



Efficient Uzawa algorithms with projection strategies for geodynamic Stokes flow

Deok-Kyu Jang¹, Kyeong-Min Lee², Cedric Thieulot³, Whan-Hyuk Choi^{1,4}, and Byung-Dal So^{2,5}

¹KNU Research Institute of Mathematical Sciences, Kangwon National University, Chuncheon, Republic of Korea

²Interdisciplinary Program in Earth Environmental System Science & Engineering, Kangwon National University, Chuncheon, Republic of Korea

³Department of Earth Sciences, Utrecht University, Princetonlaan 8A, the Netherlands

⁴Department of Mathematics, Kangwon National University, Chuncheon, Republic of Korea

⁵Department of Geophysics, Kangwon National University, Chuncheon, Republic of Korea

Correspondence: Byung-Dal So (bdso@kangwon.ac.kr)

Received: 5 November 2025 – Discussion started: 18 November 2025

Revised: 20 May 2026 – Accepted: 27 August 2026 – Published: 19 October 2026

Abstract. Stokes equations are often difficult to handle in geodynamic modelling because they form a saddle-point system and involve strong variations in viscosity. In this work, we reformulate the Uzawa iteration through finite element variational principles, replacing the standard algebraic residual with a weighted L^2 form that yields a robust preconditioned conjugate gradient iteration for the Schur complement system. Residuals are evaluated in weak form, giving an effect similar to that of a mass-matrix preconditioner. We also add a projection step so that the velocity field remains nearly divergence-free. These updates made the solver converge faster and behave more stably than the standard Uzawa method. This reformulation provides a complementary algorithmic perspective on Schur complement preconditioning within the finite element variational framework. The modified approach was tested in several cases, including ABC flow, SolCx, SolKz, mantle convection, block sinking, multi-sinker, and compressible convection under the Anelastic Liquid Approximation.

Negredo et al., 2022; Kim and So, 2020), and lithospheric buckling (Do et al., 2023; Xie et al., 2024). The mantle has a viscosity on the order of 10^{19} – 10^{23} Pa s, and the associated flows occur at low Reynolds number. As a result, inertial terms are negligible, and the Navier–Stokes equations reduce to the elliptic Stokes system. This formulation produces large and sparse saddle-point systems that are challenging to solve, especially when viscosity varies by several orders of magnitude and complex rheologies are present (Moresi and Gurnis, 1996; Zhong et al., 2000).

The spatial discretisation of the Stokes equations in geodynamics is frequently performed using finite element (FEM), finite difference (FDM) procedures, or finite volume (FVM) techniques. These algebraic systems retain the saddle-point structure of the velocity–pressure formulation. For this reason, direct solvers are not a realistic method for three-dimensional problems due to memory and computational expense. Due to this reason, iterative Krylov subspace approaches such as the conjugate gradient (CG) method for symmetric positive definite systems (Hestenes and Stiefel, 1952) and the generalized minimal residual method (GMRES) or its flexible variant (FGMRES) for nonsymmetric systems (Saad and Schultz, 1986; Saad, 2003) are often used for large-scale simulations. In the absence of suitable preconditioning, such methods progress slowly, especially when strong viscosity contrasts reduce the conditioning of the operator.

1 Introduction

Geodynamic modelling of the mantle and lithosphere is concerned with processes such as subduction (Gerya, 2022; Li and Gurnis, 2023; Sime et al., 2024), continental rifting (Huisman et al., 2001; Brune et al., 2012; So and Yuen, 2015), mantle plume upwelling (Ismail-Zadeh et al., 2006;

A common strategy is to eliminate the velocity variables and form a reduced Schur complement system for the pressure. The standard Uzawa algorithm (Arrow et al., 1958) applies Richardson-type iterations to this system and re-computes velocity at each step. The method is simple but converges slowly and is sensitive to the relaxation parameter, which limits its use in geodynamic applications. Later studies proposed variants with preconditioning and inexact solves, such as block preconditioners and mass-matrix approximations, to improve efficiency and stability (Cahouet and Chabard, 1988; Elman and Golub, 1994; Chen, 1998; Bai and Wang, 2008). Explicit formulations of adaptive Uzawa and conjugate-directions Uzawa have been presented, showing their relation to gradient and conjugate gradient iterations (Braess, 2001). In geodynamics, Uzawa-type updates have been implemented in mantle convection codes such as CitcomS (Zhong et al., 2000), and scalability studies have evaluated their performance on large parallel systems (Gmeiner et al., 2016). More generally, Schur complement systems can be solved using Krylov methods combined with suitable preconditioners, and much work has focused on block preconditioners, multigrid methods, and algebraic strategies for Stokes flow (Silvester and Wathen, 1994; May and Moresi, 2008; Kronbichler et al., 2012), including the weighted BFBT (w-BFBT) preconditioner that incorporates viscosity directly into the Schur complement approximation (Rudi et al., 2017). These approaches are now widely adopted in large-scale mantle convection modelling.

This study revisits the Uzawa framework from a variational perspective. The key observation is that defining residuals through weighted L^2 inner products in the finite element space recasts the Uzawa iteration as a preconditioned conjugate gradient method for the Schur complement system. Since the entire algorithm is expressed through inner products and variational forms, it naturally leverages the matrix assembly and parallel solver infrastructure available in general-purpose finite element libraries coupled with PETSc. Based on this reformulation, we propose two extensions. The first is a viscosity-weighted residual formulation that serves as a Schur complement preconditioner and whose convergence can be systematically improved. The second is a projection post-processing step that improves the divergence-free constraint at minimal additional cost.

The first extension changes how residuals are updated in the pressure space using inner products, which works the same as applying a mass-matrix preconditioner. This connects the L^2 -projected residual update to preconditioned conjugate gradient (PCG) iterations and brings mass-matrix stabilization into the Uzawa framework (Elman and Golub, 1994; Ramage and Wathen, 1994; Chen, 1998; Benzi et al., 2005). We also introduce a viscosity-weighted variant where the mass matrix is replaced by an η -weighted version. This idea relates to earlier work on Schur complement preconditioners for variable-viscosity Stokes flow (May and Moresi, 2008). Unlike block preconditioning, our method puts vis-

cosity weighting directly into the Uzawa iteration, making it a preconditioned conjugate gradient scheme with a preconditioner that better matches the true Schur complement spectrum. This gives faster and more stable convergence when viscosity contrasts are large.

The second extension is a post-processing projection step that projects velocity variable onto the constraint-satisfying subspace. For incompressible flow, this is the discrete Helmholtz–Hodge decomposition. This method was first introduced for time-dependent Navier–Stokes equations (Chorin, 1967), and later refined for better stability and accuracy (Bell et al., 1989; Guermond and Quartapelle, 1998; Brown et al., 2001; Pyo and Shen, 2007). These methods are standard in unsteady fluid dynamics but have received little attention for elliptic Stokes problems in geodynamics. We use projection after Uzawa iterations to reduce constraint residuals.

The remainder of this paper is organized as follows. Section 2 presents the mathematical formulation, including the Stokes equations, the saddle-point system, Uzawa-type algorithms, residual representations, and the Helmholtz–Hodge projection. Section 3 reports numerical experiments using manufactured benchmarks, incompressible and compressible mantle convection, and block sinking problems. Section 4 summarizes the main results and outlines future work.

2 Mathematical formulation

2.1 Stokes equations and saddle-point formulation

We consider the incompressible Stokes equations relevant to mantle convection. Because the Prandtl number of mantle materials is effectively infinite ($Pr \approx 10^{23}$), inertial terms in the Navier–Stokes equations can be neglected, and the governing system reduces to the elliptic Stokes equations (Schubert et al., 2001).

In a bounded domain $\Omega \subset \mathbb{R}^{n_d}$ ($n_d = 2, 3$), these are

$$\nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{g} = \mathbf{0}, \quad \nabla \cdot \mathbf{u} = 0,$$

where $\mathbf{u}$ is velocity, p is pressure, ρ is density, and $\mathbf{g}$ is gravitational acceleration. The Cauchy stress tensor is

$$\boldsymbol{\sigma} = -p\mathbf{I} + 2\eta D(\mathbf{u}), \quad D(\mathbf{u}) = \frac{1}{2} \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right),$$

with η the dynamic viscosity and $D(\mathbf{u})$ the strain-rate tensor.

In compact form, the Stokes equations then become

$$\nabla \cdot \left(\eta \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right) \right) - \nabla p + \rho \mathbf{g} = \mathbf{0}, \quad \nabla \cdot \mathbf{u} = 0.$$

In the case of constant viscosity with Dirichlet boundary conditions, the momentum equation reduces to a vector Laplacian.

Discretization with stable finite element pairs (Babuška, 1973; Brezzi, 1974; Silvester and Wathen, 1994; Logg et al.,

2012) gives the saddle-point system

$$\begin{pmatrix} K & G \\ G^T & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ h \end{pmatrix},$$

where K is the discrete viscosity matrix (symmetric positive definite) and G is the discrete gradient operator. The system is symmetric but indefinite, and efficient iterative solvers are required.

Eliminating velocity gives the Schur complement system

$$Sp = G^T K^{-1} f - h, \quad S = G^T K^{-1} G.$$

The Schur complement S is symmetric positive definite under the standard inf-sup stability condition. Forming S explicitly is too expensive, since each multiplication requires solving with K^{-1} , which is costly for large problems with variable viscosity. In practice, S is applied implicitly within iterative methods, together with suitable preconditioners.

The standard Uzawa algorithm applies Richardson-type iterations to update the pressure and recompute the velocity, with relaxation parameter $\omega > 0$. The scheme is simple, but its convergence is sensitive to the choice of ω and deteriorates with large viscosity contrasts.

Algorithm 1 Standard Uzawa.

Initialization: Set $p^0 = 0$
Iteration ($k \geq 0$):
Velocity update: Find u^{k+1} satisfying $Ku^{k+1} = f - Gp^k$
Pressure update: Compute $p^{k+1} = p^k + \omega(G^T u^{k+1} - h)$

To reduce this sensitivity, the relaxation parameter is updated adaptively at each step. This improves robustness compared with the fixed- ω method.

Algorithm 2 Adaptive Uzawa.

Initialization: Set $p^0 = 0$
Initial solve: Find u^1 satisfying $Ku^1 = f - Gp^0$
Initial residual: Compute $q^1 = h - G^T u^1$
Iteration ($k \geq 1$):
Auxiliary solve: Find z^k satisfying $Kz^k = Gq^k$
Step-size selection: Compute $\omega_k = (q^k)^T q^k / (Gq^k)^T z^k$
Pressure update: Compute $p^{k+1} = p^k - \omega_k q^k$
Velocity update: Compute $u^{k+1} = u^k + \omega_k z^k$
Residual update: Compute $q^{k+1} = h - G^T u^{k+1}$

The CD-U method accelerates convergence by constructing conjugate directions, in analogy with the conjugate gradient method. The explicit coefficients α_k and β_k highlight the connection with CG. By exploiting information from previous steps, CD-U achieves faster and more stable convergence, especially for problems with strong viscosity contrasts or heterogeneous coefficients.

2.2 Residual representation in Uzawa iterations

The definition of the residual is central to Uzawa-type algorithms. In the Schur complement formulation, the discrete residual is

$$q^k = h - G^T u^k,$$

which measures the violation of the incompressibility constraint. In its standard (vector) form, q^k is assembled directly as a discrete vector. This is simple but suffers from poor conditioning, and small singular values of G^T can lead to unstable step-size choices and slow convergence, especially with large viscosity contrasts.

An alternative is to define the residual variationally in the L^2 inner product on the pressure space Q :

$$\langle q^k, r \rangle = \langle h + \nabla \cdot u^k, r \rangle, \quad \forall r \in Q. \quad (1)$$

In discrete form this corresponds to solving a mass-matrix system,

$$M_p q^k = M_p h + b(u^k),$$

where $b(u^k)$ denotes the discrete constraint term. This formulation shows that Uzawa iterations implicitly apply the pressure mass matrix as a preconditioner (Cahouet and Chabard, 1988; Elman and Golub, 1994; Chen, 1998; Benzi et al., 2005). The resulting L^2 -projected residual is smoother and scale-independent, and yields more robust convergence at modest extra cost.

With this modification, the overall update structure of the standard, adaptive, and CD-U algorithms remains the same. Only the scalar coefficients change, as summarized below.

Standard Uzawa.

$$\langle p^{k+1}, r \rangle = \langle p^k, r \rangle - \omega \langle q^{k+1}, r \rangle, \quad \forall r \in Q. \quad (2)$$

Adaptive Uzawa. Using the L^2 residual definition in Eq. (1), the step size is updated as

$$\omega_{k+1} = \frac{\langle q^{k+1}, q^{k+1} \rangle}{\langle \nabla q^{k+1}, z^{k+1} \rangle}. \quad (3)$$

Conjugate-directions Uzawa (CD-U). With q^k defined in Eq. (1), the coefficients are

$$\alpha_k = \frac{\langle q^k, q^k \rangle}{\langle \nabla d^k, z^k \rangle}, \quad \beta_k = \frac{\langle q^{k+1}, q^{k+1} \rangle}{\langle q^k, q^k \rangle}. \quad (4)$$

Conjugate-directions Uzawa with η -weighted residuals (CD-U- η). The standard L^2 residual in Eq. (1) can be extended by replacing the mass matrix with an inverse-viscosity-weighted version, which we refer to as η -weighted for brevity. This introduces a viscosity scaling in the pressure space, which more closely reflects the spectrum of the true Schur complement for variable-viscosity problems. The

Algorithm 3 Conjugate-Directions Uzawa (CD-U).

Initialization: Set $p^0 = 0$
Initial solve: Find u^1 satisfying $Ku^1 = f - Gp^0$
Initial residual: Compute $q^1 = h - G^T u^1$, set $d^1 = -q^1$
Iteration ($k \geq 1$):
Direction solve: Find z^k satisfying $Kz^k = Gd^k$
Step-size selection: Compute $\alpha_k = (q^k)^T q^k / (Gd^k)^T z^k$
Pressure update: Compute $p^{k+1} = p^k + \alpha_k d^k$
Velocity update: Compute $u^{k+1} = u^k - \alpha_k z^k$
Residual update: Compute $q^{k+1} = h - G^T u^{k+1}$
Conjugate direction: Compute $\beta_k = (q^{k+1})^T q^{k+1} / (q^k)^T q^k$, set $d^{k+1} = -q^{k+1} + \beta_k d^k$

Algorithm 4 Conjugate-Directions Uzawa with η -weighted residuals (CD-U- η).

Initialization: Set $p^0 = 0$
Initial solve: Find u^1 satisfying $Ku^1 = f - Gp^0$
Initial residual: Find q^1 satisfying $\langle q^1, r \rangle = \langle h + \nabla \cdot u^1, r \rangle \quad \forall r \in Q$
Preconditioning: Find w^1 satisfying $\langle \frac{1}{\eta} w^1, r \rangle = \langle q^1, r \rangle \quad \forall r \in Q$, set $d^1 = -w^1$
Iteration ($k \geq 1$):
Direction solve: Find z^k satisfying $Kz^k = Gd^k$
Step-size selection: Compute $\alpha_k = \langle q^k, w^k \rangle / \langle \nabla d^k, z^k \rangle$
Pressure update: Compute $p^{k+1} = p^k + \alpha_k d^k$
Velocity update: Compute $u^{k+1} = u^k - \alpha_k z^k$
Residual update: Find q^{k+1} satisfying $\langle q^{k+1}, r \rangle = \langle h + \nabla \cdot u^{k+1}, r \rangle \quad \forall r \in Q$
Preconditioning: Find w^{k+1} satisfying $\langle \frac{1}{\eta} w^{k+1}, r \rangle = \langle q^{k+1}, r \rangle \quad \forall r \in Q$
Conjugate direction: Compute $\beta_k = \langle q^{k+1}, w^{k+1} \rangle / \langle q^k, w^k \rangle$, set $d^{k+1} = -w^{k+1} + \beta_k d^k$

resulting scheme is a preconditioned conjugate-gradient iteration in which both the residual q^k and its preconditioned form w^k appear in the update formulas.

This iteration can be interpreted as a Preconditioned Conjugate Gradient (PCG) method applied to the Schur complement system. Here q^k denotes the residual in the pressure space, and w^k its preconditioned form. The search directions d^k and coefficients α_k, β_k are then defined from inner products such as $\langle q^k, w^k \rangle$, following the standard PCG structure.

With the L^2 formulation, the preconditioner is the pressure mass matrix M_p , which is spectrally close to the identity. In this case $w^k = q^k$, and the scheme reduces to a Conjugate Gradient iteration with a trivial preconditioner. In the η -weighted formulation, the preconditioner is the viscosity-weighted mass matrix, so that $w^k \neq q^k$ and both quantities appear in the iteration. Note that the relation between q^k and w^k cannot be expressed pointwise as $w^k = \eta q^k$, since $\eta(x)$ varies spatially. Instead, w^k is defined variationally by

$$\langle \frac{1}{\eta} w^k, r \rangle = \langle q^k, r \rangle, \quad \forall r \in Q,$$

which ensures that w^k belongs to the pressure space Q and

serves as the Riesz representative of q^k under the η -weighted inner product. This formulation provides a consistent SPD preconditioner in the finite element setting.

In practice, at high resolution or extreme viscosity contrasts, the conjugate gradient iteration may stagnate when the denominators in the step-size and conjugacy coefficients become too small. To address this, we introduce an adaptive scaling parameter AAR that modifies the preconditioner weights from $1/\eta(x)$ to $\text{AAR}/\eta(x)$. Starting from $\text{AAR} = 1.0$, the parameter is progressively reduced whenever stagnation is detected. To prevent numerical instability in high-viscosity regions where the scaled weights become excessively small, we enforce a pointwise lower bound $\min_x (\text{AAR}/\eta(x)) \geq \epsilon$ for a prescribed threshold ϵ . The specific reduction strategy and the threshold ϵ are reported together with the numerical experiments.

In summary, while the L^2 formulation corresponds to a CG iteration with a trivial mass-matrix preconditioner, the η -weighted Uzawa method is a genuine PCG scheme with a physically motivated preconditioner that better reflects the spectral properties of the Schur complement, particularly under strong viscosity contrasts (May and Moresi, 2008). The

adaptive AAR strategy ensures robust convergence of this scheme across a wide range of resolutions and viscosity ratios.

2.3 Helmholtz–Hodge projection as post-processing

Uzawa-type iterations reduce the constraint violation only asymptotically, so the discrete constraint remains slightly violated after a finite number of steps. To enforce the constraint explicitly, we apply a projection step as a post-processing. The idea is to correct an intermediate velocity $\hat{u}$ by adding a gradient field so that the updated velocity satisfies the discrete constraint operator

$$\mathcal{C}(u) = h,$$

where $\mathcal{C}(u) = \nabla \cdot u$ for the incompressible case, and $\mathcal{C}(u) = \nabla \cdot (\bar{\rho}u)$ under the anelastic liquid approximation (ALA).

For the incompressible case, $\mathcal{C}(u) = \nabla \cdot u$, the projection requires solving a scalar Poisson problem

$$-\Delta \phi = h + \nabla \cdot \hat{u}, \quad u^{k+1} = \hat{u} + \nabla \phi.$$

This corresponds to the discrete Helmholtz–Hodge decomposition, which states that any vector field can be uniquely decomposed into a solenoidal part and a gradient part,

$$u^* = u + \nabla \phi, \quad \nabla \cdot u = 0.$$

Equivalently, the projection operator can be written as

$$\mathcal{P} = I - G(G^T G)^{-1} G^T,$$

which is symmetric and idempotent.

In variational form, the correction is obtained by solving

$$\langle \nabla \phi, \nabla \psi \rangle = \langle h + \nabla \cdot \hat{u}, \psi \rangle, \quad \forall \psi \in Q,$$

and updating the velocity consistently in the finite element space,

$$\langle u^{k+1}, w \rangle = \langle \hat{u}, w \rangle + \langle \nabla \phi, w \rangle, \quad \forall w \in V.$$

This weak formulation is important: a direct discrete update $u^{k+1} = \hat{u} + G\phi$ does not yield the same accuracy, whereas the variational form ensures consistency and better convergence. In practice, the projection can also be implemented component-wise to avoid assembling a large coupled system.

For the compressible case, $\mathcal{C}(u) = \nabla \cdot (\bar{\rho}u)$, and the projection takes the weighted form

$$-\nabla \cdot (\bar{\rho} \nabla \phi) = h + \nabla \cdot (\bar{\rho} \hat{u}), \quad u^{k+1} = \hat{u} + \nabla \phi,$$

so that $\nabla \cdot (\bar{\rho} u^{k+1}) = h$ is satisfied exactly. This weighted projection is the natural extension of the Helmholtz–Hodge decomposition to density-dependent flows.

Although projection methods of this type are standard in computational fluid dynamics, they have not been used previously as post-processing corrections in geodynamics. As we show in Sect. 3, this step reduces accumulated divergence errors and improves the long-term accuracy of mantle convection benchmarks.

A related approach in particle-in-cell methods is the conservative velocity interpolation (CVI) scheme (Wang et al., 2015), which enforces local mass conservation during interpolation but follows a different formulation from the projection-based correction used here. The CVI approach is specifically designed for quadrilateral Q1P0 elements, whereas our projection method is element-independent and applicable to general finite element discretizations.

3 Numerical Implementation

3.1 Finite Element Framework and Parallel Environment

We implemented the proposed algorithms using the open-source finite element library FEniCS (version 2019.2.0.13) (Logg et al., 2012; Alnæs et al., 2015). FEniCS provides a Python interface that simplifies variational form expression and automates finite element matrix assembly, which is useful in geodynamics where PDE systems often involve strongly variable rheologies (Zhamaletdinov et al., 2011; Vynnytska et al., 2013; Wilson et al., 2017; Lee et al., 2024). To solve the large sparse linear systems from discretizing the Stokes equations, we coupled FEniCS with PETSc (Balay et al., 2024). We used GMRES and CG with algebraic multigrid (AMG) preconditioning. Solver configurations were adjusted through PETSc runtime parameters.

All algorithms were implemented within the standard FEniCS–PETSc environment without custom code. This portability shows that the approach is reproducible, extendable, and transferable to other platforms, in line with current practices in large-scale geodynamic modeling.

3.2 Solver configuration and benchmark setup

Each Uzawa iteration with projection requires solving three linear systems: (i) the momentum equation for velocity, (ii) an L^2 -projection in the pressure space for residual representation and scalar potential projection, and (iii) a Poisson system for velocity correction.

The L^2 -projection system was solved with CG preconditioned by block Jacobi, which works well for the diagonally dominant mass matrices from finite element discretizations. The Poisson system for velocity correction was also solved with CG and BoomerAMG. All solvers used relative and absolute tolerances of 10^{-10} and 10^{-12} , so that the convergence behavior observed in our benchmarks is governed by the outer Uzawa iteration rather than by inner solver accuracy. For reference, the ASPECT code (Kronbichler et al.,

Algorithm 5 Projection post-processing.

<i>Input:</i>	Velocity $\hat{u}$ from an iterative solver (Case 1: $\nabla \cdot u = h$)	(Case 2: $\nabla \cdot (\bar{\rho} u) = h$)
<i>Scalar correction:</i>	$\langle \nabla \phi, \nabla \psi \rangle = \langle h + \nabla \cdot \hat{u}, \psi \rangle \quad \forall \psi \in Q$	$\langle \bar{\rho} \nabla \phi, \nabla \psi \rangle = \langle h + \nabla \cdot (\bar{\rho} \hat{u}), \psi \rangle \quad \forall \psi \in Q$
<i>Velocity update:</i>	$\langle u^{k+1}, w \rangle = \langle \hat{u}, w \rangle + \langle \nabla \phi, w \rangle \quad \forall w \in V$	

2012) uses default inner solve tolerances of 10^{-2} for the velocity block and 10^{-6} for the Schur complement block.

Uzawa iterations were terminated using a normalized divergence residual

$$R_{\text{div}} = \frac{\|\nabla \cdot u^k\|_{L^2}}{\|u^k\|_{L^2}},$$

which measures how well the continuity constraint is satisfied. We used relaxed thresholds in long mantle convection runs to balance efficiency and accuracy.

All benchmarks were discretized with Taylor–Hood (P2-P1) finite elements. Geodynamic models often use quadrilateral or hexahedral meshes, but we used structured triangular and tetrahedral meshes generated by splitting each cell along its diagonals. In two dimensions, this diagonal subdivision yields a symmetric “union-jack” pattern that preserves benchmark symmetry. This configuration provides robust velocity-pressure stability and straightforward implementation in FEniCS. Further discussion on finite element selection for geodynamic Stokes modeling can be found in (Thieulot and Bangerth, 2025). Viscosity η is represented as a finite-element field and evaluated at quadrature points during local assembly of element matrices.

To evaluate accuracy, we report the relative L^2 errors:

$$E_u = \frac{\|u - u_{\text{exact}}\|_{L^2}}{\|u_{\text{exact}}\|_{L^2}}, \quad E_p = \frac{\|p - p_{\text{exact}}\|_{L^2}}{\|p_{\text{exact}}\|_{L^2}}.$$

3.3 Baseline comparison of Uzawa variants

We first establish a baseline solver among the three Uzawa variants: standard Uzawa (S-U), adaptive Uzawa (A-U), and conjugate-directions Uzawa (CD-U). For this comparison, we use two complementary tests. The ABC flow with constant viscosity assesses convergence behavior in an ideal setting and is used to compare S-U, A-U, and CD-U directly. The SolCx problem with a viscosity contrast of 10^6 provides a stress test under strong heterogeneity; here we focus on CD-U and contrast three residual formulations – vector (algebraic), L^2 inner-product, and the viscosity-weighted CD-U- η – to examine robustness. In both tests, we track the relative L^2 errors in velocity and pressure and the normalized divergence residual over iterations, and select the most effective configuration for the remainder of the paper.

3.3.1 ABC Flow (smooth analytical solution)

The ABC (Arnold–Beltrami–Childress) flow (Zhao et al., 1993) is used as a smooth manufactured benchmark to compare the three Uzawa variants (S-U, A-U, CD-U) and to select a baseline solver. With constant viscosity ($\eta = 1.0$), all residual formulations reduce to the trivial case $w^k = q^k$. This setting isolates the algorithmic differences among the variants without interference from viscosity contrasts.

The exact velocity and pressure fields are

$$u(x, y, z) = \begin{pmatrix} \sin(\pi z) + \cos(\pi y) \\ \sin(\pi x) + \cos(\pi z) \\ \sin(\pi y) + \cos(\pi x) \end{pmatrix},$$

$$p(x, y, z) = \sin(\pi x) \cos(\pi y) \cos(\pi z).$$

Figure 1 shows the relative velocity and pressure errors together with the normalized continuity residual R_{div} as functions of iteration count. S-U converges very slowly and is highly sensitive to the relaxation parameter. A-U improves robustness but stagnates at higher error levels. CD-U, in contrast, converges rapidly and monotonically for all quantities. These results support choosing CD-U as the baseline solver for the following benchmarks.

3.3.2 SolCx (viscosity contrast benchmark)

The SolCx problem (Zhong, 1996) is a standard two-dimensional benchmark for testing solver performance under strong viscosity contrasts. It is posed on $\Omega = [0, 1]^2$ with a discontinuous viscosity,

$$\eta(x, y) = \begin{cases} 1, & x \leq 0.5, \\ 10^6, & x > 0.5, \end{cases}$$

and a density field $\rho(x, y) = \sin(\pi y) \cos(\pi x)$. A gravitational acceleration of unit magnitude ($|g| = 1$) was applied in the vertical direction, and reference solutions were taken from Underworld (Mansour et al., 2020; Moresi et al., 2007).

This benchmark poses a challenging test for Uzawa-type solvers. Here we compare three variants of the Conjugate-Directions Uzawa method. The first is the discrete residual update (Algorithm 3), where the residual is updated in purely algebraic (vector) form. The second is the L^2 inner-product formulation, in which the residual is defined variationally as in Eq. (1), and the update coefficients are computed using the inner-product relations in Eq. (4). Finally, we consider the viscosity-weighted extension CD-U- η (Algorithm 4), which

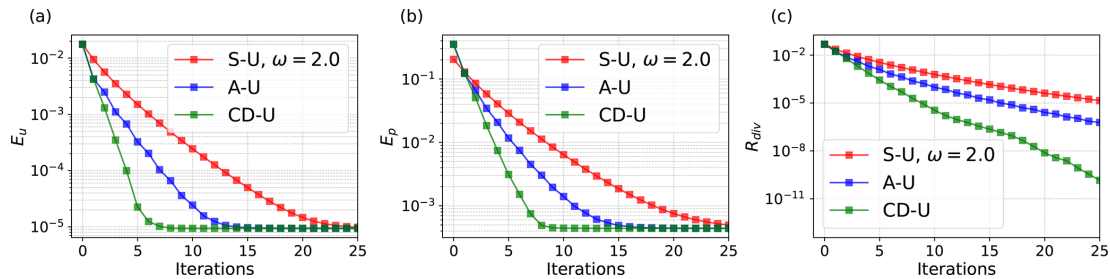


Figure 1. ABC flow benchmark on a $32 \times 32 \times 32$ mesh with P2-P1 elements. **(a)** Relative velocity error E_u , **(b)** relative pressure error E_p , and **(c)** normalized continuity residual R_{div} as functions of iteration count for the S-U, A-U, and CD-U algorithms.

realizes a preconditioned conjugate-gradient method on the Schur complement with a physically motivated η -weighted mass matrix.

Figure 2 shows that the discrete CD-U update converges slowly and stagnates at high pressure error. The L^2 inner-product version achieves faster convergence but with oscillations in R_{div} . The η -weighted variant provides the most rapid initial reduction of velocity and pressure errors and reaches much lower R_{div} within a few iterations. At later steps, R_{div} shows a mild increase. This occurs when *conjugacy* – the orthogonality of successive search directions in the preconditioned Schur complement inner product – degrades due to inexact inner solves for $Kz^k = Gd^k$ and $M_\eta w^k = q^k$. Loss of conjugacy means that a new step partly points back toward earlier directions. Progress measured in the physical L^2 divergence can be reversed locally, even though the preconditioned CG energy still decreases.

In our numerical experiments, we found that restarts or periodic reinitialization could alleviate this effect. We did not apply them here because the iteration counts are not large enough to require it. Subsequent benchmarks remain stable without reinitialization. The small increase in R_{div} is a characteristic of the baseline η -weighted formulation rather than a practical limitation.

3.3.3 Multisinker benchmark

The multisinker problem consists of multiple high-viscosity circular inclusions embedded in a low-viscosity medium, driven downward by a gravity-like body force. Unlike the SolCx benchmark with its single planar viscosity discontinuity, the multisinker configuration produces localized viscosity jumps at multiple locations, generating irregular and multiscale flow fields representative of geodynamic settings such as subducting slabs or cratonic roots. We follow the problem setup described in (Rudi et al., 2017).

We consider this problem in two dimensions on the domain $\Omega = [0, 1]^2$ with 8 high-viscosity sinkers centered at prescribed locations $\mathbf{c}_i$, $i = 1, \dots, 8$. The viscosity field is defined as

$$\eta(\mathbf{x}) = (\eta_{\text{max}} - \eta_{\text{min}})(1 - \chi_8(\mathbf{x})) + \eta_{\text{min}},$$

where the indicator function χ_8 accumulates the sinker regions through a product of smoothed Gaussian functions,

$$\chi_8(\mathbf{x}) = \prod_{i=1}^8 \left[1 - \exp\left(-\delta \max(0, |\mathbf{c}_i - \mathbf{x}| - \omega)^2\right) \right],$$

with $\delta = 200$ controlling the sharpness of the viscosity transition and $\omega = 0.05$ the radius of each sinker. Inside the sinkers $\chi_8 \approx 0$ and $\eta \approx \eta_{\text{max}}$, while in the background $\chi_8 \approx 1$ and $\eta \approx \eta_{\text{min}}$. The dynamic ratio $\text{DR}(\eta) = \eta_{\text{max}}/\eta_{\text{min}}$ was varied from 10^2 to 10^{10} , with $\eta_{\text{min}} = \text{DR}^{-1/2}$ and $\eta_{\text{max}} = \text{DR}^{1/2}$. The body force $\mathbf{f}(\mathbf{x}) = (0, \beta(\chi_8(\mathbf{x}) - 1))$ with $\beta = 10^5$ acts only inside the sinkers, driving them downward through the lighter, low-viscosity medium. The domain was discretized with 256×256 triangular P2-P1 Taylor–Hood elements, and reference solutions were computed using the MUMPS direct solver. Figure 3 shows the viscosity field and the vertical component of the body force for $\text{DR}(\eta) = 10^{10}$.

Since this problem has no analytical solution, convergence was evaluated by measuring relative errors in velocity and pressure against the MUMPS reference solution. Figure 4 shows the convergence history of the CD-U- η solver across dynamic ratios from 10^2 to 10^{10} .

Figure 4 shows that at low dynamic ratios ($\text{DR} = 10^2$), the solver converges rapidly. As the dynamic ratio increases, an extended plateau develops in the early iterations before convergence accelerates. This is characteristic of conjugate gradient methods with a poor initial guess. In all experiments, the initial pressure was set to zero, which is far from the true solution for high-viscosity-contrast problems. A better initial pressure estimate could reduce this plateau, but developing such strategies is beyond the scope of this work. For $\text{DR} \geq 10^6$, the convergence curves nearly overlap, indicating that the solver becomes insensitive to further increases in viscosity contrast. The pressure error consistently lags behind the velocity error during the initial phase. The adaptive AAR strategy was triggered only for $\text{DR} = 10^8$ and 10^{10} near the final iterations. For this benchmark, stagnation was detected when $|R_{\text{div}}^{k+1} - R_{\text{div}}^k| < 10^{-8}$ for two consecutive iterations, with a reduction factor of 0.01 and pointwise lower bound $\epsilon = 10^{-8}$.

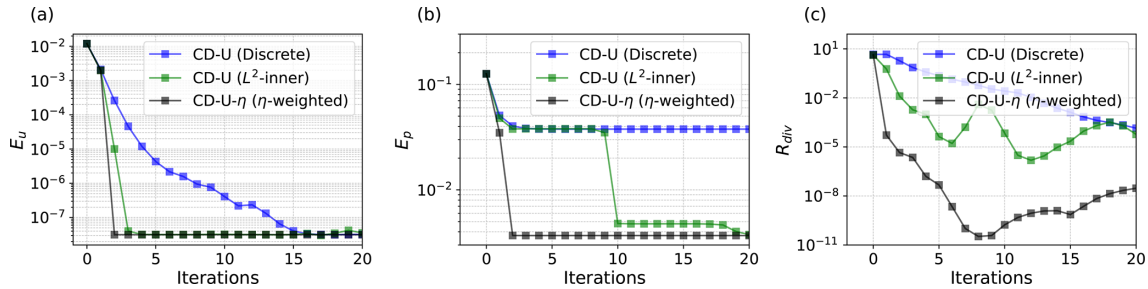


Figure 2. SolCx benchmark with P2-P1 elements. (a) Relative velocity error E_u , (b) relative pressure error E_p , and (c) normalized divergence residual R_{div} as functions of iteration count for CD-U with discrete update, L^2 inner-product update, and η -weighted update.

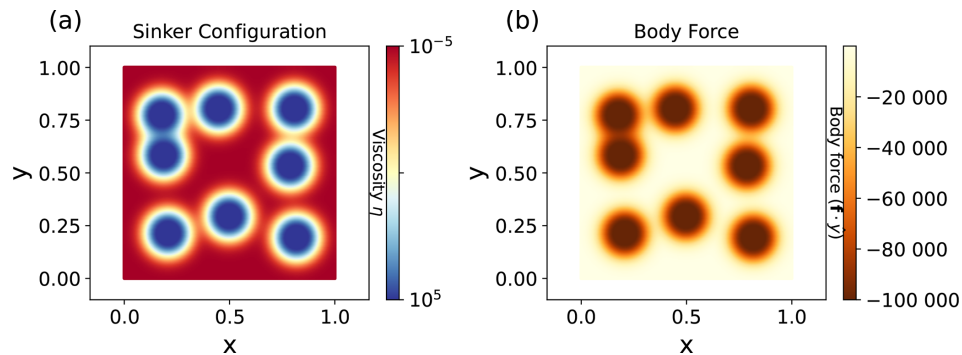


Figure 3. Multisinker benchmark configuration with 8 sinks and $DR(\eta) = 10^{10}$. (a) Viscosity field and (b) vertical component of the body force.

3.4 Post-processing strategies for incompressible problems

We next examine the effect of the post-processing correction in two incompressible tests: linear mantle convection with variable viscosity and the block sinking problem with sharp density and viscosity contrasts. In both tests the mesh, boundary conditions, and time stepping are kept identical; the only change is in the Stokes solver. Results from CD-U- η are compared with those from CD-U- η combined with post-processing. Reference solutions are obtained with the direct LU solver MUMPS (Amestoy et al., 2000), which solves the discrete system to near machine precision.

Performance is evaluated by the relative L^2 errors of velocity and scalar fields (temperature, composition, or density) and by the normalized continuity residual R_{div} . A breakdown of solver cost is also reported, showing that the post-processing step adds little to the total runtime, while momentum solves remain the dominant cost.

3.4.1 Linear mantle convection

The linear mantle convection benchmark was solved on $\Omega = [0, 2] \times [0, 1]$ using a 320×160 structured grid, subdivided into triangular P2-P1 Taylor–Hood elements for velocity and pressure. Viscosity followed a depth- and temperature-dependent exponential law (Blankenbach et al.,

1989),

$$\eta(x, y) = \exp\left(-b \frac{T}{\Delta T} + c(1 - y)\right), \quad b = \log(2.5),$$

$$c = \log(2.0).$$

The body force in the Stokes equations was given by thermal and compositional buoyancy,

$$\mathbf{f} = (Ra_T T + Ra_\phi \phi) \hat{\mathbf{g}}, \quad \hat{\mathbf{g}} = (0, -1),$$

with Rayleigh numbers

$$Ra_T = Ra_\phi = 10^6.$$

Initial temperature was prescribed from boundary-layer theory (van Keken, 1997), and the composition field ϕ was initialized as a dense basal layer (Tan and Gurnis, 2005). Boundary conditions were no-slip at the bottom and free-slip elsewhere for velocity; $T = 1$ at the bottom and $T = 0$ at the top with insulating sidewalls; and $\phi = 1$ at the bottom with homogeneous Neumann elsewhere. The transport equations for temperature T and composition ϕ are

$$\partial_t T + \mathbf{u} \cdot \nabla T - \nabla \cdot (\kappa \nabla T) = 0, \quad \partial_t \phi + \mathbf{u} \cdot \nabla \phi = 0,$$

where $\kappa = k/(\rho c_p)$ denotes the thermal diffusivity, with k the thermal conductivity, ρ the density, and c_p the specific heat

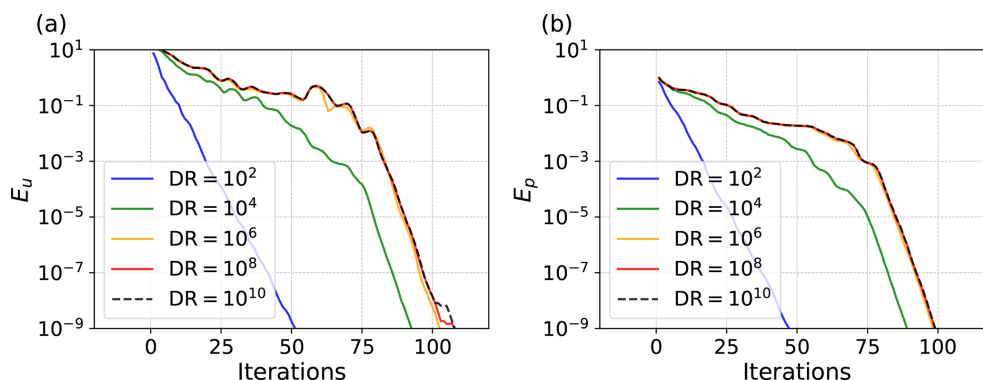


Figure 4. Multisinker benchmark with P2-P1 elements on a 256×256 mesh. Convergence of the CD-U- η solver across dynamic ratios from 10^2 to 10^{10} . (a) Relative velocity error E_u and (b) relative pressure error E_p as functions of iteration count.

capacity. For this benchmark, we set $\kappa = 1$. Both fields were discretized with P1 discontinuous Galerkin elements using upwind fluxes. Time integration employed the second-order BDF2 scheme, initialized by backward Euler. The Stokes system was solved at each step with CD-U- η or CD-U- η + Post-processing. Iterations were stopped once the normalized continuity residual R_{div} fell below 10^{-3} .

Figure 5 shows that the post-processing step improves accuracy in all fields. Velocity errors remain smaller at every stage, and the normalized continuity residual R_{div} is consistently lower. Since the composition and temperature fields are advected by the velocity, they also benefit and show clear error reductions. Over 5000 steps, these corrections accumulate, yielding higher long-term accuracy compared with the baseline solver.

Figure 6 shows composition and temperature errors at the final step. CD-U- η alone produces noticeable errors near the interface, especially in the composition field. With post-processing, these localized errors are reduced, and the solution is closer to the reference from the direct LU solver MUMPS, which is effectively exact for the discrete system. This demonstrates that post-processing improves accuracy and aligns the iterative solution more closely with that of a direct solver.

Table 1 summarizes the solver cost for CD-U and CD-U- η with post-processing. The viscosity-weighted variant reduces the average number of momentum solves per step (2.40 vs. 3.05) under the same tolerance, leading to an overall runtime reduction of about 10 %. The extra mass-matrix operations in CD-U- η (two solves per iteration instead of one) are inexpensive, contributing only about 1 %–2 % of the total cost. Similarly, the post-processing corrections, consisting of a Poisson solve and velocity update, account for only a small fraction of the runtime compared with the dominant momentum solves. The main benefit of CD-U- η is therefore the reduction in momentum iterations, while the additional scalar operations have a negligible impact on efficiency.

3.4.2 Block sinking problem

The block sinking benchmark (Gerya, 2019) is a two-material Stokes flow test designed to evaluate solver robustness under sharp density and viscosity contrasts. A dense, low-viscosity block sinks into a lighter and more viscous fluid. The interface was represented by a DG0 level-set field (ϕ), positive inside the block, negative outside, and zero on the interface. The level-set was advected in time without reinitialization.

The domain $\Omega = [0, 1]^2$ was discretized with 256×256 triangular P2-P1 Taylor–Hood elements for velocity and pressure. The block initially occupied $0.4 \leq x \leq 0.6$, $0.7 \leq y \leq 0.9$. Material properties showed strong contrasts: density 4200 vs. 2800, viscosity 0.1 vs. 100. All sides had free-slip boundaries, and the pressure was initialized uniformly. The body force was buoyancy,

$$\mathbf{f} = \rho(x, y) \hat{\mathbf{g}}, \quad \hat{\mathbf{g}} = (0, -1),$$

with ρ defined by the block geometry. The level-set was advanced using a Crank–Nicolson scheme.

Solver iterations used a relaxed stopping condition $R_{\text{div}} < 0.2$ due to the strong viscosity contrasts and interface-driven complexity. This avoided excessive momentum solves under a strict tolerance while still allowing an effective test of the post-processing step. The reference solution was computed with the direct LU solver MUMPS.

Figure 7 shows errors in velocity, density, level-set, and R_{div} for the block sinking problem. Post-processing decreases both velocity and density errors and suppresses divergence. The irregular pattern in the density error comes from discontinuities along the block interface introduced by the DG0 scheme. The level-set error profile appears smoother. We observed that this reflects the nearly uniform field inside and outside the block. Post-processing helps maintain mass balance and improves agreement with the direct solver.

Figure 8 shows the density and level-set fields at the final step. Large density errors indicate loss of mass conservation,

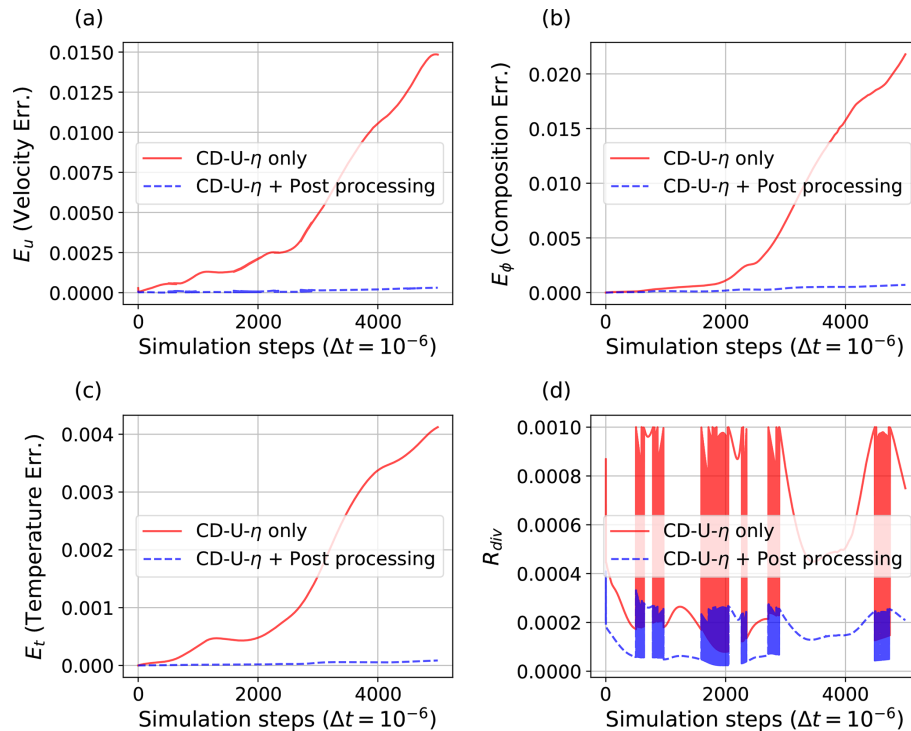


Figure 5. Mantle convection benchmark over 5000 time steps ($\Delta t = 10^{-6}$). Comparison of CD-U- η with and without projection post-processing: (a) relative velocity error, (b) composition error, (c) temperature error, and (d) normalized divergence residual R_{div} .

Table 1. Cost breakdown for 5000 steps ($\Delta t = 10^{-6}$, tolerance 10^{-3} , 32 cores). Only solver execution time is included. Both CD-U and CD-U- η are shown with post-processing.

Step	CD-U (Total 2036 s)			CD-U- η (Total 1849 s)		
	Total Solves	% Time	Solves/step	Total Solves	% Time	Solves/step
Momentum solves	15 264	73.2 %	3.05	11 998	69.1 %	2.40
Mass-matrix solves (in Uzawa)	15 264	0.5 %	3.05	23 996	1.4 %	4.80
Post-processing (Poisson)	5000	2.6 %	1.00	5000	3.0 %	1.00
Post-processing (velocity)	5000	2.1 %	1.00	5000	2.3 %	1.00
Composition advection	5000	3.3 %	1.00	5000	3.8 %	1.00
Temperature advection	5000	18.3 %	1.00	5000	20.4 %	1.00

while post-processing reduces these errors and brings the results closer to the reference solution.

Table 2 compares the solver cost for CD-U and CD-U- η with post-processing in the block sinking problem. The viscosity-weighted variant reduces the average number of momentum solves per step from 22.2 to 8.35, lowering the total runtime from 4168 to 1735 s. The extra mass-matrix solves in CD-U- η are inexpensive, contributing only 1 %–2 % of the total. Post-processing corrections and level-set advection also remain minor. The main benefit of CD-U- η is therefore the smaller iteration count in the momentum equations, while the additional scalar operations add little overhead.

3.5 Compressible mantle convection

We next consider mantle convection under the Anelastic Liquid Approximation (ALA) (Schubert et al., 2001; King et al., 2008), which incorporates depth-dependent density while filtering out fast acoustic modes. This benchmark follows (King et al., 2008) with Rayleigh number $Ra = 10^5$ and dissipation number $Di = 0.5$. This case extends the projection-based correction to compressible formulations, where mass conservation is weighted by the background density profile.

Under ALA, mass conservation is expressed as

$$\nabla \cdot (\bar{\rho} \mathbf{u}) = 0, \quad \bar{\rho}(z) = \rho_0 \exp\left(\frac{Di(1-z)}{\gamma_0}\right).$$

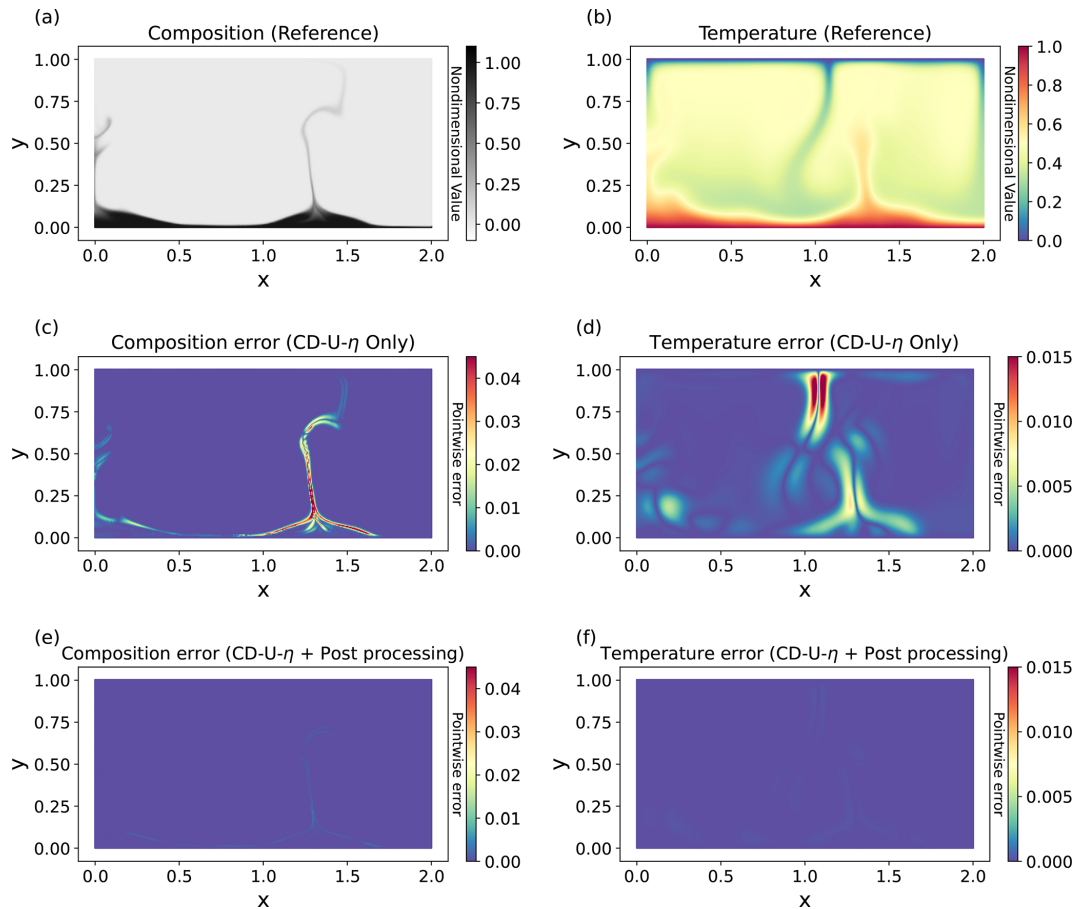


Figure 6. Mantle convection benchmark at time step 5000. Reference fields: (a) composition and (b) temperature. Absolute errors for CD-U- η : (c) composition and (d) temperature. Absolute errors for CD-U- η with projection: (e) composition and (f) temperature.

Table 2. Solver cost breakdown for the block sinking simulation (1500 steps, $\Delta t = 5 \times 10^{-6}$, tolerance 0.2, 32 cores). Only solver execution time is included.

Step	CD-U (Total 4168 s)			CD-U- η (Total 1735 s)		
	Total Solves	% Time	Solves/step	Total Solves	% Time	Solves/step
Momentum solves	33 359	98.5 %	22.2	12 523	97.2 %	8.35
Mass-matrix solves (in Uzawa)	33 359	0.8 %	22.2	25 046	1.4 %	16.7
Post-processing (Poisson)	1500	0.3 %	1.00	1500	0.7 %	1.00
Post-processing (velocity)	1500	0.3 %	1.00	1500	0.8 %	1.00
Level-set advection	1500	0.1 %	1.00	1500	0.2 %	1.00

The momentum equation is

$$\nabla \cdot \boldsymbol{\tau}(\mathbf{u}) - \nabla p' + Ra \bar{\rho} \alpha T \hat{\mathbf{k}} - \left(\frac{Di}{\gamma_0} \frac{c_p}{c_v} \right) \bar{\rho} \chi_T p' \hat{\mathbf{k}} = \mathbf{0},$$

where

$$\boldsymbol{\tau}(\mathbf{u}) = 2\mu \boldsymbol{\varepsilon}(\mathbf{u}) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u}) \mathbf{I}, \quad \boldsymbol{\varepsilon}(\mathbf{u}) = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^\top).$$

The temperature equation, including advection, diffusion, adiabatic heating, and viscous dissipation, is

$$\bar{\rho} c_p \partial_t T + \bar{\rho} c_p \mathbf{u} \cdot \nabla T - \nabla \cdot (\kappa \nabla (\bar{T} + T)) + \alpha \bar{\rho} Di u_y T - \frac{Di}{Ra} \boldsymbol{\sigma}(\mathbf{u}) : \nabla \mathbf{u} = 0,$$

with the background profile

$$\bar{T}(z) = T_0 \exp(Di(1 - z)).$$

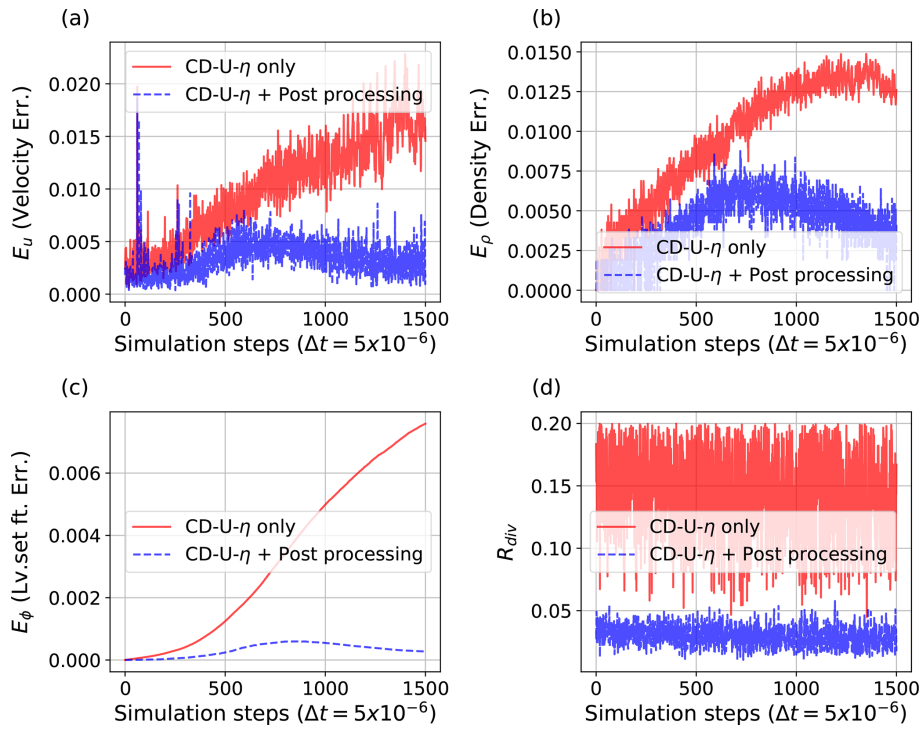


Figure 7. Block sinking benchmark over 1500 time steps ($\Delta t = 5 \times 10^{-6}$). Comparison of CD-U- η with and without projection: (a) relative velocity error E_u , (b) density error E_ρ , (c) level-set error E_ϕ , and (d) normalized divergence residual R_{div} .

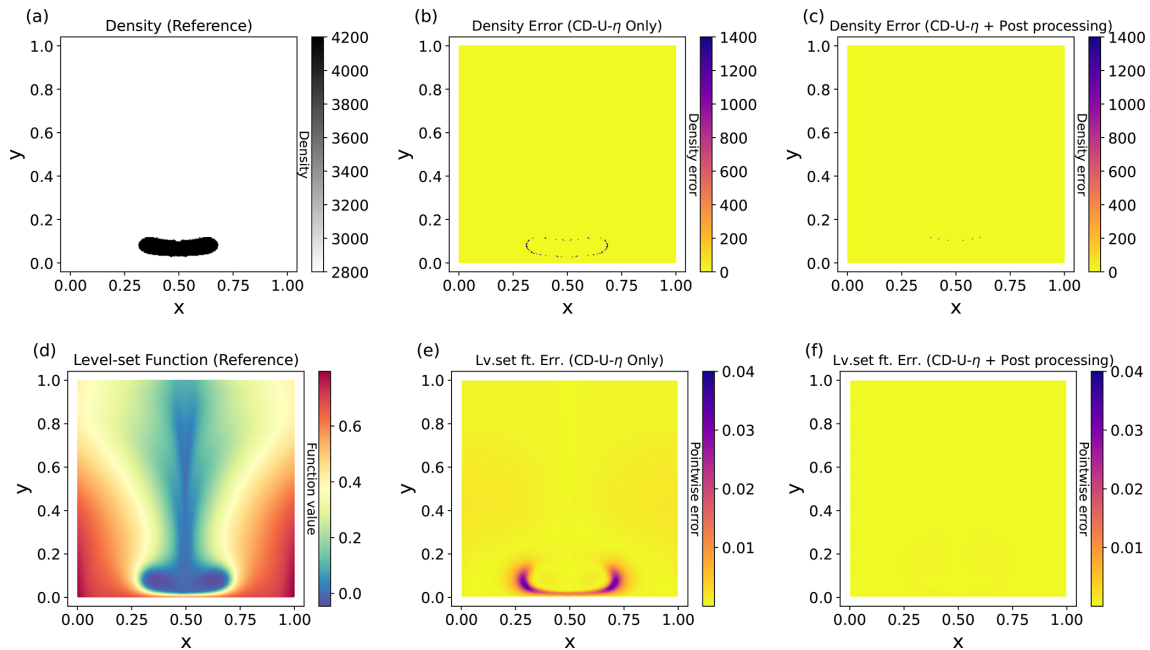


Figure 8. Block sinking benchmark at the final step (1500, $\Delta t = 5 \times 10^{-6}$). Density field: (a) reference (MUMPS), (b) CD-U- η , and (c) CD-U- η with projection. Level-set field: (d) reference (MUMPS), (e) CD-U- η , and (f) CD-U- η with projection.

Time discretization used a second-order Crank–Nicolson scheme. Since viscosity is constant, CD-U and CD-U- η coincide; the focus here is therefore on testing the robustness of the projection-based correction under compressible mass conservation.

To enforce the weighted continuity constraint after each Uzawa iteration, a $\bar{\rho}$ -weighted velocity correction was applied. Given an intermediate velocity $\hat{\mathbf{u}}$, we solved

$$-\nabla \cdot (\bar{\rho} \nabla \phi) = \nabla \cdot (\bar{\rho} \hat{\mathbf{u}}), \quad \mathbf{u} = \hat{\mathbf{u}} + \nabla \phi,$$

ensuring $\nabla \cdot (\bar{\rho} \mathbf{u}) = 0$ at the discrete level.

The computational domain $\Omega = [0, 1]^2$ was discretized into 160×160 triangular Taylor–Hood (P2-P1) elements for velocity and pressure, and P1 discontinuous Galerkin elements for temperature. Viscosity was constant ($\eta = 1$). Free-slip boundaries were imposed on the horizontal sides and no-slip on the vertical sides. The temperature was fixed at the bottom ($T \approx 0.09$) and top ($T = 0$), with insulated side-walls. Initial conditions included sinusoidal perturbations of the background temperature, and pressure was initialized to zero. The Uzawa method with conjugate directions (CD-U) was used as the baseline solver, with convergence tolerance $R_{\text{div}}^{\bar{\rho}} < 10^{-3}$, and was compared against CD-U with projection-based correction. Reference solutions were computed using the direct solver MUMPS.

Figure 9 shows that with constant viscosity the CD-U solver alone is already robust, but the $\bar{\rho}$ -weighted correction further reduces errors in velocity and temperature. Figure 10 confirms improved accuracy near steep thermal gradients.

In terms of cost, the constant-viscosity ALA setup required only about 2–3 momentum solves per step on average, making it considerably cheaper than variable-viscosity cases. The $\bar{\rho}$ -weighted correction added less than 20 % to the runtime, confirming that its accuracy benefits come at modest overhead.

4 Conclusions

We revisited Uzawa-type algorithms for Stokes equations in geodynamics and propose two enhancements. The residual calculation was reformulated in variational form, which provides a natural preconditioner. We then extended this to a viscosity weighting version (CD-U- η). This can be interpreted as a preconditioned conjugate gradient method for the Schur complement. The second enhancement applies a Helmholtz–Hodge projection as post-processing to the velocity field. This enforces the constraint more accurately. Through several benchmark tests, we confirmed that these two approaches help the algorithm obtain more accurate solutions.

Our benchmarks include smooth analytical flows, high-contrast SolCx, SolKz, multisinker, linear mantle convection, block sinking with multiphase interfaces, and compressible mantle convection under ALA. CD-U- η converged faster and stayed robust under large viscosity contrasts. The projection step improved constraint satisfaction and enhanced long-term accuracy. In time-dependent simulations, small corrections at each step accumulated and substantially improved transported fields such as temperature, composition, and density.

In our benchmark tests, the additional cost of L^2 residual and projection corrections was modest – less than 5 % of total solver time. Momentum solves remained the dominant component. The enhancements do not significantly change the overall cost. As the problem size increases, the relative cost of these corrections should decrease further since momentum solves become more dominant. Additionally, the viscosity-weighted variant helps reduce momentum iterations, further improving computational efficiency.

It is important to note that the projection step only corrects the velocity field and does not directly enhance pressure convergence. This highlights an inherent limitation of projection-based strategies, which are designed to enforce incompressibility. Since pressure plays a critical role in many geodynamic applications, its convergence remains governed by the underlying Uzawa iteration. Our focus here was to improve velocity and advected fields. Improving pressure convergence represents an important direction for future research.

In conclusion, projection-enhanced Uzawa solvers provide an algorithmic perspective on Schur complement preconditioning grounded in finite element variational principles. Since the entire formulation relies on inner products and variational forms, it naturally fits within general-purpose FEM libraries coupled with PETSc, and works across different platforms. The method performs well for both incompressible and compressible mantle convection problems with strong rheological complexity. These findings suggest that projection-enhanced Uzawa solvers can serve as a useful baseline for future work in geodynamic modeling.

Appendix A: Mesh refinement study using the SolKz benchmark

One of the most important properties of a numerical method is its behavior under mesh refinement, which verifies that the discrete solution converges to the continuous solution at the expected rate. All experiments in the main body use a fixed discretization, so we present here a systematic mesh refinement study.

We use the SolKz benchmark (Duretz et al., 2011), a two-dimensional Stokes problem on the unit square $\Omega = [0, 1]^2$ with a smooth, exponentially varying viscosity $\eta(y) = \exp(cy)$ and a dynamic ratio of 10^6 . The body force

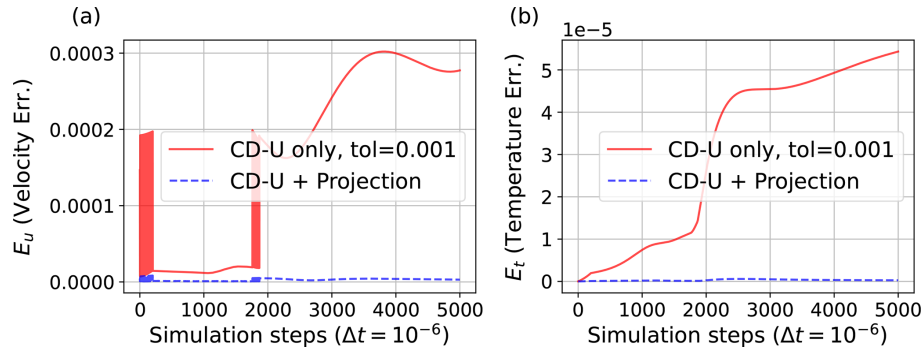


Figure 9. Compressible mantle convection under ALA over 5000 time steps ($\Delta t = 10^{-6}$). (a) Relative velocity error E_u and (b) relative temperature error E_t , comparing CD-U- η with and without $\bar{\rho}$ -weighted velocity correction.

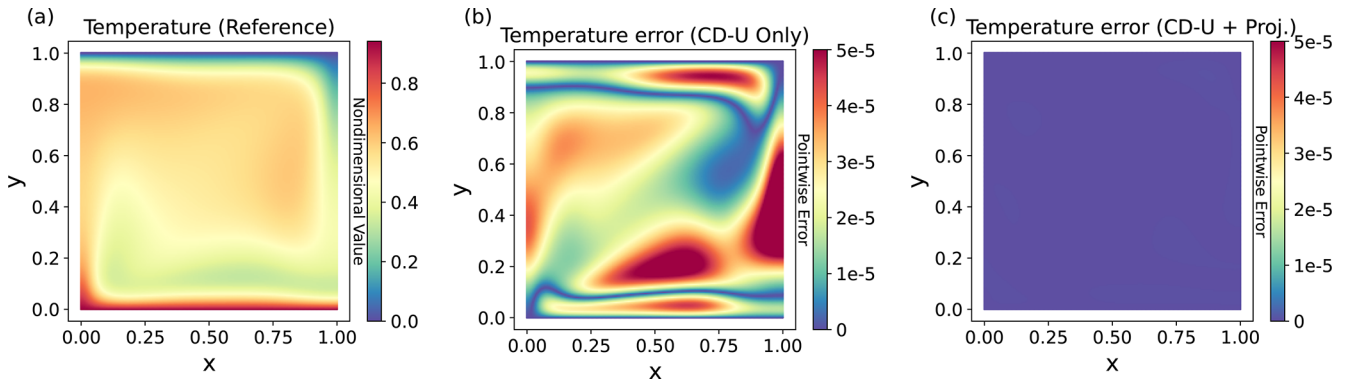


Figure 10. Compressible mantle convection at the final step (5000, $\Delta t = 10^{-6}$). (a) Reference temperature field (MUMPS), (b) absolute temperature error for CD-U- η , and (c) absolute temperature error for CD-U- η with $\bar{\rho}$ -weighted correction.

is chosen so that the analytical solution is known. We chose SolKz over the SolCx benchmark presented in Sect. 3 because the discontinuous viscosity in SolCx reduces solution regularity and prevents observation of optimal convergence rates with P2-P1 Taylor–Hood elements. The domain was discretized with structured triangular meshes from 32×32 to 1024×1024 using P2-P1 Taylor–Hood elements, and reference solutions were obtained from the Underworld software (Mansour et al., 2020).

Table A1 reports the relative errors and the corresponding convergence orders. The method achieves the optimal convergence rates for P2-P1 Taylor–Hood elements: third order for velocity and second order for pressure. These rates are maintained consistently across all resolutions.

For resolutions up to 256×256 , the solver converged with $\text{AAR} = 1.0$ without requiring adaptation. At 512×512 , AAR adjustment occurred occasionally. At 1024×1024 , the adaptive AAR strategy described in Sect. 2.2 was essential for convergence. For this benchmark, stagnation was detected when $|R_{\text{div}}^{k+1} - R_{\text{div}}^k| < 10^{-10}$ for two consecutive iterations, and the AAR parameter was reduced by a factor of 0.1 at each trigger. The pointwise lower bound was set to $\epsilon = 10^{-6}$.

Table A1. Mesh refinement results for the SolKz benchmark using CD-U- η .

Resolution	E_u	Order	E_p	Order
32×32	1.877×10^{-3}	–	1.591×10^{-2}	–
64×64	2.316×10^{-4}	3.02	3.970×10^{-3}	2.00
128×128	2.888×10^{-5}	3.00	9.872×10^{-4}	2.01
256×256	3.610×10^{-6}	3.00	2.401×10^{-4}	2.04
512×512	4.514×10^{-7}	3.00	5.845×10^{-5}	2.04
1024×1024	5.648×10^{-8}	3.00	1.490×10^{-5}	1.97

Appendix B: Projection post-processing vs. additional iterations

In time-dependent geodynamic simulations, the Stokes system must be solved at every time step. Iterating to tight convergence tolerances at each step is computationally expensive, since the dominant cost is the momentum solve. In practice, relaxed tolerances are often used to reduce runtime. However, the divergence residual that remains after each solve accumulates over thousands of time steps, degrading the accuracy of advected fields such as temperature, composition, and density. The projection post-processing in-

troduced in Sect. 2.3 offers a potential remedy by enforcing the divergence-free constraint at the cost of only one additional Poisson solve per time step. An alternative strategy is simply to perform additional Uzawa iterations per step. To compare these two approaches in terms of cost and accuracy, we tested both on the mantle convection and block sinking benchmarks from Sect. 3.

In the results below, CD-U- η (n iter) denotes n iterations of the CD-U- η solver per time step, and +Proj indicates projection post-processing. Numbers in parentheses show the projection overhead.

Table B1 shows results for the mantle convection benchmark over 5000 steps with $\Delta t = 10^{-6}$.

Table B2 shows results for the block sinking benchmark over 1500 steps with $\Delta t = 5 \times 10^{-6}$. For this benchmark, the initial step iterates to tolerance, while subsequent steps use fixed iteration counts with the previous solution as an initial guess.

Both benchmarks show that projection adds only a small fraction of the total runtime while substantially improving accuracy. This efficiency comes from the cost difference between the two strategies: each additional Uzawa iteration requires solving the momentum equation, which is the most expensive component, whereas projection requires only one Poisson solve per time step. Moreover, since projection operates as a post-processing step, it is not limited to the Uzawa framework and can be combined with any iterative Stokes solver.

Table B1. Cost and accuracy comparison for the mantle convection benchmark.

	Uzawa time	E_u	R_{div}	E_T
CD-U- η (2iter)	1277 s	0.0177	0.0138	0.00447
CD-U- η (2iter)+Proj	1355 s (+78 s)	0.000633	0.00144	0.000120
CD-U- η (3iter)	1704 s	0.00775	0.00975	0.00187
CD-U- η (3iter)+Proj	1783 s (+79 s)	0.0000970	0.000263	0.0000178

Table B2. Cost and accuracy comparison for the block sinking benchmark.

	Uzawa time	E_u	R_{div}	E_ρ
CD-U- η (5iter)	1253 s	0.0449	0.0234	0.0236
CD-U- η (5iter)+Proj	1278 s (+25 s)	0.00200	0.00384	0.00127
CD-U- η (6iter)	1463 s	0.0336	0.0198	0.0160
CD-U- η (6iter)+Proj	1485 s (+22 s)	0.00875	0.00608	0.00113

Appendix C: Three-dimensional thermochemical convection

To verify that the CD-U- η algorithm and the projection post-processing operate correctly in three dimensions, we extend the two-dimensional thermochemical convection benchmark of Sect. 3.4.1 by adding a unit-depth lateral dimension. This test also demonstrates the algorithm with Mini elements (P1+Bubble/P1), which satisfy the LBB stability condition independently of the Taylor–Hood elements used in all two-dimensional experiments. Since Mini elements carry fewer degrees of freedom per element than P2-P1 Taylor–Hood elements, they allow finer meshes to be used within the same computational budget, which is particularly advantageous in three dimensions.

The physical setup follows Sect. 3.4.1, extended to the three-dimensional domain $[0, 2] \times [0, 1] \times [0, 1]$ discretized with a $100 \times 50 \times 50$ structured tetrahedral mesh. Free-slip boundary conditions were applied on the additional sidewalls ($y = 0$ and $y = 1$), with insulating conditions for temperature and homogeneous Neumann for composition. All other parameters, including the Rayleigh numbers, viscosity law, initial conditions, and transport discretization, are identical to Sect. 3.4.1. We adopted a fixed time step of $\Delta t = 10^{-5}$, ten times larger than in the two-dimensional case, since the purpose of this test is to verify the algorithmic behavior rather than to assess solution accuracy, and the BDF2 time integration remains stable at this step size. The solution was advanced to the same target time $t = 5 \times 10^{-3}$. The Stokes system was solved at each step with CD-U- η , with and without projection post-processing. Iterations were stopped once the normalized continuity residual R_{div} fell below 10^{-3} .

Since a direct solver reference solution is not feasible at this mesh resolution in three dimensions, the comparison focuses on the effect of the projection strategy rather than on accuracy relative to a reference.

Figure C1 shows the composition and temperature fields at $t = 5 \times 10^{-3}$, where thermal plumes and entrainment of the basal dense layer are consistent with the two-dimensional results.

Figure C2a and b show that the projection suppresses the divergence error throughout the domain, with the localized concentrations near the plume regions in (a) largely eliminated in (b). The R_{div} history in Fig. C2c confirms that this reduction is maintained consistently across all time steps, with the projection lowering R_{div} by approximately a factor of two, consistent with the two-dimensional results.

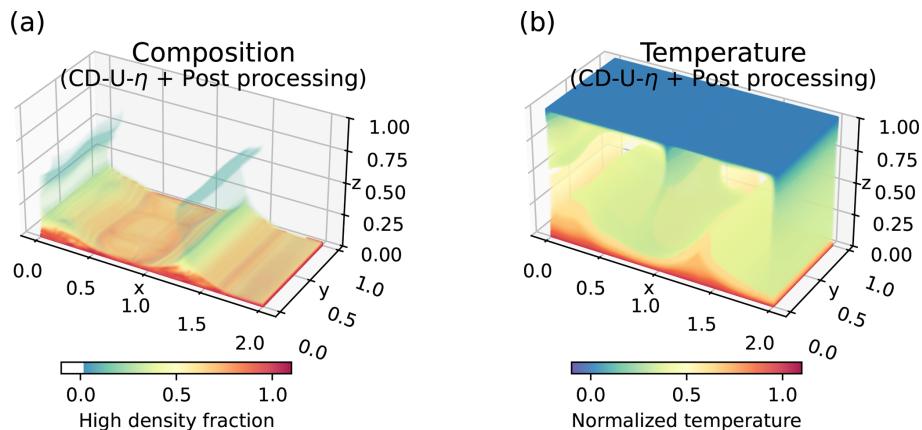


Figure C1. Three-dimensional thermochemical convection at $t = 5 \times 10^{-3}$ solved with CD-U- η + projection on a $100 \times 50 \times 50$ Mini element mesh. (a) Composition field showing entrainment of the basal dense layer. (b) Normalized temperature field.

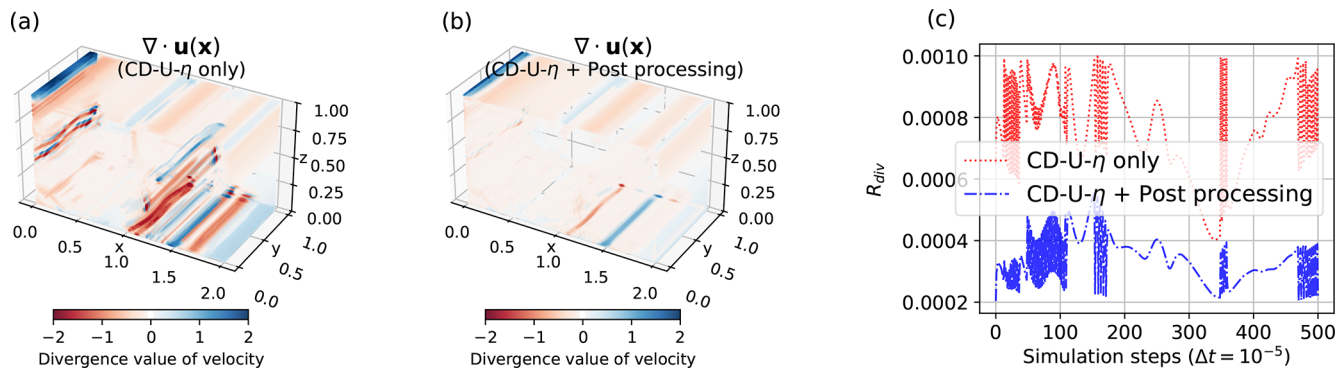


Figure C2. Spatial distribution of $\nabla \cdot \mathbf{u}$ at $t = 5 \times 10^{-3}$ for (a) CD-U- η only and (b) CD-U- η + projection. (c) Evolution of the normalized weak divergence residual R_{div} over 501 time steps for CD-U- η only and CD-U- η + projection.

Code availability. The numerical benchmarks used in this study (ABC flow, SolCx, SolKz, multisinker, mantle convection, block sinking, and compressible convection) are publicly available and documented in the cited references. All simulations were performed using the open-source finite element library FEniCS (version 2019.2.0.13) coupled with PETSc. The implementation code is available at <https://doi.org/10.5281/zenodo.22135967> (Jang et al., 2026). Specific solver configurations and benchmark parameters are described in Sect. 3.

Author contributions. D.-K. J. designed the algorithms, implemented the software, and prepared the first draft of the manuscript.

K.-M. L. carried out the numerical experiments, performed the data analysis and contributed to the review of the manuscript.

C. T. contributed substantially to shaping the core ideas of the paper through revisions and provided major input during the manuscript editing process.

W.-H. C. assisted in numerical implementation, provided feedback on data interpretation, and contributed to securing financial support.

B.-D. S. supervised the project, acquired funding, and revised the manuscript.

All authors discussed the results and contributed to the final version of the paper.

Competing interests. The contact author has declared that none of the authors has any competing interests.

Disclaimer. Publisher's note: Copernicus Publications remains neutral with regard to jurisdictional claims made in the text, published maps, institutional affiliations, or any other geographical representation in this paper. The authors bear the ultimate responsibility for providing appropriate place names. Views expressed in the text are those of the authors and do not necessarily reflect the views of the publisher.

Acknowledgements. This work was supported by the National Research Foundation of Korea (NRF) grants funded by the Korea government (no. RS-2025-00519120 to Deok-Kyu Jang, nos. RS-2025-25415913 and NRF-2022R1C2011689 to Whan-Hyuk Choi, and nos. RS-2019-NR040076, RS-2023-00256330 and RS-2025-02293161 to Byung-Dal So).

Financial support. This research has been supported by the National Research Foundation of Korea (grant nos. RS-2025-00519120, RS-2025-25415913, NRF-2022R1C2011689, RS-2019-NR040076, RS-2023-00256330, and RS-2025-02293161).

Review statement. This paper was edited by Taras Gerya and reviewed by G. Stadler and Stefan Markus Schmalholz.

References

- Alnæs, M., Blechta, J., Hake, J., Johansson, A., Kehlet, B., Logg, A., Richardson, C., Ring, J., Rognes, M. E., and Wells, G. N.: The FEniCS project version 1.5, Archive of Numerical Software, 3, <https://doi.org/10.11588/ans.2015.100.20553>, 2015.
- Amestoy, P. R., Duff, I. S., L'Excellent, J.-Y., and Koster, J.: MUMPS: a general purpose distributed memory sparse solver, in: International Workshop on Applied Parallel Computing, Springer, 121–130, https://doi.org/10.1007/3-540-70734-4_16, 2000.
- Arrow, K. J., Hurwicz, L., Uzawa, H., Chenery, H. B., Johnson, S., and Karlin, S.: Studies in linear and non-linear programming, vol. 2, Stanford University Press Stanford, 229 pp., 1958.
- Babuška, I.: The finite element method with Lagrangian multipliers, Numer. Math., 20, 179–192, <https://doi.org/10.1007/BF01436561>, 1973.
- Bai, Z.-Z. and Wang, Z.-Q.: On parameterized inexact Uzawa methods for generalized saddle point problems, Linear Algebra Appl., 428, 2900–2932, <https://doi.org/10.1016/j.laa.2008.01.018>, 2008.

- Balay, S., Abhyankar, S., Adams, M. F., Benson, S., Brown, J., Brune, P., Buschelman, K., Constantinescu, E. M., Dalcin, L., Dener, A., Eijkhout, V., Faibussowitsch, J., Gropp, W. D., Hapla, V., Isaac, T., Jolivet, P., Karpeev, D., Kaushik, D., Knepley, M. G., Kong, F., Kruger, S., May, D. A., McInnes, L. C., Mills, R. T., Mitchell, L., Munson, T., Roman, J. E., Rupp, K., Sanan, P., Sarich, J., Smith, B. F., Suh, H., Zampini, S., Zhang, H., Zhang, H., and Zhang, J.: PETSc/TAO Users Manual Revision 3.22, Tech. rep., ANL-21/39 Rev. 3.22, Argonne National Laboratory (ANL), Argonne, IL, United States, 2024.
- Bell, J. B., Colella, P., and Glaz, H. M.: A second-order projection method for the incompressible Navier-Stokes equations, *J. Comput. Phys.*, 85, 257–283, [https://doi.org/10.1016/0021-9991\(89\)90151-4](https://doi.org/10.1016/0021-9991(89)90151-4), 1989.
- Benzi, M., Golub, G. H., and Liesen, J.: Numerical solution of saddle point problems, *Acta Numer.*, 14, 1–137, <https://doi.org/10.1017/S0962492904000212>, 2005.
- Blankenbach, B., Busse, F., Christensen, U., Cserepes, L., Gunkel, D., Hansen, U., Harder, H., Jarvis, G., Koch, M., Marquart, G., Moore, D., Olson, P., Schmeling, H., and Schnaubelt, T.: A benchmark comparison for mantle convection codes, *Geophys. J. Int.*, 98, 23–38, <https://doi.org/10.1111/j.1365-246X.1989.tb05511.x>, 1989.
- Braess, D.: Finite elements: Theory, fast solvers, and applications in solid mechanics, 2nd edn., ISBN 978-0-521-01195-2, Cambridge University Press, 2001.
- Brezzi, F.: On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers, *RAIRO Anal. Numér.*, 8, 129–151, <https://doi.org/10.1051/m2an/197408R201291>, 1974.
- Brown, D. L., Cortez, R., and Minion, M. L.: Accurate projection methods for the incompressible Navier-Stokes equations, *J. Comput. Phys.*, 168, 464–499, <https://doi.org/10.1006/jcph.2001.6715>, 2001.
- Brune, S., Popov, A. A., and Sobolev, S. V.: Modeling suggests that oblique extension facilitates rifting and continental break-up, *J. Geophys. Res.-Sol. Ea.*, 117, <https://doi.org/10.1029/2011JB008860>, 2012.
- Cahouet, J. and Chabard, J.-P.: Some fast 3D finite element solvers for the generalized Stokes problem, *Int. J. Numer. Meth. Fl.*, 8, 869–895, <https://doi.org/10.1002/fld.1650080802>, 1988.
- Chen, X.: On preconditioned Uzawa methods and SOR methods for saddle-point problems, *J. Computat. Appl. Math.*, 100, 207–224, [https://doi.org/10.1016/S0377-0427\(98\)00197-6](https://doi.org/10.1016/S0377-0427(98)00197-6), 1998.
- Chorin, A. J.: The numerical solution of the Navier-Stokes equations for an incompressible fluid, *B. Am. Math. Soc.*, 73, 928–931, <https://doi.org/10.1090/S0002-9904-1967-11853-6>, 1967.
- Do, S.-H., So, B.-D., Kim, Y.-G., and Kim, G.-B.: Lithospheric strength inferred from modeling of buckling structure: Implications for stress state of the East Sea (Japan Sea), *Tectonophysics*, 858, 229859, <https://doi.org/10.1016/j.tecto.2023.229859>, 2023.
- Duret, T., May, D. A., Gerya, T., and Tackley, P.: Discretization errors and free surface stabilization in the finite difference and marker-in-cell method for applied geodynamics: A numerical study, *Geochem. Geophys. Geosy.*, 12, <https://doi.org/10.1029/2011GC003567>, 2011.
- Elman, H. C. and Golub, G. H.: Inexact and preconditioned Uzawa algorithms for saddle point problems, *SIAM J. Numer. Anal.*, 31, 1645–1661, <https://doi.org/10.1137/0731085>, 1994.
- Gerya, T.: Introduction to numerical geodynamic modelling, Cambridge University Press, <https://doi.org/10.1017/9781316534243>, 2019.
- Gerya, T.: Numerical modeling of subduction: State of the art and future directions, *Geosphere*, 18, 503–561, <https://doi.org/10.1130/GES02416.1>, 2022.
- Gmeiner, B., Huber, M., John, L., Rüde, U., and Wohlmuth, B.: A quantitative performance study for Stokes solvers at the extreme scale, *J. Comput. Sci.*, 17, 509–521, <https://doi.org/10.1016/j.jocs.2016.06.006>, 2016.
- Guermond, J.-L. and Quartapelle, L.: On the approximation of the unsteady Navier-Stokes equations by finite element projection methods, *Numer. Math.*, 80, 207–238, <https://doi.org/10.1007/s002110050366>, 1998.
- Hestenes, M. R. and Stiefel, E.: Methods of conjugate gradients for solving linear systems, *J. Res. Nat. Bur. Stand.*, 49, 409–436, <https://doi.org/10.6028/jres.049.044>, 1952.
- Huismans, R. S., Podladchikov, Y. Y., and Cloetingh, S.: Dynamic modeling of the transition from passive to active rifting, application to the Pannonian basin, *Tectonics*, 20, 1021–1039, <https://doi.org/10.1029/2001TC900010>, 2001.
- Ismail-Zadeh, A., Schubert, G., Tsepelev, I., and Korotkii, A.: Three-dimensional forward and backward numerical modeling of mantle plume evolution: Effects of thermal diffusion, *J. Geophys. Res.-Sol. Ea.*, 111, <https://doi.org/10.1029/2005JB003782>, 2006.
- Jang, D.-K., Lee, K.-M., Thieulot, C., Choi, W.-H., and So, B.-D.: Efficient Uzawa Algorithms with Projection Strategies for Geodynamic Stokes Flow, Zenodo [code], <https://doi.org/10.5281/zenodo.22135967>, 2026.
- Kim, D.-H. and So, B.-D.: Effects of rheology and mantle temperature structure on edge-driven convection: Implications for partial melting and dynamic topography, *Phys. Earth Planet. In.*, 303, 106487, <https://doi.org/10.1016/j.pepi.2020.106487>, 2020.
- King, S., Lee, C., Leng, W., Zhong, S., van Keken, P., Tan, E., Gurnis, M., and Tosi, N.: A Community Benchmark for 2D Cartesian Compressible Convection, AGU Fall Meeting Abstracts, 2008, DI31A-1772, 2008.
- Kronbichler, M., Heister, T., and Bangerth, W.: High accuracy mantle convection simulation through modern numerical methods, *Geophys. J. Int.*, 191, 12–29, <https://doi.org/10.1111/j.1365-246X.2012.05609.x>, 2012.
- Lee, K.-M., Kim, H.-I., Jang, D.-K., and So, B.-D.: A comparative study of 2D numerical simulations using vector and potential methods for extending 3D modeling of planetary evolution, *Geosci. J.*, 28, 465–481, <https://doi.org/10.1007/s12303-024-0013-5>, 2024.
- Li, Y. and Gurnis, M.: A simple force balance model of subduction initiation, *Geophys. J. Int.*, 232, 128–146, <https://doi.org/10.1093/gji/ggac332>, 2023.
- Logg, A., Mardal, K.-A., and Wells, G. (Eds.): Automated solution of differential equations by the finite element method: The FEniCS book, vol. 84, Springer Science & Business Media, <https://doi.org/10.1007/978-3-642-23099-8>, 2012.
- Mansour, J. A., Giordani, J., Moresi, L., Beucher, R., Kaluza, O., Velic, M., Farrington, R., Quenette, S., and Beall, A.: Underworld2: Python geodynamics modelling for desktop, HPC and cloud, *Journal of Open Source Software*, 5, 1797, <https://doi.org/10.21105/joss.01797>, 2020.

- May, D. A. and Moresi, L.: Preconditioned iterative methods for Stokes flow problems arising in computational geodynamics, *Phys. Earth Planet. In.*, 171, 33–47, <https://doi.org/10.1016/j.pepi.2008.07.036>, 2008.
- Moresi, L. and Gurnis, M.: The accuracy of finite element solutions of Stokes flow with strongly varying viscosity, *Phys. Earth Planet. In.*, 97, 83–94, [https://doi.org/10.1016/0031-9201\(96\)03163-9](https://doi.org/10.1016/0031-9201(96)03163-9), 1996.
- Moresi, L., Quenette, S., Lemiale, V., Meriaux, C., Appelbe, B., and Mühlhaus, H.-B.: Computational approaches to studying non-linear dynamics of the crust and mantle, *Phys. Earth Planet. In.*, 163, 69–82, <https://doi.org/10.1016/j.pepi.2007.06.009>, 2007.
- Negredo, A. M., van Hunen, J., Rodríguez-González, J., and Fullea, J.: On the origin of the Canary Islands: Insights from mantle convection modelling, *Earth Planet. Sc. Lett.*, 584, 117506, <https://doi.org/10.1016/j.epsl.2022.117506>, 2022.
- Pyo, J.-H. and Shen, J.: Gauge–Uzawa methods for incompressible flows with variable density, *J. Computat. Phys.*, 221, 181–197, <https://doi.org/10.1016/j.jcp.2006.06.013>, 2007.
- Ramage, A. and Wathen, A. J.: Iterative solution techniques for the Stokes and Navier-Stokes equations, *Int. J. Numer. Meth. Fl.*, 19, 67–83, <https://doi.org/10.1002/fld.1650190106>, 1994.
- Rudi, J., Stadler, G., and Ghattas, O.: Weighted BFBT preconditioner for Stokes flow problems with highly heterogeneous viscosity, *SIAM J. Sci. Comput.*, 39, S272–S297, <https://doi.org/10.1137/16M108450X>, 2017.
- Saad, Y.: Iterative methods for sparse linear systems, *SIAM*, <https://doi.org/10.1137/1.9780898718003>, 2003.
- Saad, Y. and Schultz, M. H.: GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Stat. Comp.*, 7, 856–869, <https://doi.org/10.1137/0907058>, 1986.
- Schubert, G., Turcotte, D. L., and Olson, P.: *Mantle Convection in the Earth and Planets*, Cambridge University Press, <https://doi.org/10.1017/CBO9780511612879>, 2001.
- Silvester, D. and Wathen, A.: Fast iterative solution of stabilised Stokes systems Part II: Using general block preconditioners, *SIAM J. Numer. Anal.*, 31, 1352–1367, <https://doi.org/10.1137/0731070>, 1994.
- Sime, N., Wilson, C. R., and van Keken, P. E.: Thermal modeling of subduction zones with prescribed and evolving 2D and 3D slab geometries, *Progress in Earth and Planetary Science*, 11, 14, <https://doi.org/10.1186/s40645-024-00611-4>, 2024.
- So, B.-D. and Yuen, D. A.: Generation of tectonic over-pressure inside subducting oceanic lithosphere involving phase-loop of olivine–wadsleyite transition, *Earth Planet. Sc. Lett.*, 413, 59–69, <https://doi.org/10.1016/j.epsl.2014.12.048>, 2015.
- Tan, E. and Gurnis, M.: Metastable superplumes and mantle compressibility, *Geophys. Res. Lett.*, 32, <https://doi.org/10.1029/2005GL024190>, 2005.
- Thieulot, C. and Bangerth, W.: On the choice of finite element for applications in geodynamics – Part 2: A comparison of simplex and hypercube elements, *Solid Earth*, 16, 457–476, <https://doi.org/10.5194/se-16-457-2025>, 2025.
- van Keken, P.: Evolution of starting mantle plumes: a comparison between numerical and laboratory models, *Earth Planet. Sc. Lett.*, 148, 1–11, [https://doi.org/10.1016/S0012-821X\(97\)00042-3](https://doi.org/10.1016/S0012-821X(97)00042-3), 1997.
- Vynnytska, L., Rognes, M. E., and Clark, S.: Benchmarking FEniCS for mantle convection simulations, *Comput. Geosci.*, 50, 95–105, <https://doi.org/10.1016/j.cageo.2012.05.012>, 2013.
- Wang, H., Agrusta, R., and van Hunen, J.: Advantages of a conservative velocity interpolation (CVI) scheme for particle-in-cell methods with application in geodynamic modeling, *Geochem. Geophys. Geosy.*, 16, 4148–4166, <https://doi.org/10.1002/2015GC005824>, 2015.
- Wilson, C. R., Spiegelman, M., and van Keken, P. E.: Terra FERMA: The Transparent Finite Element Rapid Model Assembler for multiphysics problems in Earth sciences, *Geochem. Geophys. Geosy.*, 18, 769–810, <https://doi.org/10.1002/2016GC006702>, 2017.
- Xie, R., Chen, L., Morgan, J. P., and Chen, Y. J.: Various lithospheric deformation patterns derived from rheological contrasts between continental terranes: insights from 2-D numerical simulations, *Solid Earth*, 15, 789–806, <https://doi.org/10.5194/se-15-789-2024>, 2024.
- Zhamaletdinov, A., Shevtsov, A., Korotkova, T., Kopytenko, Y. A., Ismagilov, V. S., Petrishev, M. S., Efimov, B. V., Barannik, M. B., Kolobov, V. V., Prokopchuk, P. I., Smirnov, M. Yu., Vagin, S. A., Pertel, M. I., Tereshchenko, E. D., Vasil'ev, A. N., Grigoryev, V. F., Gokhberg, M. B., Trofimchik, V. I., Yampolsky, Yu. M., Koloskov, A. V., Fedorov, A. V., and Korja, T.: Deep electromagnetic sounding of the lithosphere in the Eastern Baltic (Fennoscandian) shield with high-power controlled sources and industrial power transmission lines (FENICS experiment), *Izvestiya, Physics of the Solid Earth*, 47, 22, <https://doi.org/10.1134/S1069351311010149>, 2011.
- Zhao, X.-H., Kwek, K.-H., Li, J.-B., and Huang, K.-L.: Chaotic and resonant streamlines in the ABC flow, *SIAM J. Appl. Math.*, 53, 71–77, <https://doi.org/10.1137/0153005>, 1993.
- Zhong, S.: Analytic solutions for Stokes' flow with lateral variations in viscosity, *Geophys. J. Int.*, 124, 18–28, <https://doi.org/10.1111/j.1365-246X.1996.tb06349.x>, 1996.
- Zhong, S., Zuber, M. T., Moresi, L., and Gurnis, M.: Role of temperature-dependent viscosity and surface plates in spherical shell models of mantle convection, *J. Geophys. Res.-Sol. Ea.*, 105, 11063–11082, <https://doi.org/10.1029/2000JB900003>, 2000.