.

The now decommissioned Stampede multi peta-scale supercomputer began production in 2013 as a 6400+ node cluster of Dell PowerEdge nodes. Each node contained two Intel Sandy Bridge 8-core CPUs (Xeon E5-2680), 32 GBytes of main memory, and the coprocessor (or accelerator) Intel Xeon Phi (Knights Corner) with 61 cores. The FDR InfiniBand interconnect provided 56 GBytes/s of bandwidth between nodes. The aggregate peak performance of the CPUs was 2+ PFlops/s, while the coprocessors delivered an additional aggregate peak performance of 7+ PFlops/s. We utilized only the CPUs of Stampede in our runs.

The Lonestar 5 supercomputer entered production in January 2016 and is a Cray XC40 system consisting of 1252 compute nodes. Each node is equipped with two Intel Haswell 12-core processors (Xeon E5-2680v3) and 64 GBytes of memory. Inter-node communication is based on an Aries Dragonfly topology network that provides dynamic routing and thus enables optimal use of the system bandwidth.

Currently the newest supercomputer at TACC, Stampede 2, started full production in Fall 2017 with 18 PFlops/s peak performance. Phase 1 of Stampede 2 consists of 4200 nodes, each of which is equipped with Intel's second generation Xeon Phi (Knights Landing) architecture and a total of 112 GBytes of memory per node. Knights Landing (KNL) is a stand-alone processor with 68 cores and supports up to 4 H/W threads per core. The interconnect is a 100 GBytes/s Intel Omni-Path network with a fat tree topology. In Phase 2, additional 1736 Intel Skylake CPU nodes were added. Each Skylake (SKX) node contains two sockets with Intel Skylake (Xeon Platinum 8160) 24-core processors and 192 GBytes of memory.

Table 8.4: IBM BlueGene/Q supercomputers. Each rack has 1024 nodes with 16-core CPUs.

System name	**Racks**	**Cores**	**H/W threads**	**Peak** [PFlops/s]
AMOS	5	81,920	327,680	1.0
Vulcan	24	393,216	1,572,864	5.0
JUQUEEN	28	458,752	1,835,008	5.8
Sequoia	96	1,572,864	6,291,456	20.1

In addition to Intel-based systems at TACC, we used IBM BlueGene/Q (BG/Q) supercomputers [87] that are summarized in Table 8.4 to analyze performance in detail, carry out numerous optimizations (see Section 7.3), and demonstrate scalability to extreme scales. The smaller systems were used for testing, optimization, scaling, and science runs. The largest runs have been performed on the Sequoia supercomputer at the Lawrence Livermore National Laboratory (LLNL). Sequoia consists of 96 IBM Blue Gene/Q racks, reaching a theoretical peak performance of 20.1 PFlops/s. Each rack consists of 1024 compute nodes, which host an 18 core A2 chip that runs at 1.6 GHz. Of these 18 cores, 16 are devoted to computation, one for the lightweight O/S kernel, and one for redundancy. Every core supports 4 H/W threads, thus, in total Sequoia has 1,572,864 cores and can support up to 6,291,456 H/W threads. The total available system memory is 1.458 PBytes. BG/Q nodes are connected by a five-dimensional (5D) bidirectional network, with a network bandwidth of 2 GBytes/s for sending and receiving data. Each BG/Q rack features dedicated I/O nodes with 4 GBytes/s I/O bandwidth. The system implements optimized collective communication and allows specialized tuning of point-to-point communication. We obtained all timing and performance measurements by means of the IBM HPC Toolkit for BG/Q. The toolkit retrieves performance information about the processor, memory hierarchy, and interconnect. All runs involved double-precision arithmetic and the code is compiled using the IBM XL C compilers for BG/Q, version 12.1.10.

8.3 Parallel scalability on Intel-based systems

We present parallel weak and strong scalability of our linear Stokes solver, where we measure the runtime of GMRES with HMG preconditioning for the viscous block and within w-BFBT as well as diag($\mathbf{A}$)-BFBT preconditioners for the Schur complement. Since applying the preconditioners consumes the bulk of overall nonlinear solver runtime, the results here indicate the scalability of the full nonlinear Stokes solver. Note that the methods w-BFBT and diag($\mathbf{A}$)-BFBT for Schur complement preconditioning are comparable in terms of parallel scalability, because they exhibit the same computational complexity per GMRES iteration and our parallel scalability runs report time per GMRES iteration.

This section's results were obtained on the Intel-based clusters, Stampede, Lonestar 5, and Stampede 2, which are described in Section 8.2. Whereas in the following section, we utilized IBM's BlueGene/Q architecture.

Weak and strong scalability on Stampede for global mantle flow

The parallel scalability on Stampede is performed for a realistic Earth mantle convection problem at global scale, where we employ the $\mathbb{Q}_2 \times \mathbb{P}_1^{\text{disc}}$ velocity–pressure pairing of finite elements and the diag($\mathbf{A}$)-BFBT approximation for the inverse Schur complement. We refine the mesh locally in the regions of highest viscosity variations resulting in up to six refinement levels difference.

For the weak scalability in Figure 8.1a, we maintain the DOF/core at ∼0.2 millions over a 128-fold increase in cores. The *blue curve* in the figure shows the computational speedup of the Stokes solver in degrees of freedom processed per runtime of one preconditioned GMRES iteration. The solver achieves 88% parallel efficiency at the highest core count of 16,384. The setup of the Stokes solver is performed once before the solve and is dominated by the generation of the HMG hierarchies for the viscous block and the pressure Poisson operators within BFBT. Its weak scalability *(green curve)* documents the computational speedup in degrees of freedom per seconds of setup runtime and reaches over 50% parallel efficiency at 4096 cores, but then reduces to 24% efficiency. Note that for the overall time to solution, the contribution of the solve is significantly larger than setup runtime.

Figure 8.1b demonstrates strong scalability on Stampede for a Stokes solve. Here, the overall number of degrees of freedom stays fixed at 97 million and the *curve* in the figure represents the speedup in baseline runtime (128 cores) over runtime. The parallel efficiency is gradually decreasing as the core count grows, which is expected for an implicit solver due to growing point-to-point communication, and reaches 52% at the largest run.

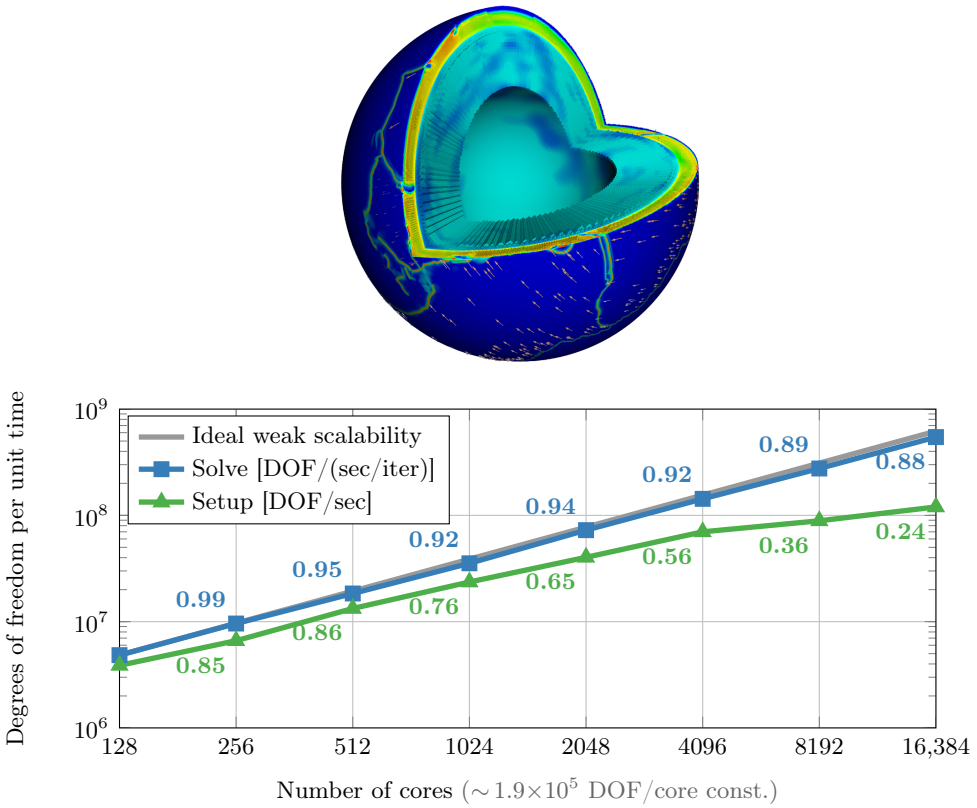
Finally, note that these results were obtained at an earlier stage of the solver's implementation, when not all of the optimizations described in Section 7.3 were completed.

Weak and strong scalability on Lonestar 5 for sinker model problem

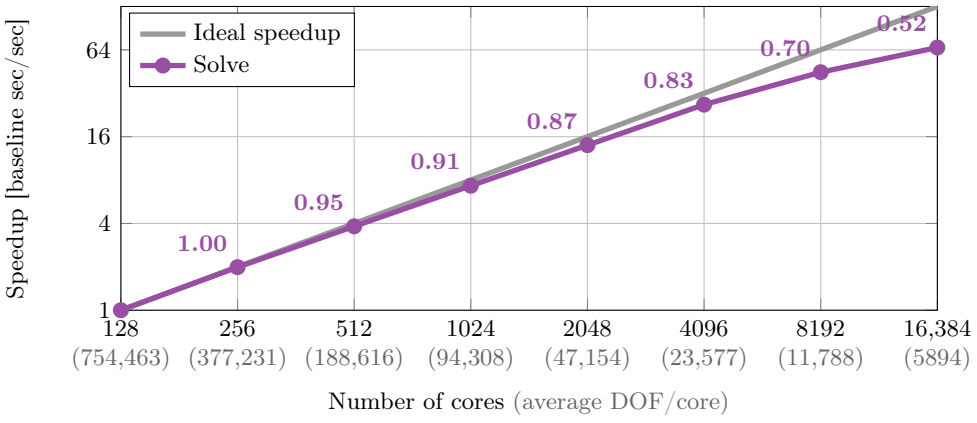
We switch to the sinker model problem from Section 6.2 (S16-rand, DR(μ) = 10^6 as in Table 8.1a) and the w-BFBT Schur preconditioner. The discretization with the $\mathbb{Q}_2 \times \mathbb{P}_1^{\text{disc}}$ finite element pairing is carried out on a uniform mesh. Results in Figure 8.2a for weak scalability on the full Lonestar 5 peta-scale system, where we fixed the DOF/core to ∼1 million, show that the Stokes solver with w-BFBT *(blue curve)* maintains 90% parallel efficiency over a 618-fold increase in degrees of freedom along with cores. Even for the setup of the Stokes solver *(green curve)*, which mainly involves generation of the HMG hierarchy, we observe 71% parallel efficiency. These are excellent results for such a complex implicit multilevel solver with optimal algorithmic performance (when the mesh is refined, or nearly algorithmically optimal when the order is increased) and with convergence that is independent of the number of cores.

Figure 8.2b reports strong scalability results (overall DOF fixed to 59 million) and how the number of OpenMP (OMP) threads substituting MPI ranks influences speedup.[1] Over the 78-fold increase from 48 to 3744 cores, efficiency reduces moderately, to a worst-case 68% for 24×OMP1. However,

[1] Even though the processors of Lonestar 5 support two threads per physical core (Intel Hyper-Threading Technology), assigning more than one OpenMP thread per core did not improve performance.



(a) Weak scalability



(b) Strong scalability

Figure 8.1: Parallel scalability on **Stampede** running our Stokes solver with HMG+diag(**A**)-BFBT preconditioning for a global mantle convection problem. (a) Weak scalability of setup *(green)* and solve phases *(blue)*; solve speed is normalized w.r.t. deviations from const. DOF/core. Ideal weak scalability for solve is depicted by the *gray line*. *Numbers along the graph lines* indicate weak parallel efficiency w.r.t. ideal weak scalability (efficiency baseline is 128 cores result). The largest problem size on 16,384 cores has 3 billion DOF. (b) Strong scalability of solve phase; *numbers along the graph lines* indicate strong efficiency w.r.t. ideal speedup (baseline is 128 cores result).

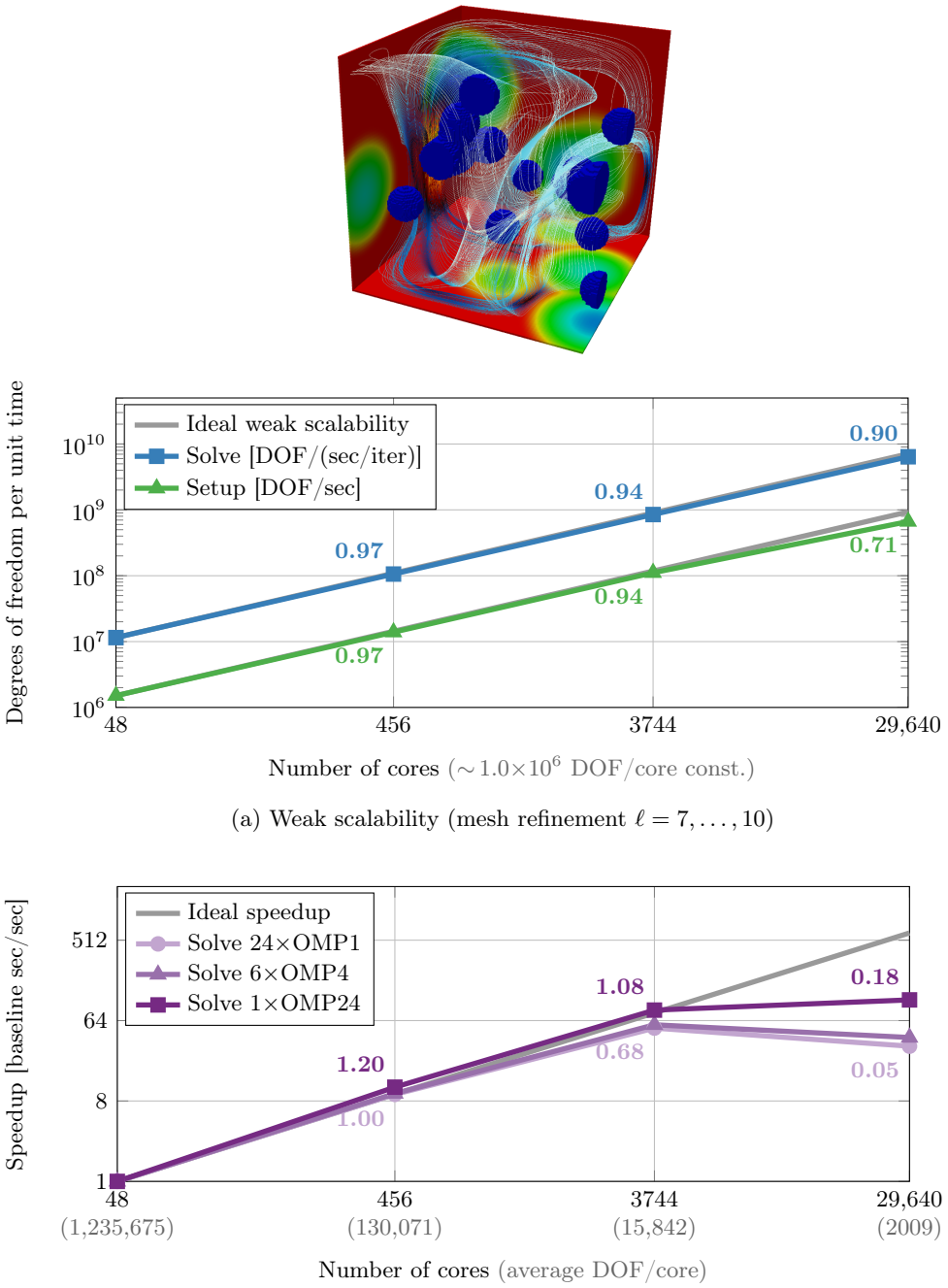
note that in the largest run with 29,640 cores, the granularity is only ∼2000 DOF/core, which is extremely challenging for strong scalability. In this case, due to the increased communication volume, overlapping with decreased amounts of computation becomes impossible and communication dominates the runtime. This behavior is expected for an implicit solver, especially for a multilevel method that does not sacrifice algorithmic optimality for parallel scalability.

Weak and strong scalability on Stampede 2 for sinker model problem

The Stampede 2 supercomputer is composed of 68-core Intel Xeon Phi (KNL) nodes and dual 24-core Intel Xeon CPU (SKX) nodes. We report parallel weak and strong scalability for each node type in Figure 8.3, where the *left two axes* correspond to KNL and the *right two axes* correspond to SKX. As before, we carry out simulations of the sinker model problem from Section 6.2, use uniform meshes with the $\mathbb{Q}_2 \times \mathbb{P}_1^{\text{disc}}$ finite element pairing, and w-BFBT for the inverse Schur complement approximation.

Weak scalability results for the Stokes solve and setup are shown as *blue* and *green curves* in Figure 8.3a over a 512-fold increase in cores along with degrees of freedom. The number of DOF/core differs due to a larger memory capacity on SKX nodes, allowing for ∼1.2 million DOF per SKX core and about one third less DOF per KNL core. The parallel efficiency on Stampede 2's CPU nodes is comparable with Lonestar 5, demonstrating remarkable 90% at peak core count of 24,576. The KNL nodes are more difficult to scale and reach just under 70%. A comparison of the solver's performance on KNL and SKX nodes shows that execution time on one SKX node is about similar to two KNL nodes.

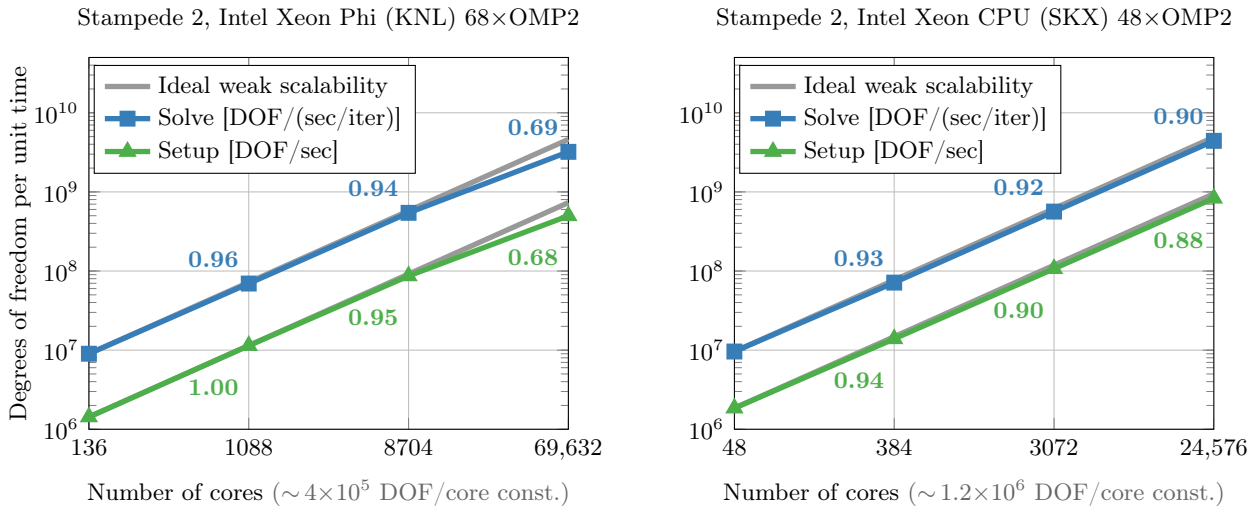
Strong scalability in Figure 8.3b reveals a relatively high 28% parallel efficiency on 24,576 SKX cores, which is better than ∼8% strong efficiency on KNL over the same 512-fold increase in cores and also better than on Lonestar 5 (see Figure 8.2b). We tested the effect of substituting OpenMP (OMP) threads for MPI ranks on KNL nodes and see in Figure 8.3b, *left* that speedup remains largely the same.



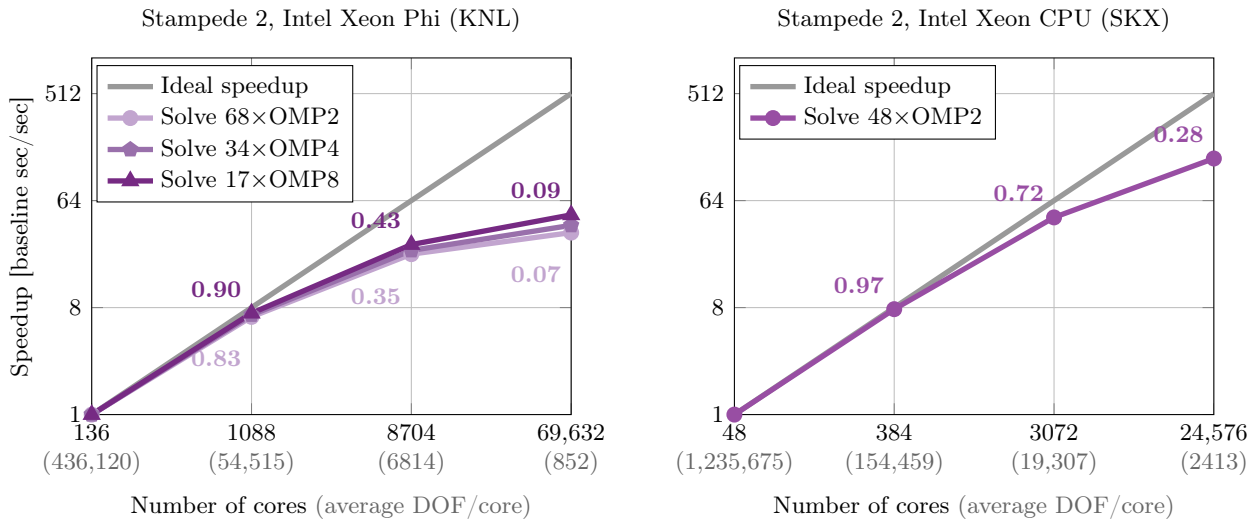
(a) Weak scalability (mesh refinement $\ell = 7, \ldots, 10$)

(b) Strong scalability (mesh refinement $\ell = 7$)

Figure 8.2: Parallel scalability on **Lonestar 5** running our Stokes solver with HMG+w-BFBT preconditioning (sinker problem S16-rand, DR(μ) = 10^6 as in Table 8.1a). (a) Weak scalability of setup and solve phases (normalized w.r.t. deviations from const. DOF/core). *Numbers along the graph lines* indicate weak parallel efficiency w.r.t. ideal weak scalability (efficiency baseline is 48 cores result). The largest problem size on 29,640 cores has 30 billion DOF. (b) Strong scalability of solve phase for different configurations of OpenMP threads (OMP) substituting MPI ranks on each node consisting of 24 cores. *Numbers along the graph lines* indicate strong efficiency w.r.t. ideal speedup (baseline is 48 cores result).



(a) Weak scalability (mesh refinement $\ell = 7, \ldots, 10$)



(b) Strong scalability (mesh refinement $\ell = 7$)

Figure 8.3: Parallel scalability on **Stampede 2** nodes with Intel Xeon Phi (KNL) on *left* and with Intel Xeon CPU (SKX) on *right* running our Stokes solver with HMG+w-BFBT preconditioning (sinker problem S16-rand, $\mathrm{DR}(\mu) = 10^6$ as in Table 8.1a). (a) Weak scalability of setup and solve phases (normalized w.r.t. deviations from const. DOF/core); *numbers along the graph lines* indicate weak parallel efficiency w.r.t. ideal weak scalability. (b) Strong scalability of solve phase; *numbers along the graph lines* indicate strong efficiency w.r.t. ideal speedup.

8.4 Parallel scalability and performance on IBM BG/Q systems

This section presents extreme scalability and performance results utilizing IBM's BlueGene/Q architecture. We achieve 97% weak parallel efficiency over a 96-fold core increase up to 1.6 million cores for the solve phase of the HMG+diag($\mathbf{A}$)-BFBT preconditioned Stokes solver with the additional difficulty of handling highly adapted meshes for realistic simulations of global-scale mantle flow with plates.

The important physical parameter that determines the difficulty of the problem is the viscosity field. In our subsequent performance analysis, we use real Earth data to generate a physically realistic representation of viscosity. The viscosity varies over six orders of magnitude globally. However, what makes realistic mantle flow problems even more poorly conditioned as well as nonlinear is the extremely thin layer in which this contrast develops. The viscosity drops by six orders of magnitude within a thin layer between two plates (the plate boundary).

To assess our solver's weak and strong scalability, the 10^6 factor viscosity drop occurs within just 3 km. Since tectonic plates (the largest surface structures) are 2000–14,000 km across, and Earth's circumference is 40,075 km, this results in a very wide range of length scales of interest. To capture the viscosity variation, the mesh is refined to $\sim$75 m local resolution in our largest simulations, resulting in a mesh with 9 levels of local refinement. For all performance results, we use a velocity discretization with polynomial order $k = 2$.

Weak and strong scalability for global mantle flow

We present weak and strong scalability results on the Vulcan and Sequoia BG/Q supercomputers from 1 rack with 16,384 cores up to 96 racks with 1,572,864 cores (see Section 8.2). Scalability measurements corresponding to 1, 2, and 4 racks were obtained on Vulcan, whereas the remaining runs on 8–96 racks were performed on Sequoia.

The cost of our large-scale nonlinear mantle convection simulations is overwhelmingly dominated by the cost of the GMRES iterations during a linear Stokes solve. These GMRES iterations include one HMG V-cycle for the (1,1) Stokes block and two V-cycles in the Schur complement approximation, as explained earlier. For the extreme-scale runs on Sequoia, we had limited access to the system, which allowed us to run 10 representative GMRES iterations. However, we extrapolate that the influence of I/O and setup costs are marginal compared to the cost of the combined GMRES iterations that are required for a nonlinear mantle convection simulation at global scale. Moreover, if multiple linear Stokes solves are performed in a sequence of Newton steps, then the setup cost drops further, because we reuse the part of HMG hierarchy data associated with a non-changing mesh and only re-discretize differential operators on each level of the hierarchy. Finally, note that in our observations the setup time for HMG is largely bounded independent of the number of cores for a constant problem size.

The main result is the weak scalability shown in Figure 8.4. The solver maintained 97% parallel efficiency *(blue curve)* over a 96-fold increase in problem size, from 16K to 1.6M cores of the full

Sequoia system. The largest problem involved 602 billion DOF. The I/O for writing output data has to be performed only once at the end of a nonlinear solve. The problem sizes used in the weak scalability runs would produce ∼8.5 GBytes of output per BG/Q I/O node. With an I/O bandwidth of 4 GBytes/s we can also consider the writing of the output to be negligible for overall runtime (note that we did not output solution fields, since the full nonlinear simulation could not be run to completion due to limited access). The negligible time for I/O and problem setup stem from the advantages of adaptive implicit solvers: adaptivity results in the problem itself being generated online as part of the solver; implicit means that fewer outputs/checkpoints would be required.

In Figure 8.5, we show strong scalability results for a mantle convection simulation with 8.3 billion DOF. Starting from one rack with 16,384 cores (granularity of 506K DOF/core), we achieve a 32-fold speedup on 96 racks with 1,572,864 cores (granularity of 5K DOF/core), indicating 33% solver efficiency in strong scalability.

Contrary to conventional wisdom, this work shows that algorithmically optimal implicit finite element solvers for highly nonlinear, severely poorly conditioned, heterogeneous, indefinite PDEs can be designed to scale to $\mathcal{O}(10^6)$ cores.

Node performance analysis

The performance results on BG/Q compute nodes further support our scalability results. The top pie charts of Figure 8.6 decompose the overall runtime into the largest contributors. We can observe that the (highly optimized) matrix-free apply routines dominate with 80.6% in the 1 rack case. Furthermore, their portion remains very stable with 78% on 96 racks. This result demonstrates a key component of a highly scalable, parallel multigrid implementation. The percent runtime for intergrid transfer operations is low compared to MatVecs and stays low even at 1.6 million cores. Hence, we have achieved a balance between MatVecs and intergrid operations that results in nearly optimal scalability.

MatVecs represent the portion of the code where the maximal performance in terms of flops can be achieved. With their dominance in runtime we are able to increase total performance close to its maximum. That way our implementation is performing at the limits of the roofline model as predicted in Figure 7.4b.

MPI communication analysis

Figure 8.7 summarizes MPI communication time measured during weak and strong scalability runs: tasks with minimum, median, and maximum communication time are displayed. Indeed, for weak scalability, we clearly observe that percentage of time spent in MPI communication remains nearly constant relative to runtime (Figure 8.7a). This contributes to the nearly perfect scalability results presented in Figure 8.4. The increase in median and maximum communication time in the 64 racks case can be justified by the lack of 5D torus connectivity in that particular configuration (due to specific job partitioning). Another reason can be found in a more aggressive repartitioning of coarser multigrid levels, which leaves a greater amount of cores idle during a short period of time in the

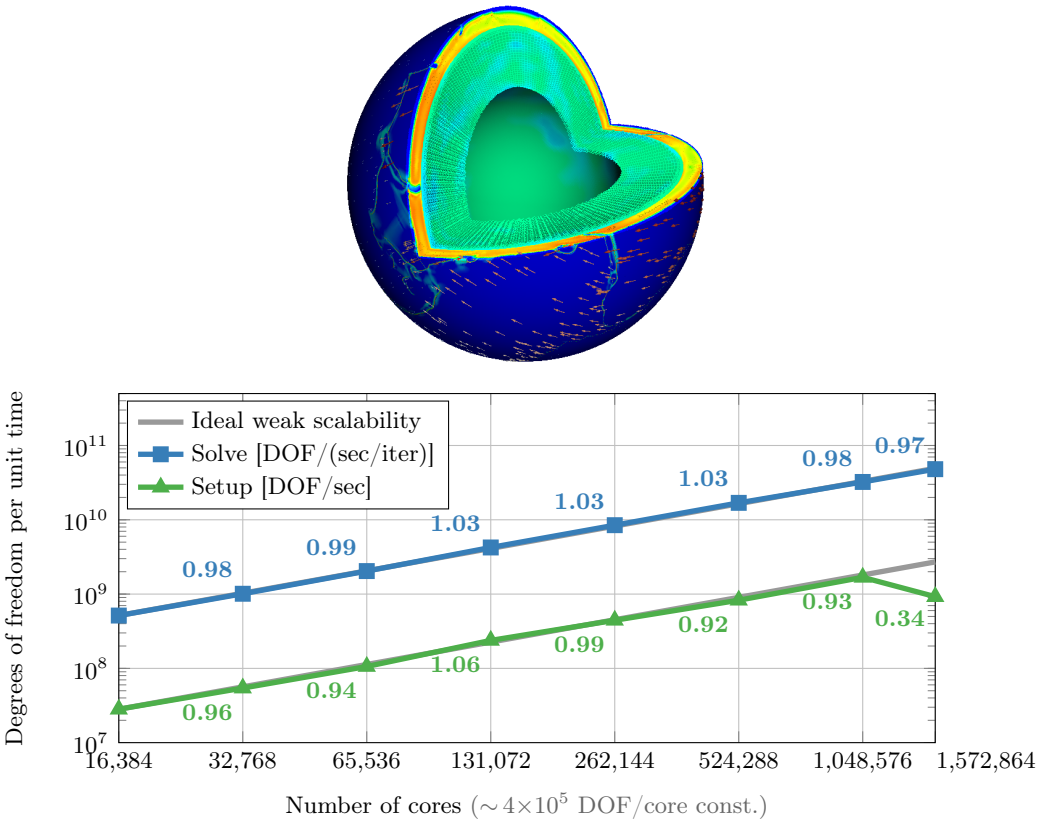


Figure 8.4: Weak scalability results on Vulcan and Sequoia from 1 to 96 racks. Performance is normalized by time and number of GMRES iterations. *Numbers along the graph lines* indicate efficiency w.r.t. ideal speedup (efficiency baseline is the 1 rack result). We report both the weak scalability of the linear solver's iterations *(blue)* and the one-time setup cost of the linear solver *(green)*. The largest problem size on 96 racks has 602 billion DOF.

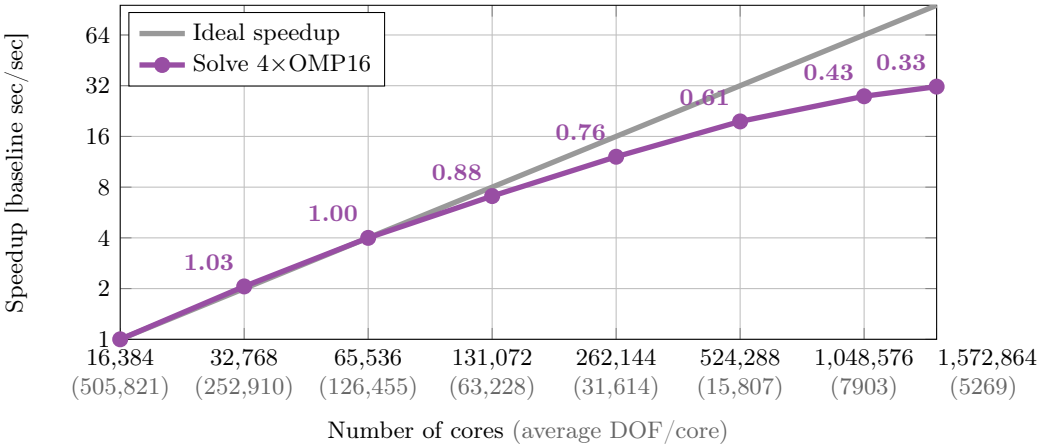


Figure 8.5: Strong scalability results on Vulcan and Sequoia from 1 to 96 racks. *Numbers along the graph lines* indicate efficiency with respect to ideal speedup (baseline is the 1 rack result). We report the strong scalability of the linear solver *(purple curve)*.

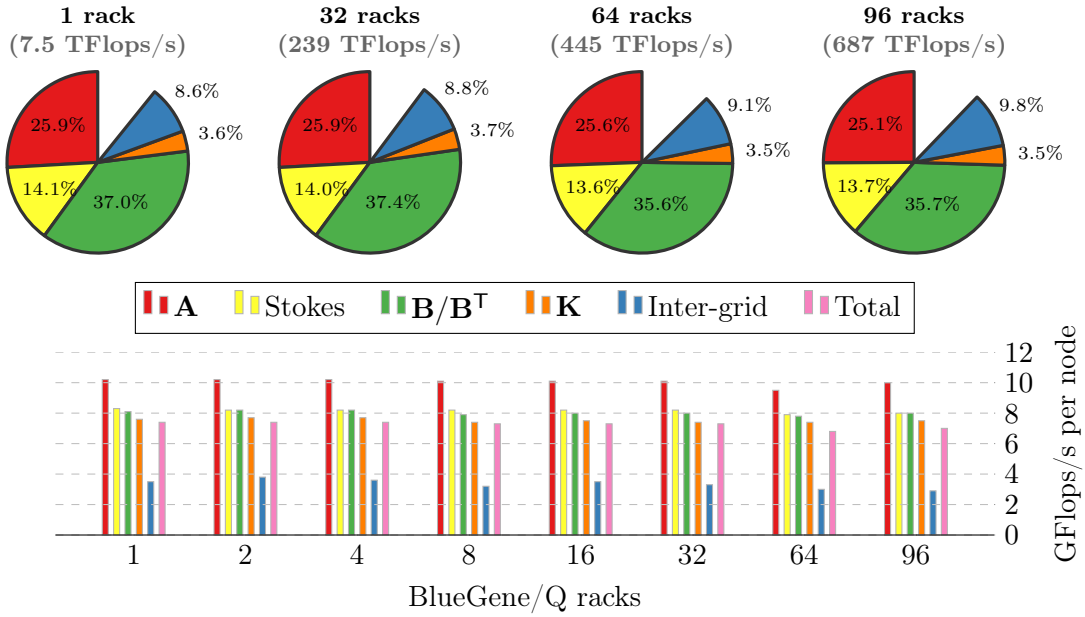


Figure 8.6: Analysis of MatVecs and intergrid operators within the Stokes solves of the weak scalability runs on Vulcan and Sequoia. Pie charts show fraction of time in each routine, while the histograms show corresponding average GFlops/s per BG/Q node. The symbols denote MatVecs for viscous stress $\mathbf{A}$, continuous, nodal Poisson operator $\mathbf{K}$, and divergence/gradient $\mathbf{B}/\mathbf{B}^{\mathsf{T}}$.The empty slices in the pie charts consist of all other routines with generally low GFlops/s per node (e.g., GMRES orthogonalization, null space projections).

V-cycle. This is suggested by the higher percentage of MPI_Waitall time on 64 racks in Figure 8.8. However, this does not need to affect scalability in a negative way since fewer cores may perform the same task quicker because of higher granularity of DOF.

For the strong scalability runs, we observe a gradual increase of relative MPI communication time (Figure 8.7b), as is expected for implicit solvers. Note that the increase begins only at 4 racks. Communication time exceeds 50% of overall runtime only at about 1 million cores. At its maximum, communication time is still below 30%.

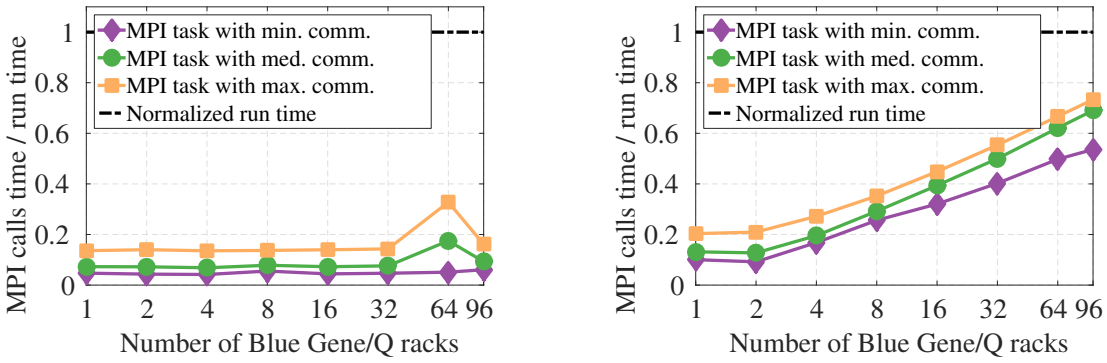


Figure 8.7: MPI communication time relative to total runtime for (a) weak scalability and (b) strong scalability on Vulcan and Sequoia.

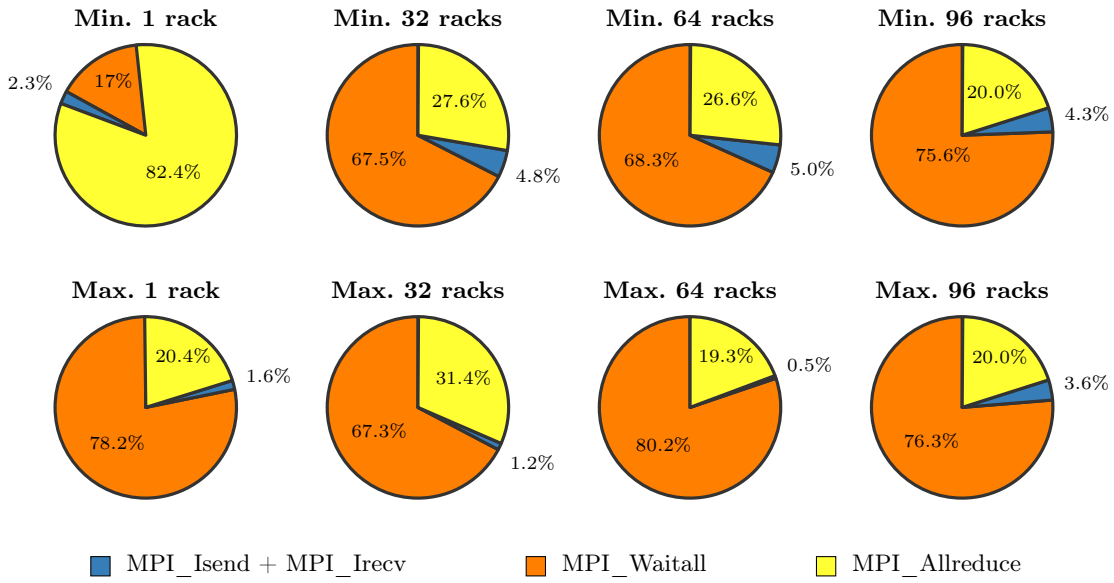


Figure 8.8: MPI routine communication time for the weak scalability runs on Vulcan and Sequoia supercomputers.

9

Conclusions

9.1 Mathematical and computational contributions

Even though the properties of the Stokes equations and their numerical solution have been studied over many years, simulating Earth's mantle convection at global scale while resolving thin plate boundaries has remained sufficiently challenging that computational solutions were intractable. The challenges are due to the mantle's complex constitutive relationship, which is characterized by severe nonlinearities (due to shear thinning and plastic yielding), six orders of magnitude viscosity variations modeling plate boundary regions, and a wide range of spatial scales (from narrow trenches to continent-sized plates). Accommodating the required DOF for a globally uniform mesh at the resolution demanded by small-scale features has been infeasible even on today's largest supercomputers. Major advances in parallel adaptive mesh refinement using octrees and space-filling curves have made high-fidelity discretization of the mantle feasible with the current generation of supercomputers. However, this was only one step toward the goal of realistic mantle convection simulations. Even after the efficiency improvements due to adaptivity, the resulting algebraic systems demanded large-scale parallel resources. Further, the severe nonlinearities and heterogeneities of mantle flow problems lead to extremely poor conditioning, resulting in prohibitive computational costs in the past for accurate solutions. With the advances introduced here in numerical methods, parallel algorithms, and computationally efficient implementation in the areas of multigrid methods, Schur complement approximation, inexact Newton–Krylov methods, and nonlinear preconditioning, we have made significant strides toward the goal of realistic mantle convection simulation. We achieved fast convergence, robustness with respect to highly heterogeneous coefficients and severe nonlinearities, and excellent parallel scalability, which now enables us to study the physics of mantle flow in detail and to constrain parameters in the constitutive relationship.

We demonstrate in this dissertation how successive barriers of intractability were overcome: *(i)* p4est for parallel AMR (DOF reduction, scalable mesh generation), *(ii)* HMG for parallel multigrid (algorithmic optimality, scalability of setup and solve), *(iii)* HMG-based weighted BFBT for Schur complement preconditioning (robustness for heterogeneous viscosities), and *(iv)* inexact Newton–

Krylov with nonlinear preconditioning (fast and robust nonlinear solve). Our advances are driven by mantle convection problems, but the methods we develop are applicable across a variety of problems since heterogeneity, anisotropy, multiple scales, and nonlinearities analogous to those stemming from the mantle's rheology are prevalent in many applications.

Our inexact Newton–Krylov nonlinear solver uses grid continuation to resolve viscosity variations caused by the nonlinear rheology and employs the H^{-1}-norm for backtracking line search. The perturbations that we introduced to the Newton linearization result in better Newton step directions in the presence of plastic yielding. This significantly improves nonlinear convergence with effectively no additional computational cost.

Schur complement preconditioning is critical for Stokes systems with highly heterogeneous viscosities. We developed weighted BFBT approximations of the inverse Schur complement that exhibit robustness for up to ten orders of magnitude viscosity contrast. In our analysis of BFBT methods, spectral equivalence is derived theoretically and supported by numerical computation of eigenvalues. Additionally, detailed numerical experiments document viscosity-variation robustness over a wide range of benchmark problems and optimal or near optimal algorithmic scalability is reported when mesh elements are refined or discretization order increases.

Both our BFBT preconditioner and the viscous block preconditioner are based on our parallel multigrid, HMG. The combination of high-order accurate intergrid interpolations and restrictions together with efficient smoothers and their careful implementation results in algorithmic optimality for high-order discretizations and locally adapted meshes. We demonstrate the optimality throughout this work in the form of convergence rates, setup and solver runtimes, and time-to-accuracy. HMG is central to achieving excellent performance and parallel scalability, which we achieve due to algorithm design, computationally efficient implementation, overlap of communication with computation, and code optimizations down to low-level kernels.

In essence, the computational challenges of mantle convection can be addressed with innovations at the intersection of sophisticated numerical mathematics, algorithm design for shared and distributed memory parallel architectures, careful implementation, and detailed numerical experimentation and performance assessment.

9.2 Implications for mantle flow modeling

Building on algorithmic innovations for implicit solvers described in this dissertation, we are able to represent the depth and distribution of oceanic trenches—the most extreme topographic features on Earth's surface. Trench depth reflects both the downward pull from plate-driving forces [121] and the variable resistance associated with seismic coupling from great earthquakes [108]. We are able to forward-predict the width ($\sim$50 km) and depth ($\sim$10 km) of oceanic trenches on a global scale while predicting plate motions (Figure 9.1). The simultaneous prediction of these quantities—large-scale flow and fine-scale stress at plate boundaries—in a model with realistic, nonlinear rheology employing

scalable, robust solvers opens new directions for geophysical research. Solver robustness to thin plate boundaries is crucial, as can be seen in Figure 9.2, where we observe great sensitivity of the simulation outcome (in terms of plate velocities) to the width of plate boundary regions. Our scalable solver in combination with adjoints, which are a byproduct of the Newton solver, will allow systematic inference of uncertain parameters in global mantle flow systems with tectonic plates. For regional mantle models, a systematic inference approach for the nonlinear constitutive parameters n and τ_{yield}, and plate coupling factors $w(\boldsymbol{x})$, for several subduction zones was illustrated [93]. Adjoint-based inversions will require thousands of forward model solutions, so that availability of the scalable implicit solver presented here is paramount.

Bringing observations of topography (trench depth), plate motions, and others into a global inversion will allow the merging of two distinct geophysical approaches at different scales addressing different questions. First, what is the degree of coupling associated with great earthquakes? In particular, we seek to determine whether that coupling is due to the frictional properties of the incoming plate or the magnitude of normal stress across the fault driven by tectonic processes [101,103]. The second question concerns the forces driving and resisting global plate motion and the degree to which inter-plate coupling governs plate motions [49,59,110]. These questions have eluded solution over the past three decades. Bridging the local-to-global scales, with modern data sets, will arguably allow us to make an important leap toward the simultaneous solution of two of the most fundamental questions in Earth sciences.

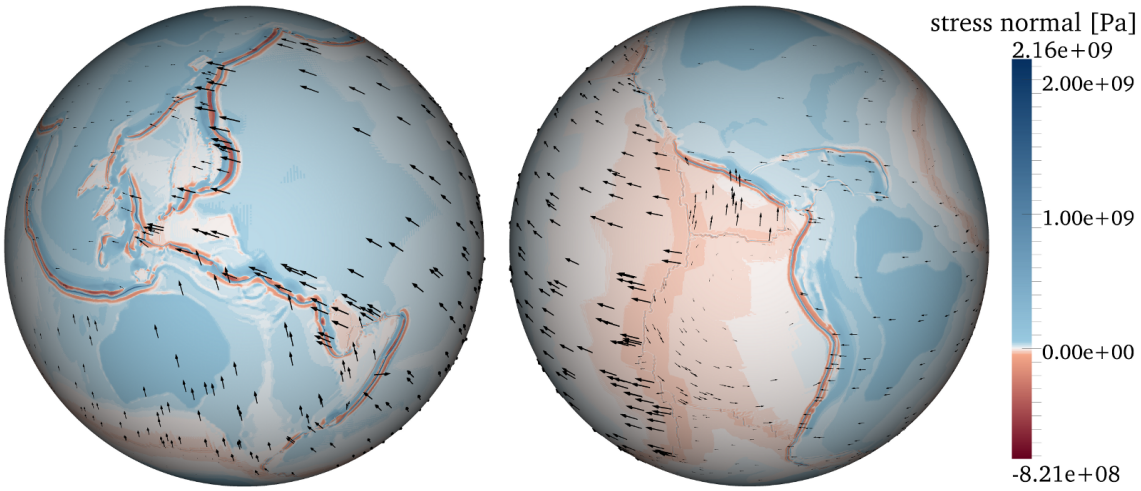


Figure 9.1: Surficial visualization of nonlinear mantle flow simulation: View centered on 180°W *(left)* and 90°W *(right)* showing north–westward motion of the Pacific Plate *(black arrows)* and the total normal stress field *(color coded)*. This stress is proportional to the dynamic topography and for the first time we are able to forward predict narrow (∼50 km in width) ocean trenches *(narrow lines with dark blue color)* along plate boundaries in a global model with plate motions.

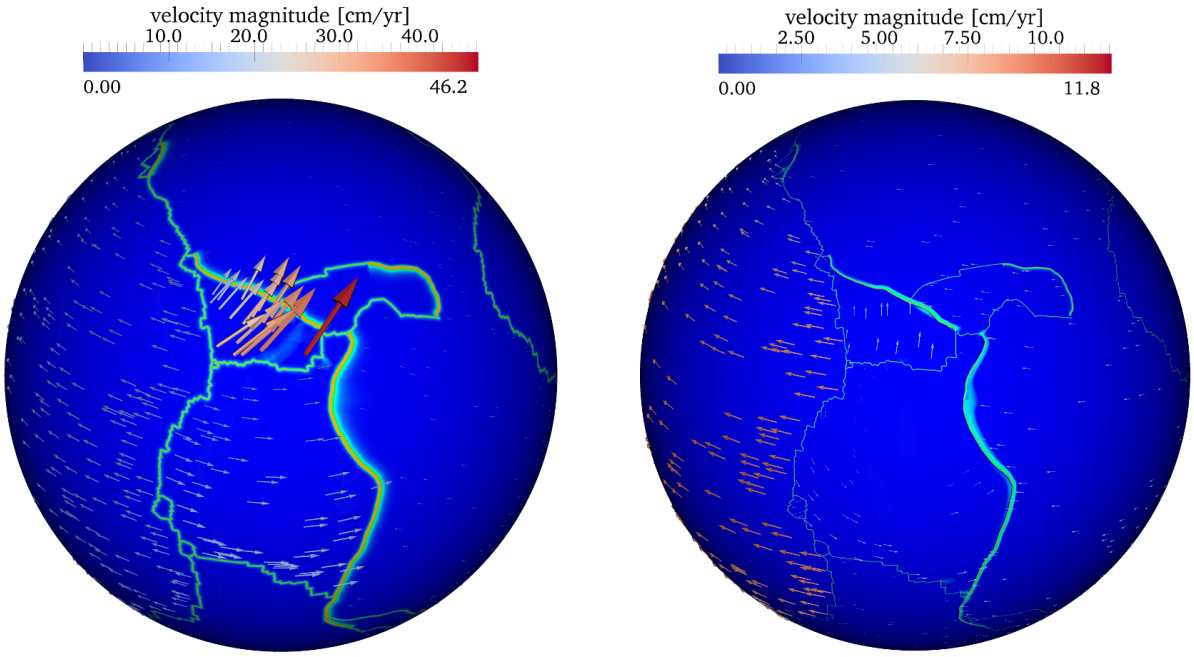


Figure 9.2: Comparison of Earth plate velocities of a low-fidelity model *(left)* and a high-fidelity model *(right)* with thinner plate boundaries. Significant sensitivity of velocities of the Cocos Plate *(in center)* are observed. This illustrates the importance of the solver's ability to handle a wide range of values for plate boundary width.

Bibliography

[1] Gabriel Acosta, Ricardo G. Durán, and Ariel L. Lombardi. Weighted Poincaré and Korn inequalities for Hölder α domains. *Mathematical Methods in the Applied Sciences*, 29(4):387–400, 2006.

[2] Volkan Akçelik, George Biros, and Omar Ghattas. Parallel multiscale Gauss–Newton–Krylov methods for inverse wave propagation. In *Proceedings of IEEE/ACM SC2002 Conference*, Baltimore, MD, November 2002.

[3] Volkan Akçelik, George Biros, Omar Ghattas, Judith Hill, David Keyes, and Bart van Bloeman Waanders. Parallel PDE-constrained optimization. In M. Heroux, P. Raghaven, and H. Simon, editors, *Parallel Processing for Scientific Computing*. SIAM, 2006.

[4] Triantaphyllos R. Akylas. Advanced fluid dynaamics (lecture notes), 2014. Retrieved from http://web.mit.edu/1.63/www/.

[5] Laura Alisic, Michael Gurnis, Georg Stadler, Carsten Burstedde, and Omar Ghattas. Multi-scale dynamics and rheology of mantle flow with plates. *Journal of Geophysical Research*, 117:B10402, 2012.

[6] Laura Alisic, Michael Gurnis, Georg Stadler, Carsten Burstedde, Lucas C. Wilcox, and Omar Ghattas. Slab stress and strain rate as constraints on global mantle flow. *Geophysical Research Letters*, 37:L22308, 2010.

[7] Donald F. Argus, Richard G. Gordon, and Charles DeMets. Geologically current motion of 56 plates relative to the no-net-rotation reference frame. *Geochemistry, Geophysics, Geosystems*, 12(11), 2011. Q11001.

[8] Santiago Badia, Alberto F. Martín, and Javier Principe. A highly scalable parallel implementation of balancing domain decomposition by constraints. *SIAM Journal on Scientific Computing*, 36(2):C190–C218, 2014.

[9] Allison H. Baker, Robert D. Falgout, Tzanio V. Kolev, and Ulrike Meier Yang. Scaling hypre's multigrid solvers to 100,000 cores. In Michael W. Berry, Kyle A. Gallivan, Efstratios Gallopoulos, Ananth Grama, Bernard Philippe, Yousef Saad, and Faisal Saied, editors, *High-Performance Scientific Computing*, pages 261–279. Springer London, 2012.

[10] Satish Balay, Shrirang Abhyankar, Mark F. Adams, Jed Brown, Peter Brune, Kris Buschelman, Lisandro Dalcin, Victor Eijkhout, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Lois Curfman McInnes, Karl Rupp, Barry F. Smith, Stefano Zampini, and Hong Zhang. PETSc users manual. Technical Report ANL-95/11 - Revision 3.6, Argonne National Laboratory, 2015.

[11] Wolfgang Bangerth and Timo Heister. ASPECT*: Advanced Solver for Problems in Earth's ConvecTion*. Computational Infrastructure in Geodynamics, 2015.

[12] Michele Benzi, Gene H. Golub, and Jörg Liesen. Numerical solution of saddle point problems. *Acta Numerica*, 14:1–137, 2005.

[13] Magali I. Billen and Greg Hirth. Rheologic controls on slab dynamics. *Geochemistry, Geophysics, Geosystems*, 8:Q08012, 2007.

[14] Peter Bird. An updated digital model of plate boundaries. *Geochemistry, Geophysics, Geosystems*, 4(3), 2003.

[15] Alfio Borzì and Volker Schulz. *Computational Optimization of Systems Governed by Partial Differential Equations*. SIAM, 2012.

[16] Dietrich Braess. *Finite elements*. Cambridge University Press, Cambridge, third edition, 2007. Theory, fast solvers, and applications in elasticity theory, Translated from the German by Larry L. Schumaker.

[17] James H. Bramble. *Multigrid Methods*. Longman Scientific & Technical, 1993.

[18] James H. Bramble, Joseph E. Pasciak, Jun Ping Wang, and Jinchao Xu. Convergence estimates for multigrid algorithms without regularity assumptions. *Mathematics of Computation*, 57(195):23–45, 1991.

[19] James H. Bramble, Joseph E. Pasciak, and Jinchao Xu. Parallel multilevel preconditioners. *Mathematics of Computation*, 55(191):1–22, 1990.

[20] James H. Bramble and Xuejun Zhang. The analysis of multigrid methods. In P. G Ciarlet and J. Lions, editors, *Handbook of numerical analysis, Vol. VII*, pages 173–415. North-Holland, Amsterdam, 2000.

[21] James H. Bramble and Xuejun Zhang. Uniform convergence of the multigrid v-cycle for an anisotropic problem. *Mathematics of Computation*, 70(234):453–470, 2001.

[22] Achi Brandt and Oren E. Livne. *Multigrid techniques—1984 guide with applications to fluid dynamics*, volume 67 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2011. Revised edition of the 1984 original.

[23] Franco Brezzi and Michel Fortin. *Mixed and hybrid finite element methods*, volume 15 of *Springer Series in Computational Mathematics*. Springer-Verlag, New York, 1991.

[24] Tan Bui-Thanh, Omar Ghattas, James Martin, and Georg Stadler. A computational framework for infinite-dimensional Bayesian inverse problems Part I: The linearized case, with application to global seismic inversion. *SIAM Journal on Scientific Computing*, 35(6):A2494–A2523, 2013.

[25] Hans-Peter Bunge, C.R. Hagelberg, and B.J. Travis. Mantle circulation models with variational data assimilation: Inferring past mantle flow and structure from plate motion histories and seismic tomography. *Geophysical Journal International*, 152:280–301, 2003.

[26] Carsten Burstedde, Omar Ghattas, Michael Gurnis, Tobin Isaac, Georg Stadler, Tim Warburton, and Lucas C. Wilcox. Extreme-scale AMR. In *SC10: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. ACM/IEEE, 2010.

[27] Carsten Burstedde, Omar Ghattas, Michael Gurnis, Eh Tan, Tiankai Tu, Georg Stadler, Lucas C. Wilcox, and Shijie Zhong. Scalable adaptive mantle convection simulation on petascale supercomputers. In *SC08: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. ACM/IEEE, 2008.

[28] Carsten Burstedde, Omar Ghattas, Georg Stadler, Tiankai Tu, and Lucas C. Wilcox. Parallel scalable adjoint-based adaptive solution for variable-viscosity Stokes flows. *Computer Methods in Applied Mechanics and Engineering*, 198:1691–1700, 2009.

[29] Carsten Burstedde, Georg Stadler, Laura Alisic, Lucas C. Wilcox, Eh Tan, Michael Gurnis, and Omar Ghattas. Large-scale adaptive mantle convection simulation. *Geophysical Journal International*, 192(3):889–906, 2013.

[30] Carsten Burstedde, Lucas C. Wilcox, and Omar Ghattas. `p4est`: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees. *SIAM Journal on Scientific Computing*, 33(3):1103–1133, 2011.

[31] Daniela Calvetti and Erkki Somersalo. *Introduction to Bayesian Scientific Computing: Ten Lectures on Subjective Computing*. Springer, New York, 2007.

[32] Lawrence M. Cathles. *Viscosity of the Earth's Mantle*. Princeton University Press, 1975.

[33] Tony F. Chan, Gene H. Golub, and Pep Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 20(6):1964–1977, 1999.

[34] Seng-Kee Chua and Richard L. Wheeden. Estimates of best constants for weighted Poincaré inequalities on convex domains. *Proceedings of the London Mathematical Society*, 93(1):197–226, 2006.

[35] Michel Crouzeix and Pierre-Arnaud Raviart. Conforming and nonconforming finite element methods for solving the stationary Stokes equations. I. *Rev. Française Automat. Informat. Recherche Opérationnelle Sér. Rouge*, 7(R-3):33–75, 1973.

[36] Charles DeMets, Richard G. Gordon, and Donald F. Argus. Geologically current plate motions. *Geophysical Journal International*, 181(1):1–80, 2010.

[37] Donald J. DePaolo, Thure E. Cerling, Sidney R. Hemming, Andrew H. Knoll, Frank M. Richter, Leigh H. Royden, Roberta L. Rudnick, Lars Stixrude, and James S. Trefil. Origin and Evolution of Earth: Research Questions for a Changing Planet. National Academies Press, Committee on Grand Research Questions in the Solid Earth Sciences, National Research Council of the National Academies, 2008.

[38] Michel O. Deville, Paul F. Fischer, and Ernest H. Mund. *High-order methods for incompressible fluid flow*, volume 9 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, Cambridge, 2002.

[39] Jean Donea and Antonio Huerta. *Finite Element Methods for Flow Problems*. John Wiley & Sons, 2003.

[40] Jack Dongarra, Jeffrey Hittinger, John Bell, Luis Chacón, Robert Falgout, Michael Heroux, Paul Hovland, Esmond Ng, Clayton Webster, and Stefan Wild. Applied mathematics research for exascale computing. Report of the DOE/ASCR Exascale Mathematics Working Group, March 2014.

[41] Stanley C. Eisenstat and Homer F. Walker. Globally convergent inexact Newton methods. *SIAM Journal on Optimization*, 4(2):393–422, 1994.

[42] Stanley C. Eisenstat and Homer F. Walker. Choosing the forcing terms in an inexact Newton method. *SIAM Journal on Scientific Computing*, 17(1):16–32, 1996.

[43] Howard C. Elman. Preconditioning for the steady-state Navier–Stokes equations with low viscosity. *SIAM Journal on Scientific Computing*, 20(4):1299–1316, 1999.

[44] Howard C. Elman, Victoria E. Howle, John Shadid, Robert Shuttleworth, and Raymond S. Tuminaro. Block preconditioners based on approximate commutators. *SIAM Journal on Scientific Computing*, 27(5):1651–1668, 2006.

[45] Howard C. Elman, Victoria E. Howle, John Shadid, Robert Shuttleworth, and Raymond S. Tuminaro. A taxonomy and comparison of parallel block multi-level preconditioners for the incompressible Navier–Stokes equations. *Journal of Computational Physics*, 227(3):1790–1808, 2008.

[46] Howard C. Elman, David J. Silvester, and Andrew J. Wathen. *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics*. Oxford University Press, 2014.

[47] Howard C. Elman and Raymond S. Tuminaro. Boundary conditions in approximate commutator preconditioners for the Navier–Stokes equations. *Electronic Transactions on Numerical Analysis*, 35:257–280, 2009.

[48] Pearl H. Flath, Lucas C. Wilcox, Volkan Akçelik, Judy Hill, Bart van Bloemen Waanders, and Omar Ghattas. Fast algorithms for Bayesian uncertainty quantification in large-scale linear inverse problems based on low-rank partial Hessian approximations. *SIAM Journal on Scientific Computing*, 33(1):407–432, 2011.

[49] Donald Forsyth and Seiya Uyeda. On the relative importance of the driving forces of plate motion. *Geophysical Journal International*, 43(1):163–200, 1975.

[50] Alessandro M. Forte and W. Richard Peltier. Plate tectonics and aspherical earth structure: The importance of poloidal-toroidal coupling. *Journal of Geophysical Research: Solid Earth*, 92(B5):3645–3679, 1987.

[51] Andrew Fowler. *Mathematical geoscience*. Springer, 2011.

[52] Mikito Furuichi, Dave A. May, and Paul J. Tackley. Development of a Stokes flow solver robust to large viscosity jumps using a Schur complement approach with mixed precision arithmetic. *Journal of Computational Physics*, 230(24):8835–8851, 2011.

[53] Vivette Girault and Pierre-Arnaud Raviart. *Finite element methods for Navier-Stokes equations*, volume 5 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1986. Theory and algorithms.

[54] Roland Glowinski and Jinchao Xu. *Numerical Methods for Non-Newtonian Fluids: Special Volume*, volume 16 of *Handbook of Numerical Analysis*. North-Holland, 2011.

[55] Björn Gmeiner, Ulrich Rüde, Holger Stengel, Christian Waluga, and Barbara Wohlmuth. Performance and scalability of Hierarchical Hybrid Multigrid solvers for Stokes systems. *SIAM Journal on Scientific Computing*, 37(2):C143–C168, 2015.

[56] Oscar Gonzalez and Andrew M. Stuart. *A first course in continuum mechanics*. Cambridge Texts in Applied Mathematics. Cambridge University Press, Cambridge, 2008.

[57] Piotr P. Grinevich and Maxim A. Olshanskii. An iterative method for the Stokes-type problem with variable viscosity. *SIAM Journal on Scientific Computing*, 31(5):3959–3978, 2009.

[58] Wolfgang Hackbusch. *Multigrid Methods and Applications*, volume 4 of *Springer Series in Computational Mathematics*. Springer, 1985.

[59] Bradford H. Hager and Richard J. O'Connell. A simple global model of plate dynamics and mantle convection. *Journal of Geophysical Research: Solid Earth*, 86(B6):4843–4867, 1981.

[60] Norman A. Haskell. The motion of a viscous fluid under a surface load. *Journal of Applied Physics*, 6(8):265–269, 1935.

[61] Matthias Heinkenschloss and Luis N. Vicente. Analysis of inexact trust–region SQP algorithms. *SIAM Journal on Optimization*, 12(2):283–302, 2001.

[62] Vincent Heuveline and Friedhelm Schieweck. On the inf-sup condition for higher order mixed FEM on meshes with hanging nodes. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(1):1–20, 2007.

[63] Jason E Hicken. Inexact Hessian-vector products in reduced-space differential-equation constrained optimization. *Optimization and Engineering*, 15(3):575–608, 2014.

[64] Thomas W.C. Hilde and Seiya Uyeda. Trench depth: Variation and significance. In Thomas W.C. Hilde and Seiya Uyeda, editors, *Geodynamics of the Western Pacific-Indonesian Region*, volume 11 of *Geodynamics Series*, pages 75–89. American Geophysical Union, 1983.

[65] Michael Hintermüller and Georg Stadler. An infeasible primal-dual algorithm for total variation-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28(1):1–23, 2006.

[66] Michael Hinze, Rene Pinnau, Michael Ulbrich, and Stefan Ulbrich. *Optimization with PDE Constraints*. Springer, 2009.

[67] André Horbach, Hans-Peter Bunge, and Jens Oeser. 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 - International Journal on Geomathematics*, 5(2):163–194, 2014.

[68] Kolumban Hutter. *Theoretical Glaciology*. Mathematical Approaches to Geophysics. D. Reidel Publishing Company, 1983.

[69] Olaf Ippisch and Markus Blatt. Scalability test of $\mu\varphi$ and the parallel algebraic multigrid solver of DUNE-ISTL. In B. Mohr and W. Frings, editors, *Jülich Blue Gene/P Extreme Scaling Workshop 2011, Technical Report FZJ-JSC-IB-2011-02*, 2011.

[70] Tobin Isaac, Carsten Burstedde, Lucas C. Wilcox, and Omar Ghattas. Recursive algorithms for distributed forests of octrees. *SIAM Journal on Scientific Computing*, 37(5), September 2015.

[71] Tobin Isaac, Georg Stadler, and Omar Ghattas. Solution of nonlinear Stokes equations discretized by high-order finite elements on nonconforming and anisotropic meshes, with application to ice sheet dynamics. *SIAM Journal on Scientific Computing*, 37(6):B804–B833, 2015.

[72] Alik Ismail-Zadeh, Gerald Schubert, Igor Tsepelev, and Alexander Korotkii. Inverse problem of thermal convection: numerical approach and application to mantle plume restoration. *Physics of The Earth and Planetary Interiors*, 145(1-4):99–114, 2004.

[73] Volker John, Kristine Kaiser, and Julia Novo. Finite element methods for the incompressible Stokes equations with variable viscosity. *ZAMM - Zeitschrift für Angewandte Mathematik und Mechanik*, 96(2):205–216, 2016.

[74] Jari Kaipio and Erkki Somersalo. *Statistical and Computational Inverse Problems*, volume 160 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 2005.

[75] Vasileios Karakasis, Theodoros Gkountouvas, Kornilios Kourtis, Georgios Goumas, and Nectarios Koziris. An extended compression format for the optimization of sparse matrix-vector multiplication. *IEEE Transactions on Parallel and Distributed Systems*, 24(10):1930–1940, Oct 2013.

[76] David Kay, Daniel Loghin, and Andrew Wathen. A preconditioner for the steady-state Navier-Stokes equations. *SIAM Journal on Scientific Computing*, 24(1):237–256, 2002.

[77] Corné Kreemer, William E. Holt, and A. John Haines. An integrated global model of present-day plate motions and plate boundary deformation. *Geophysical Journal International*, 154(1):8–34, 2003.

[78] Martin Kronbichler, Timo Heister, and Wolfgang Bangerth. High accuracy mantle convection simulation through modern numerical methods. *Geophysical Journal International*, 191(1):12–29, October 2012.

[79] Robert Michael Lewis and Stephen G. Nash. Using inexact gradients in a multilevel optimization algorithm. *Computational Optimization and Applications*, 56(1):39–61, 2013.

[80] Lijun Liu, Sonja Spasojevic, and Michael Gurnis. Reconstructing Farallon plate subduction beneath North America back to the late Cretaceous. *Science*, 322(5903):934–938, 2008.

[81] Anders Logg, Kent-Andre Mardal, and Garth Wells. *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS book*, volume 84. Springer Science & Business Media, 2012.

[82] James Martin, Lucas C. Wilcox, Carsten Burstedde, and Omar Ghattas. A stochastic Newton MCMC method for large-scale statistical inverse problems with application to seismic inversion. *SIAM Journal on Scientific Computing*, 34(3):A1460–A1487, 2012.

[83] Dave A. May, Jed Brown, and Laetitia Le Pourhiet. pTatin3D: High-performance methods for long-term lithospheric dynamics. In *SC14: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, pages 274–284. IEEE Press, 2014.

[84] Dave A. May, Jed Brown, and Laetitia Le Pourhiet. A scalable, matrix-free multigrid preconditioner for finite element discretizations of heterogeneous Stokes flow. *Computer Methods in Applied Mechanics and Engineering*, 290:496–523, 2015.

[85] Dave A. May and Louis Moresi. Preconditioned iterative methods for Stokes flow problems arising in computational geodynamics. *Physics of the Earth and Planetary Interiors*, 171:33–47, 2008.

[86] Dan McKenzie. The generation and compaction of partially molten rock. *Journal of Petrology*, 25(3):713–765, 1984.

[87] James Milano and Pamela Lembke. *IBM system Blue Gene solution: Blue Gene/Q hardware overview and installation planning*. IBM Redbooks, 2013.

[88] Jerry X. Mitrovica. Haskell [1935] revisited. *Journal of Geophysical Research: Solid Earth*, 101(B1):555–569, 1996.

[89] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer Verlag, Berlin, Heidelberg, New York, second edition, 2006.

[90] Clemens Pechstein and Robert Scheichl. Weighted Poincaré inequalities. *IMA Journal of Numerical Analysis*, 33(2):652–686, 2013.

[91] Noemi Petra, James Martin, Georg Stadler, and Omar Ghattas. A computational framework for infinite-dimensional Bayesian inverse problems: Part II. Stochastic Newton MCMC with application to ice sheet inverse problems. *SIAM Journal on Scientific Computing*, 36(4):A1525–A1555, 2014.

[92] Kumbakonam R. Rajagopal. Mechanics of non-Newtonian fluids. In G.P. Galdi and J. Necas, editors, *Recent Developments in Theoretical Fluid Mechanics*, volume 291, pages 129–162. Longman's Scientific and Technical, 1993.

[93] Vishagan Ratnaswamy, Georg Stadler, and Michael Gurnis. Adjoint-based estimation of plate coupling in a non-linear mantle flow model: theory and examples. *Geophysical Journal International*, 202(2):768–786, 2015.

[94] Sebastian Reiter, Andreas Vogel, Ingo Heppner, Martin Rupp, and Gabriel Wittum. A massively parallel geometric multigrid solver on hierarchically distributed grids. *Computing and Visualization in Science*, 16(4):151–164, 2013.

[95] Diego Rossinelli, Babak Hejazialhosseini, Panagiotis Hadjidoukas, Costas Bekas, Alessandro Curioni, Adam Bertsch, Scott Futral, Steffen J. Schmidt, Nikolaus A. Adams, and Petros Koumoutsakos. 11 PFLOP/s simulations of cloud cavitation collapse. In *Proceedings of the*

International Conference on High Performance Computing, Networking, Storage and Analysis, SC '13, pages 3:1–3:13, New York, NY, USA, 2013. ACM.

[96] Johann Rudi. Parallel, robust geometric multigrid for adaptive high-order meshes and highly heterogeneous, nonlinear Stokes flow of earth's mantle. unpublished competition paper, Finalist at the 26$^{\text{th}}$ Robert J. Melosh Medal Competition, Duke University, Durham, North Carolina, USA, 2015.

[97] Johann Rudi, A. Cristiano I. Malossi, Tobin Isaac, Georg Stadler, Michael Gurnis, Peter W. J. Staar, Yves Ineichen, Costas Bekas, Alessandro Curioni, and Omar Ghattas. An extreme-scale implicit solver for complex PDEs: Highly heterogeneous flow in earth's mantle. In *SC15: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 5:1–5:12. ACM, 2015.

[98] Johann Rudi, Georg Stadler, and Omar Ghattas. μ-BFBT preconditioner for Stokes flow problems with highly heterogeneous viscosity. unpublished competition paper, Winner of the Student Paper Competition at the 14$^{\text{th}}$ Copper Mountain Conference on Iterative Methods, Copper Mountain, Colorado, USA, 2016.

[99] Johann Rudi, Georg Stadler, and Omar Ghattas. Weighted BFBT preconditioner for Stokes flow problems with highly heterogeneous viscosity. *SIAM Journal on Scientific Computing*, 39(5):S272–S297, 2017.

[100] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D. Nonlinear Phenomena*, 60(1-4):259–268, 1992. Experimental mathematics: computational issues in nonlinear science (Los Alamos, NM, 1991).

[101] Larry Ruff and Hiroo Kanamori. Seismic coupling and uncoupling at subduction zones. *Tectonophysics*, 99(2):99–117, 1983. Third Annual Symposium of the Geodynamics Research Program, Texas A & M University.

[102] Yousef Saad. *Iterative methods for sparse linear systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, second edition, 2003.

[103] Christopher H. Scholz and Jaime Campos. The seismic coupling of subduction zones revisited. *Journal of Geophysical Research: Solid Earth*, 117(B5), 2012.

[104] Gerald Schubert, Donald L. Turcotte, and Peter Olson. *Mantle Convection in the Earth and Planets*. Cambridge University Press, 2001.

[105] David J. Silvester, Howard C. Elman, David Kay, and Andrew J. Wathen. Efficient preconditioning of the linearized Navier–Stokes equations for incompressible flow. *Journal of Computational and Applied Mathematics*, 128(1–2):261–279, 2001. Numerical analysis 2000, Vol. VII, Partial differential equations.

[106] Valeria Simoncini and Daniel B. Szyld. Theory of inexact Krylov subspace methods and applications to scientific computing. *SIAM Journal on Scientific Computing*, 25(2):454–477, 2003.

[107] Mark Simons and Bradford H. Hager. Localization of the gravity field and the signature of glacial rebound. *Nature*, 390:500–504, 1997.

[108] Teh-Ru Alex Song and Mark Simons. Large trench-parallel gravity variations predict seismogenic behavior in subduction zones. *Science*, 301(5633):630–633, 2003.

[109] Sonja Spasojevic, Lijun Liu, and Michael Gurnis. Adjoint models of mantle convection with seismic, plate motion, and stratigraphic constraints: North america since the late cretaceous. *Geochemistry, Geophysics, Geosystems*, 10(5), 2009.

[110] Georg Stadler, Michael Gurnis, Carsten Burstedde, Lucas C. Wilcox, Laura Alisic, and Omar Ghattas. The dynamics of plate tectonics and mantle flow: From local to global scales. *Science*, 329(5995):1033–1038, 2010.

[111] Rolf Stenberg and Manil Suri. Mixed *hp* finite element methods for problems in elasticity and Stokes flow. *Numerische Mathematik*, 72(3):367–389, 1996.

[112] Andrew M. Stuart. Inverse problems: A Bayesian perspective. *Acta Numerica*, 19:451–559, 2010.

[113] Hari Sundar, George Biros, Carsten Burstedde, Johann Rudi, Omar Ghattas, and Georg Stadler. Parallel geometric-algebraic multigrid on unstructured forests of octrees. In *SC12: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, Salt Lake City, UT, 2012. ACM/IEEE.

[114] Albert Tarantola. *Inverse Problem Theory and Methods for Model Parameter Estimation*. SIAM, Philadelphia, PA, 2005.

[115] Ulrich Trottenberg, Cornelius W. Oosterlee, and Anton Schüller. *Multigrid*. Academic Press, Inc., San Diego, CA, 2001. With contributions by A. Brandt, P. Oswald and K. Stüben.

[116] Donald L. Turcotte and Gerald Schubert. *Geodynamics*. Cambridge University Press, 2nd edition, 2002.

[117] Jasper van den Eshof and Gerard L.G. Sleijpen. Inexact Krylov subspace methods for linear systems. *SIAM Journal on Matrix Analysis and Applications*, 26(1):125–153, 2004.

[118] Rüdiger Verfürth. *A Posteriori Error Estimation Techniques for Finite Element Methods*. Oxford University Press, 2013.

[119] Jennifer Worthen, Georg Stadler, Noemi Petra, Michael Gurnis, and Omar Ghattas. Towards adjoint-based inversion for rheological parameters in nonlinear viscous mante flow. *Physics of the Earth and Planetary Interiors*, 234:23–34, 2014.

[120] Jinchao Xu. Iterative methods by space decomposition and subspace correction. *SIAM Review*, 34(4):581–613, 1992.

[121] Shijie Zhong and Michael Gurnis. Controls on trench topography from dynamic models of subducted slabs. *Journal of Geophysical Research: Solid Earth*, 99(B8):15683–15695, 1994.