# The impact of velocity update frequency on time accuracy for mantle convection particle methods

**S. J. Trim[1], S. L. Butler[2], and R. J. Spiteri[1]**

[1]Department of Computer Science, University of Saskatchewan, Canada
[2]Department of Geological Sciences, University of Saskatchewan, Canada

**Key Points:**

- Computation of the velocity field is expensive for mantle convection codes
- Computing the velocity for a subset of Runge–Kutta stages can give acceptable results for modest error tolerances at reduced cost
- The reduced cost from not updating the velocity at each time integration stage is not worth the accuracy lost for tight error tolerances

Corresponding author: Sean Trim, `sean.trim@usask.ca`

## Abstract

Computing the velocity field is an expensive process for mantle convection codes. This has implications for particle methods used to model the advection of quantities such as temperature or composition. A common choice for the numerical treatment of particle trajectories is classical fourth-order Runge–Kutta (ERK4) integration, which involves a velocity computation at each of its four stages. To reduce the cost per time step, it is possible to evaluate the velocity for a subset of the four time integration stages. We explore two such alternative schemes, in which velocities are only computed for: a) stage 1 on odd-numbered time steps and stages 2–4 for even-numbered time steps, and b) stage 1 for all time steps. A theoretical analysis of stability and accuracy is presented for all schemes. It was found that the alternative schemes are first-order accurate with stability regions different from that of ERK4. The efficiency and accuracy of the alternate schemes were compared against ERK4 in four test problems covering isothermal, thermal, and thermochemical flows. Exact solutions were used as reference solutions when available. In agreement with theory, the alternate schemes were observed to be first-order accurate for all test problems. Accordingly, they may be used to efficiently compute solutions to within modest error tolerances. For small error tolerances, however, ERK4 was the most efficient.



## Plain Language Summary

Computation of the flow velocity is an expensive process for mantle convection codes. For high-velocity flows, tracer particle methods can be used to model the transport of mantle material while minimizing model errors such as artificial diffusion. A major element of tracer particle methods is the computation of particle trajectories, which is commonly done using Runge–Kutta (RK) methods. A popular choice is the classical fourth-order accurate RK method, which is comprised of four stages per model time step. Strictly speaking, flow velocities are to be computed for each RK stage. However, flow velocities can be computed less often in order to reduce computation time. In this study, we examine the impact of how often the flow velocities are computed during time integration on accuracy and efficiency. We find that velocities can be computed less frequently to efficiently compute solutions to modest error tolerances. However, computing the velocities for every time integration stage is the most efficient way to find solutions with tight error tolerances.


## 1 Introduction

### 1.1 Motivation

Numerical modeling of mantle convection can be a computationally intensive venture. Modeling large domains over geologic time may require weeks of computation time. Accordingly, methods that are sufficiently accurate while being computationally efficient are of great value.

Mantle convection is an advection-dominated process in two respects. First, for models featuring distinct compositional components, compositional diffusion is extremely small and is often assumed to be zero (van Keken et al., 1997). Second, the effect of thermal diffusion is small relative to that of advection (except at thermal boundary layers), particularly for high Rayleigh number convection (Schubert et al., 2001). In these situations, particles are often used to model the process of advection in order to minimize numerical diffusion and spurious oscillations in the numerical solution. However, to limit statistical noise, many particles must be used, leading to a substantial computational cost.

In previous mantle convection studies, the use of compositionally distinct materials has been motivated by Large Low Shear wave Velocity Provinces (LLSVPs) (S. Trim et al., 2014; S. J. Trim & Lowman, 2016; McNamara & Zhong, 2005), Ultra Low Velocity Zones (ULVZs) (Li et al., 2017), oceanic crust (Brandenburg & Van Keken, 2007; Li & McNamara, 2013), and continents (Lenardic et al., 2003; Rolf & Tackley, 2011). Distinct compositional components have also been used to approximate a free surface boundary condition (Crameri et al., 2012). Thermal buoyancy in the mantle is driven by radiogenic heating and the temperature contrast between the surface and the outer core that is approximately 4000 K (Schubert et al., 2001).

As part of model time integration, particles must be numerically advected, with explicit Runge–Kutta (ERK) integrators being commonly used for this purpose. For instance, ERK integrators are available for particle advection in ASPECT (Gassmöller et al., 2018; Puckett et al., 2018; Gassmöller et al., 2019), CitcomS (Zhong & Hager, 2003; McNamara & Zhong, 2004), ConMan (Tackley & King, 2003), I2VIS (Gerya & Yuen, 2003), MC3D (Gable et al., 1991), ProjecTracer (S. Trim et al., 2020), StagYY (Tackley & King, 2003; Tackley, 2008), and Underworld2 (Moresi et al., 2002, 2003). ERK methods are self-starting and can offer high-order accuracy. The formal definition of ERK methods requires the particle velocity be evaluated for each stage of the integration. This requirement can be problematic due to the substantial expense of evaluating the velocity field (by solving the momentum conservation and continuity equations) for the ERK stages.

To reduce computation time, it is possible to contemplate the use of ERK methods without computing the velocity via the governing equations for each stage, an approach that has been used in previous studies (Puckett et al., 2018; Gerya & Yuen, 2003; S. Trim et al., 2014; S. J. Trim & Lowman, 2016). Although it is clear that order of accuracy is reduced by such an approach, the practical impact on simulation accuracy and efficiency is less clear and has not been deeply explored. In this paper, we examine the effect of different velocity update schemes (see section 2.3) on the classical fourth-order ERK time integrator, which has been widely used in prior work. In particular, we examine the trade-offs between accuracy and efficiency of the ensuing time integrators for four distinct test problems, which we now describe.

### 1.2 Test Problems

We test the efficacy of different velocity update schemes under three convective regimes: compositional convection where composition drives buoyancy; thermal convection where temperature drives buoyancy; and thermochemical where both temperature and composition drive buoyancy. Each convective regime is explored using a suite of four test problems that we now outline (details appear in section 2.5).

1. A compositional Rayleigh–Taylor (RT) problem from van Keken et al. (1997).
2. A thermal RT problem, similar to the RT problem from van Keken et al. (1997) but with a thermal field in place of composition.
3. A thermal convection problem with a variable internal heating rate and known exact solution.
4. A thermochemical convection problem with a variable internal heating rate and known exact solution.

## 2 Methods

### 2.1 Governing Equations

Although solid over short periods, the Earth's mantle acts as a highly viscous fluid over geologic time (Schubert et al., 2001). Mathematically, the mantle is modeled using a set of conservation equations obtained from fluid dynamics and thermodynamics. Specifically, the continuity equation specifies the conservation of mass, the Navier–Stokes equation models the conservation of momentum, and an advection-diffusion equation governs the conservation of energy. In addition, an advection equation is used to model compositional transport of distinct mantle components. In this study, we employ the Boussinesq approximation to simplify the effect of compressibility. The infinite Prandtl number approximation is also used, for which the inertial terms in the Navier–Stokes equation are considered negligible (resulting in Stokes flow). These commonly used approximations result in time derivatives appearing in a subset of the conservation equations, leading to a set of partial differential-algebraic equations. The nondimensional conservation equations are

$$\nabla \cdot \mathbf{v} = 0, \tag{1}$$

$$\nabla \cdot \left(2\eta \underline{\underline{\dot{\epsilon}}}\right) - \nabla P = \left(Ra_T T - Ra_C C\right) \hat{\mathbf{g}}, \tag{2}$$

$$\frac{\partial T}{\partial t} + \mathbf{v} \cdot \nabla T = \nabla^2 T + H(\mathbf{r}, t), \tag{3}$$

and

$$\frac{\partial C}{\partial t} + \mathbf{v} \cdot \nabla C = 0, \tag{4}$$

where $\mathbf{r}$ is the position; $t$ is time; $\mathbf{v}$ is the velocity; $\eta$ is the viscosity (set to unity in this study); $\underline{\underline{\dot{\epsilon}}}$ is the strain rate tensor; $P$ is pressure; $T$ is temperature; $C$ is composition; $H(\mathbf{r}, t)$ is the internal heating rate; and $\hat{\mathbf{g}}$ is a unit vector opposite the direction of gravity. In two dimensions (2D), the strain rate tensor is given by

$$\underline{\underline{\dot{\epsilon}}} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{1}{2}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \\ \frac{1}{2}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) & \frac{\partial w}{\partial z} \end{bmatrix}, \tag{5}$$

where $x$ and $z$ are the horizontal and vertical Cartesian coordinates, respectively, and $u$ and $w$ are the corresponding components of the velocity such that $\mathbf{r} = [x, z]$ and $\mathbf{v} = [u, w]$.

The thermal Rayleigh number is

$$Ra_T = \frac{\alpha \rho_0 g \Delta T L^3}{\kappa \eta}, \tag{6}$$

where $\alpha$ is the thermal expansivity; $\rho_0$ is surface density with $C = T = 0$; $g$ is gravitational acceleration; $\Delta T$ is the temperature difference across the mantle; $L$ is the mantle thickness; and $\kappa$ is thermal diffusivity. The compositional Rayleigh number is

$$Ra_C = \frac{\Delta \rho_C g L^3}{\kappa \eta}, \tag{7}$$

where $\Delta \rho_C$ is the compositional density contrast between enriched and ambient mantle materials.

### 2.2 Numerical Solutions

We use the code ProjecTracer (S. Trim et al., 2020) for our numerical solutions (see Open Research section). The numerical methods of the code relevant to this study are described in sections 2.2.1–2.2.3.

#### *2.2.1 Alternate Forms of the Governing Equations*

Instead of directly solving equations 1–4, we numerically solve alternate forms of these governing equations that offer advantages for 2D incompressible advection-dominated flows.

Using a stream function formulation, equations 1–2 (with $\eta \equiv 1$) can be recast as the biharmonic equation

$$\frac{\partial^4 \psi}{\partial x^4} + \frac{\partial^4 \psi}{\partial x^2 \partial z^2} + \frac{\partial^4 \psi}{\partial z^4} = Ra_T \frac{\partial T}{\partial x} - Ra_C \frac{\partial C}{\partial x}, \tag{8}$$

where $\psi$ is the stream function (Batchelor, 1967; van Keken et al., 1997). After solving equation 8 for the stream function, the velocity is computed using

$$\mathbf{v} = \left( \frac{\partial \psi}{\partial z}, -\frac{\partial \psi}{\partial x} \right). \tag{9}$$

The stream function formulation reduces the number of scalar equations to be solved and removes the need to solve for the pressure albeit at the expense of solving a higher-order equation.

Equations 3 and 4 are written in the Eulerian reference frame. In the Lagrangian reference frame, the equations can be rewritten as

$$\frac{DT}{Dt} = \nabla^2 T + H \tag{10}$$

and

$$\frac{DC}{Dt} = 0, \tag{11}$$

where $\frac{D}{Dt} = \frac{d}{dt} + \mathbf{v} \cdot \nabla$ is the substantive derivative. Calculations in the Lagrangian frame are performed from the viewpoint of moving fluid parcels. Consequently, the trajectories of fluid parcels must be tracked. For a fluid parcel at position $\mathbf{r}$, the trajectory equation is

$$\frac{d\mathbf{r}}{dt} = \mathbf{v}. \tag{12}$$

#### *2.2.2 Spatial Discretization*

Calculations are performed using a combination of a fixed Eulerian grid and a moving Lagrangian grid. Specifically, equation 8 is solved using the Eulerian grid, whereas equations 10–12 are solved using the Lagrangian grid. We employ an Eulerian grid that has constant horizontal and vertical spacings, denoted by $\Delta x$ and $\Delta z$, respectively. Further, we use $\Delta x = \alpha \Delta z$, where $\alpha$ is the aspect ratio of the domain and chosen to be at or near unity for all problems (see section 2.5). The Lagrangian grid positions, known as tracer particles, correspond to moving fluid parcels. Spatial derivatives are approximated using centered finite differences with coefficients generated automatically from an algorithm developed by Fornberg (1988) (see also Larsen et al. (1997)) that permits the order of accuracy to be specified. Our results are based upon the use of fourth-order finite differences because experimentation revealed that fourth-order accuracy was the most effective choice for the problems considered. The system of equations obtained after discretizing equation 8 is solved using banded LU factorization.

In equation 10, the right-hand side is first calculated using centered finite differences for the diffusion term on the Eulerian mesh. The value of the right-hand side for tracer $q$, denoted by $\left\langle \nabla^2 T + H \right\rangle_q$, is calculated via interpolation using proximal values on the Eulerian mesh. The interpolation is carried out using Taylor polynomials with a sufficient number of terms to match the order of accuracy of the finite differences (see Appendix A of S. Trim et al. (2020) for an example).

In equation 12, the fluid parcel velocity is approximated by the velocity of tracer $q$, denoted by $\langle \mathbf{v} \rangle_q$, which is computed using three steps. First, equation 8 is solved based on values of $T$ and $C$ on the Eulerian grid. Second, equation 9 is used to calculated the velocity at the Eulerian grid points. Third, Taylor polynomials (similar to the above) are used to interpolate nearby Eulerian velocity values to compute $\langle \mathbf{v} \rangle_q$.

In order to calculate Eulerian values of $T$ and $C$ used in equation 8, the corresponding Lagrangian values must be interpolated to the Eulerian grid. This interpolation is done using bilinear shape functions as described in Tackley and King (2003).

We define particle density using the tracers per control volume (TPCV) at $t = 0$, where a control volume corresponds to a cell of the dual Eulerian mesh. Because the positions of the tracer particles fluctuate with time, the particle density for $t > 0$ varies. However, the total number of particles is fixed for each calculation. Uneven particle coverage may benefit from the dynamic redistribution of tracers (S. Trim et al., 2020). However, it was found that tracer repositioning did not significantly impact numerical results over the integration times used in this study. Accordingly, we do not employ tracer repositioning for simplicity.

#### *2.2.3 Temporal Discretization*

Spatial discretization of the equations 10-12 according to section 2.2.2 results in the system of ordinary differential equations (ODEs) given by

$$d\mathbf{y}_q/dt = \mathbf{F}(t, \mathbf{y}_q) \equiv \mathbf{D}(t, \mathbf{y}_q) + \mathbf{V}(\mathbf{y}_q), \tag{13}$$

with

$$\mathbf{y}_q = [T_q, C_q, \mathbf{r}_q]^\intercal, \tag{14}$$

$$\mathbf{D}(t, \mathbf{y}_q) = [\left\langle \nabla^2 T + H \right\rangle_q, 0, 0]^\intercal, \tag{15}$$

and

$$\mathbf{V}(\mathbf{y}_q) = [0, 0, \langle \mathbf{v} \rangle_q]^\intercal, \tag{16}$$

where $q$ is the tracer number (ranging between 1 and the total number of tracer particles, i.e., the product of the TPCV value and the number of control volumes). We write the right side of equation 13 in split form to better describe the time integration schemes that follow in section 2.3. We note that $H$ can have explicit time dependence in general; see problems 3-4. In contrast, the velocity field does not explicitly depend on $t$.

Time steps are performed using the classical four-stage, fourth-order explicit Runge–Kutta method (ERK4) and variations (outlined in section 2.3). To advance the approximate solution of equation 13 from time $t^n$ to time $t^{n+1} = t^n + \Delta t$, the ERK4 method is

$$\widehat{\mathbf{y}}_q^{n+\frac{1}{2}} = \mathbf{y}_q^n + \frac{\Delta t}{2} \mathbf{F}(t^n, \mathbf{y}_q^n), \tag{17}$$

$$\widetilde{\mathbf{y}}_q^{n+\frac{1}{2}} = \mathbf{y}_q^n + \frac{\Delta t}{2} \mathbf{F}(t^{n+\frac{1}{2}}, \widehat{\mathbf{y}}_q^{n+\frac{1}{2}}), \tag{18}$$

$$\overline{\mathbf{y}}_q^{n+1} = \mathbf{y}_q^n + \Delta t \, \mathbf{F}(t^{n+\frac{1}{2}}, \widetilde{\mathbf{y}}_q^{n+\frac{1}{2}}), \tag{19}$$

$$\begin{aligned} \mathbf{y}_q^{n+1} = \mathbf{y}_q^n + \frac{\Delta t}{6} \Big\{ & \mathbf{F}(t^n, \mathbf{y}_q^n) + 2\mathbf{F}(t^{n+\frac{1}{2}}, \widehat{\mathbf{y}}_q^{n+\frac{1}{2}}) \\ & + 2\mathbf{F}(t^{n+\frac{1}{2}}, \widetilde{\mathbf{y}}_q^{n+\frac{1}{2}}) + \mathbf{F}(t^{n+1}, \overline{\mathbf{y}}_q^{n+1}) \Big\}, \end{aligned} \tag{20}$$

where the superscript $n$ denotes the time step number and $\Delta t$ is the time step size. In this method, stages 1–4 correspond to equations 17–20, respectively.

### 2.3 Velocity Update Schemes

During each ERK4 time step, the velocity field is updated four times, corresponding to evaluations of $\mathbf{V}$ for each stage. In addition to ERK4, however, we also investigate two other schemes that do not update the velocity field (i.e., do not require distinct evaluations of $\mathbf{V}$) at each stage.

For odd-numbered $n$, the first alternate scheme, which we denote by ODD, is given by

$$\widehat{\mathbf{y}}_q^{n+\frac{1}{2}} = \mathbf{y}_q^n + \frac{\Delta t}{2}\left[\mathbf{D}(t^n, \mathbf{y}_q^n) + \mathbf{V}(\mathbf{y}_q^n)\right], \tag{21}$$

$$\widetilde{\mathbf{y}}_q^{n+\frac{1}{2}} = \mathbf{y}_q^n + \frac{\Delta t}{2}\left[\mathbf{D}(t^{n+\frac{1}{2}}, \widehat{\mathbf{y}}_q^{n+\frac{1}{2}}) + \mathbf{V}(\mathbf{y}_q^n))\right], \tag{22}$$

$$\overline{\mathbf{y}}_q^{n+1} = \mathbf{y}_q^n + \Delta t\left[\mathbf{D}(t^{n+\frac{1}{2}}, \widetilde{\mathbf{y}}_q^{n+\frac{1}{2}}) + \mathbf{V}(\mathbf{y}_q^n)\right], \tag{23}$$

$$\begin{aligned}\mathbf{y}_q^{n+1} = \mathbf{y}_q^n + \frac{\Delta t}{6}\Big\{&\left[\mathbf{D}(t^n, \mathbf{y}_q^n) + \mathbf{V}(\mathbf{y}_q^n)\right] + 2\left[\mathbf{D}(t^{n+\frac{1}{2}}, \widehat{\mathbf{y}}_q^{n+\frac{1}{2}}) + \mathbf{V}(\mathbf{y}_q^n)\right] \\ &+2\left[\mathbf{D}(t^{n+\frac{1}{2}}, \widetilde{\mathbf{y}}_q^{n+\frac{1}{2}}) + \mathbf{V}(\mathbf{y}_q^n)\right] + \left[\mathbf{D}(t^{n+1}, \overline{\mathbf{y}}_q^{n+1}) + \mathbf{V}(\mathbf{y}_q^n)\right]\Big\},\end{aligned} \tag{24}$$

$$\widehat{\mathbf{y}}_q^{n+\frac{3}{2}} = \mathbf{y}_q^{n+1} + \frac{\Delta t}{2}\left[\mathbf{D}(t^{n+1}, \mathbf{y}_q^{n+1}) + \mathbf{V}(\mathbf{y}_q^n)\right], \tag{25}$$

$$\widetilde{\mathbf{y}}_q^{n+\frac{3}{2}} = \mathbf{y}_q^{n+1} + \frac{\Delta t}{2}\left[\mathbf{D}(t^{n+\frac{3}{2}}, \widehat{\mathbf{y}}_q^{n+\frac{3}{2}}) + \mathbf{V}(\widehat{\mathbf{y}}_q^{n+\frac{3}{2}})\right], \tag{26}$$

$$\overline{\mathbf{y}}_q^{n+2} = \mathbf{y}_q^{n+1} + \Delta t\left[\mathbf{D}(t^{n+\frac{3}{2}}, \widetilde{\mathbf{y}}_q^{n+\frac{3}{2}}) + \mathbf{V}(\widetilde{\mathbf{y}}_q^{n+\frac{3}{2}})\right], \tag{27}$$

$$\begin{aligned}\mathbf{y}_q^{n+2} = \mathbf{y}_q^{n+1} + \frac{\Delta t}{6}\Big\{&\left[\mathbf{D}(t^{n+1}, \mathbf{y}_q^{n+1}) + \mathbf{V}(\mathbf{y}_q^n)\right] + 2\left[\mathbf{D}(t^{n+\frac{3}{2}}, \widehat{\mathbf{y}}_q^{n+\frac{3}{2}}) + \mathbf{V}(\widehat{\mathbf{y}}_q^{n+\frac{3}{2}})\right] \\ &+2\left[\mathbf{D}(t^{n+\frac{3}{2}}, \widetilde{\mathbf{y}}_q^{n+\frac{3}{2}}) + \mathbf{V}(\widetilde{\mathbf{y}}_q^{n+\frac{3}{2}})\right] + \left[\mathbf{D}(t^{n+2}, \overline{\mathbf{y}}_q^{n+2}) + \mathbf{V}(\overline{\mathbf{y}}_q^{n+2})\right]\Big\},\end{aligned} \tag{28}$$

where $\mathbf{y}_q^{n+1}$ and $\mathbf{y}_q^{n+2}$ are values of $\mathbf{y}_q$ at subsequent even- and odd-numbered time steps, respectively. We note that equations 21–28, corresponding to stages 1–8, apply to a time interval of $2\Delta t$. We denote this scheme by ODD because stage-1 velocity updates only occur on odd-numbered time steps. Specifically, velocity updates occur for stages 1, 6, 7, and 8. A similar scheme is available as an option in the mantle convection code MC3D (Gable et al., 1991), which utilizes an Eulerian solver for $T$ and a particle method for $C$ (see S. Trim (2017) for details on the numerical methods). We are unaware of any published studies that use this option. However, this velocity update scheme remains of interest due to the existence of an implementation in a published code.

The second alternate scheme considered only updates the velocity field at the first stage of the method. We denote this scheme by SG1 (for "stage 1"). The SG1 scheme has been used in the mantle convection codes MC3D (S. Trim et al., 2014; S. J. Trim & Lowman, 2016) and I2VIS (Gerya & Yuen, 2003). This scheme may be easily adapted to other Runge–Kutta (RK) methods. The SG1 scheme is given by equations 21–24, corresponding to stages 1–4 over a time interval of $\Delta t$, for all $n$.

The details for all schemes tested are summarized in table 1.

**Table 1.** Velocity update schemes[a]

| Scheme | Number of Stages | Stages With Velocity Updates | Time Interval |
|---|---|---|---|
| ERK4 | 4 | 1,2,3,4 | $\Delta t$ |
| ODD | 8 | 1,6,7,8 | $2\Delta t$ |
| SG1 | 4 | 1 | $\Delta t$ |

[a] For the schemes named in the first column, the second column indicates the total number of stages, the third column indicates which stages apply velocity updates, and the fourth column indicates the time interval covered.

### 2.4 Numerical analysis of the velocity update schemes

#### *2.4.1 General analysis*

To determine the accuracy and stability properties of the schemes presented in sections 2.2.3 and 2.3, we apply them to a test ODE in eigenspace, analogous to the ODE governing the evolution of an arbitrarily selected tracer particle (see equation 13). For our schemes, the test ODE is given by

$$d\mathbf{y}/dt = \mathbf{F}(t,\mathbf{y}) = \mathbf{D}(t,\mathbf{y}) + \mathbf{V}(\mathbf{y}) = \lambda\mathbf{y} + \mathbf{a}e^{\mu t} \tag{29}$$

where the eigenvalue $\lambda$, vector $\mathbf{a}$, and $\mu$ are complex constants. To accommodate the splitting of derivative evaluations in the ODD and SG1 schemes, we apply the definitions

$$\mathbf{D}(t,\mathbf{y}) = \lambda_D\mathbf{y} + \mathbf{a}e^{\mu t} \tag{30}$$

and

$$\mathbf{V}(\mathbf{y}) = \lambda_V\mathbf{y}, \tag{31}$$

where $\lambda = \lambda_D + \lambda_V$. To quantify the proportions of $\lambda_D$ and $\lambda_V$, we define $\lambda_V = \gamma\lambda$ and $\lambda_D = (1-\gamma)\lambda$. With this definition, $\gamma$ values of zero or unity indicate that $\lambda$ is independent of $\lambda_V$ or $\lambda_D$, respectively. Application of a time integration scheme to equation 29 results in an ordinary difference equation (O$\Delta$E) that may be written in the form

$$P(E)\mathbf{y}^n = Q(E)\,\mathbf{a}e^{\mu\Delta tn}, \tag{32}$$

where $E$ is the shift operator with the property $\mathbf{y}^{n+k} = E^k\mathbf{y}^n$, and $P(E)$ and $Q(E)$ are the characteristic and particular polynomials, respectively (Lomax et al., 2001). The roots of the characteristic polynomial, denoted by $\sigma_j$ and satisfying $P(\sigma_j) = 0$, determine numerical stability. Specifically, $|\sigma_j| \leq 1$ for $j = 1 \ldots N_\sigma$, where $N_\sigma$ is the number of roots, is required for numerical stability.

The exact and approximate solutions of equation 29 are given by

$$\mathbf{y}(t) = \mathbf{d}e^{\lambda\Delta t} + \frac{\mathbf{a}e^{\mu t}}{\mu-\lambda} \tag{33}$$

and

$$\mathbf{y}^n = \sum_{j=1}^{N_\sigma}\mathbf{d}_j(\sigma_j)^n + \mathbf{a}e^{\mu\Delta tn}\frac{Q(e^{\mu\Delta t})}{P(e^{\mu\Delta t})}, \tag{34}$$

respectively, where $\mathbf{d}$ and $\mathbf{d}_j$ are constants. The first and second terms on the right sides of equations 33–34 correspond to the transient and particular solutions of the representative equation. A comparison of equations 33–34 allows numerical accuracy to be quantified.

For schemes with multiple $\sigma$-roots, $\sigma_1$ is defined as the principal root and is the root that has the smallest difference with the Taylor series of $e^{\lambda\Delta t}$ about $\Delta t = 0$. The corresponding local error in the transient solution is quantified by

$$err_\lambda = e^{\lambda\Delta t} - \sigma_1. \tag{35}$$

We note that only the principal $\sigma$-root contributes to equation 35 because it is presumed that the value of $\mathbf{d}_j$ is near zero for $j > 1$. Similarly, the local error in the particular solution is quantified using

$$err_\mu = \Delta t\left[\frac{Q(e^{\mu\Delta t})}{P(e^{\mu\Delta t})}(\mu-\lambda) - 1\right]. \tag{36}$$

The lowest-order term in $\Delta t$ from equations 35–36 determines the theoretical order of accuracy of the time integration scheme. Specifically, if $err_\lambda = \mathcal{O}(\Delta t^{p_\lambda})$ and $err_\mu = \mathcal{O}(\Delta t^{p_\mu})$, then the theoretical global order of accuracy is $\min(p_\lambda, p_\mu) - 1$.

The characteristic polynomial for the ERK4 scheme is

$$P_{\mathrm{ERK4}}(E) = E - 1 - \Delta t\lambda - \frac{1}{2}\,\Delta t^2\lambda^2 - \frac{1}{6}\,\Delta t^3\lambda^3 - \frac{1}{24}\,\Delta t^4\lambda^4, \tag{37}$$

and the corresponding particular polynomial is

$$\begin{aligned} Q_{\mathrm{ERK4}}(E) = {} & \frac{1}{24}\,\Delta t^4\lambda^3 + \frac{1}{12}\,\Delta t^3\lambda^2 + \frac{1}{6}\,\Delta t^2\lambda + \frac{1}{6}\,(E+1)\Delta t \\ & + \frac{1}{12}\left(\Delta t^3\lambda^2 + 4\,\Delta t^2\lambda + 8\,\Delta t\right)\sqrt{E}. \end{aligned} \tag{38}$$

From equation 37, we obtain a single $\sigma$-root given by

$$\sigma_{1,\mathrm{ERK4}} = 1 + \Delta t\lambda + \frac{1}{2}\,\Delta t^2\lambda^2 + \frac{1}{6}\,\Delta t^3\lambda^3 + \frac{1}{24}\,\Delta t^4\lambda^4, \tag{39}$$

from which the stability properties of the ERK4 scheme may be obtained. The stability region of the ERK4 scheme is shown for reference in figure 1 parts a and b.

The local errors in the transient and particular solutions for the ERK4 scheme are

$$err_{\lambda,\mathrm{ERK4}} = \frac{1}{120}\,\Delta t^5\lambda^5 + \mathcal{O}(\Delta t^6) \tag{40}$$

and

$$err_{\mu,\mathrm{ERK4}} = \frac{\left(30\,\lambda^3 - 10\,\lambda^2\mu + 5\,\lambda\mu^2 - \mu^3\right)\mu^2}{2880\,(\lambda-\mu)}\Delta t^5 + \mathcal{O}(\Delta t^6), \tag{41}$$

respectively. Both errors have leading terms of magnitude $\mathcal{O}(\Delta t^5)$. This gives a theoretical global fourth-order accuracy for the ERK4 scheme. As expected, the stability and accuracy of the ERK4 scheme are unaffected by the value of $\gamma$.

The characteristic polynomial for the ODD scheme is

$$\begin{aligned} P_{\mathrm{ODD}}(E) = {} & E^2 - 1 - 2\,\Delta t\lambda + \frac{2}{3}\,(\gamma-3)\Delta t^2\lambda^2 - \frac{1}{12}\left(3\,\gamma^2 - 13\,\gamma + 16\right)\Delta t^3\lambda^3 \\ & + \frac{1}{72}\left(5\,\gamma^3 - 31\,\gamma^2 + 65\,\gamma - 48\right)\Delta t^4\lambda^4 - \frac{1}{144}\left(\gamma^4 - 13\,\gamma^3 + 47\,\gamma^2 - 71\,\gamma + 36\right)\Delta t^5\lambda^5 \\ & - \frac{1}{144}\left(\gamma^4 - 8\,\gamma^3 + 23\,\gamma^2 - 26\,\gamma + 10\right)\Delta t^6\lambda^6 - \frac{1}{288}\left(\gamma^4 - 7\,\gamma^3 + 15\,\gamma^2 - 13\,\gamma + 4\right)\Delta t^7\lambda^7 \\ & - \frac{1}{576}\left(\gamma^4 - 4\,\gamma^3 + 6\,\gamma^2 - 4\,\gamma + 1\right)\Delta t^8\lambda^8, \end{aligned} \tag{42}$$

and its particular polynomial is

$$\begin{aligned} Q_{\mathrm{ODD}}(E) = {} & \frac{1}{576}\left(\gamma^4 - 4\,\gamma^3 + 6\,\gamma^2 - 4\,\gamma + 1\right)\Delta t^8\lambda^7 + \frac{1}{288}\left(\gamma^4 - 6\,\gamma^3 + 12\,\gamma^2 - 10\,\gamma + 3\right)\Delta t^7\lambda^6 \\ & + \frac{1}{144}\left(\gamma^4 - 7\,\gamma^3 + 17\,\gamma^2 - 17\,\gamma + 6\right)\Delta t^6\lambda^5 \\ & + \frac{1}{144}\left(\gamma^4 - 11\,\gamma^3 - (E+40)\gamma + 33\,\gamma^2 + E + 17\right)\Delta t^5\lambda^4 \\ & - \frac{1}{72}\left(4\,\gamma^3 + (E+31)\gamma - 19\,\gamma^2 - 5\,E - 17\right)\Delta t^4\lambda^3 \\ & - \frac{1}{36}\left((E+14)\gamma - 4\,\gamma^2 - 6\,E - 12\right)\Delta t^3\lambda^2 - \frac{1}{36}\left((E+7)\gamma - 12\,E - 12\right)\Delta t^2\lambda \\ & + \frac{1}{6}\left(E^2 + 2\,E + 1\right)\Delta t - \frac{1}{288}\Big[\left(\gamma^3 - 3\,\gamma^2 + 3\,\gamma - 1\right)\Delta t^7\lambda^6 \\ & + 2\left(\gamma^3 - 6\,\gamma^2 + 9\,\gamma - 4\right)\Delta t^6\lambda^5 + 4\left(\gamma^3 - 7\,\gamma^2 + 15\,\gamma - 9\right)\Delta t^5\lambda^4 \\ & + 4\left(\gamma^3 - 12\,\gamma^2 + 33\,\gamma - 26\right)\Delta t^4\lambda^3 - 8\left(5\,\gamma^2 + 3\,E - 24\,\gamma + 27\right)\Delta t^3\lambda^2 \\ & - 32\,(3\,E - 4\,\gamma + 9)\Delta t^2\lambda - 192\,(E+1)\Delta t\Big]\sqrt{E}. \end{aligned} \tag{43}$$

The characteristic polynomial $P_{\mathrm{ODD}}$ gives two $\sigma$-roots. The principal root is

$$\begin{aligned}\sigma_{1,\mathrm{ODD}} = \frac{1}{24}\Big[&\left(\gamma^4 - 4\,\gamma^3 + 6\,\gamma^2 - 4\,\gamma + 1\right)\Delta t^8\lambda^8 + 2\left(\gamma^4 - 7\,\gamma^3 + 15\,\gamma^2 - 13\,\gamma + 4\right)\Delta t^7\lambda^7 \\ &+ 4\left(\gamma^4 - 8\,\gamma^3 + 23\,\gamma^2 - 26\,\gamma + 10\right)\Delta t^6\lambda^6 + 4\left(\gamma^4 - 13\,\gamma^3 + 47\,\gamma^2 - 71\,\gamma + 36\right)\Delta t^5\lambda^5 \\ &- 8\left(5\,\gamma^3 - 31\,\gamma^2 + 65\,\gamma - 48\right)\Delta t^4\lambda^4 + 48\left(3\,\gamma^2 - 13\,\gamma + 16\right)\Delta t^3\lambda^3 - 384\,(\gamma - 3)\Delta t^2\lambda^2 \\ &+ 1152\,\Delta t\lambda + 576\Big]^{1/2},\end{aligned} \tag{44}$$

and the second root is $\sigma_{2,\mathrm{ODD}} = -\sigma_{1,\mathrm{ODD}}$. We note that the moduli of both $\sigma$-roots are equal. Accordingly, we may define the stability region for the ODD scheme using equation 44. Additionally, $\sigma_{1,\mathrm{ODD}}$ is the amplification factor per time step. To obtain the amplification factor for one iteration of the ODD scheme (i.e., over two successive time steps) one must take the square of equation 44. Figure 1a shows plots of the stability regions for several values of $\gamma$ for the ODD scheme. We note that the stability region for the ODD scheme is equivalent to that of the ERK4 scheme when $\gamma = 0$ (it can be shown after some algebra that $\sigma_{1,\mathrm{ODD}} = \sigma_{1,\mathrm{ERK4}}$ if $\gamma = 0$).

The local error in the transient solution for the ODD scheme is

$$\begin{aligned}err_{\lambda,\mathrm{ODD}} = &-\frac{1}{1440}\left(5\,\gamma^4\lambda^5 + 45\,\gamma^3\lambda^5 - 5\,\gamma^2\lambda^5 - 15\,\gamma\lambda^5 - 12\,\lambda^5\right)\Delta t^5 \\ &+ \frac{1}{144}\left(5\,\gamma^3\lambda^4 - 5\,\gamma^2\lambda^4 + 11\,\gamma\lambda^4\right)\Delta t^4 - \frac{1}{24}\left(3\,\gamma^2\lambda^3 - 5\,\gamma\lambda^3\right)\Delta t^3 + \frac{1}{3}\,\gamma\Delta t^2\lambda^2 \\ &+ \mathcal{O}(\Delta t^6),\end{aligned} \tag{45}$$

and the local error in the particular solution is

$$\begin{aligned}err_{\mu,\mathrm{ODD}} = \frac{\Delta t^2\mu}{25920\,(\lambda-\mu)^4}\Big[&155\,\gamma^4\Delta t^3\lambda^7 - 660\,\gamma^4\Delta t^3\lambda^6\mu + 735\,\gamma^4\Delta t^3\lambda^5\mu^2 + 90\,\gamma^4\Delta t^3\lambda^4\mu^3 \\ &- 585\,\gamma^3\Delta t^3\lambda^7 + 2715\,\gamma^3\Delta t^3\lambda^6\mu - 3945\,\gamma^3\Delta t^3\lambda^5\mu^2 + 1005\,\gamma^3\Delta t^3\lambda^4\mu^3 \\ &+ 810\,\gamma^3\Delta t^3\lambda^3\mu^4 + 780\,\gamma^2\Delta t^3\lambda^7 - 3870\,\gamma^2\Delta t^3\lambda^6\mu + 7530\,\gamma^2\Delta t^3\lambda^5\mu^2 \\ &- 6165\,\gamma^2\Delta t^3\lambda^4\mu^3 + 1320\,\gamma^2\Delta t^3\lambda^3\mu^4 + 405\,\gamma^2\Delta t^3\lambda^2\mu^5 - 300\,\gamma^3\Delta t^2\lambda^6 \\ &- 180\,\gamma\Delta t^3\lambda^7 + 660\,\gamma^3\Delta t^2\lambda^5\mu + 1350\,\gamma\Delta t^3\lambda^6\mu + 540\,\gamma^3\Delta t^2\lambda^4\mu^2 \\ &- 4410\,\gamma\Delta t^3\lambda^5\mu^2 - 900\,\gamma^3\Delta t^2\lambda^3\mu^3 + 7290\,\gamma\Delta t^3\lambda^4\mu^3 - 6210\,\gamma\Delta t^3\lambda^3\mu^4 \\ &+ 2520\,\gamma\Delta t^3\lambda^2\mu^5 - 360\,\gamma\Delta t^3\lambda\mu^6 + 1080\,\gamma^2\Delta t^2\lambda^6 - 3060\,\gamma^2\Delta t^2\lambda^5\mu + 270\,\Delta t^3\lambda^6\mu \\ &+ 540\,\gamma^2\Delta t^2\lambda^4\mu^2 - 900\,\Delta t^3\lambda^5\mu^2 + 3780\,\gamma^2\Delta t^2\lambda^3\mu^3 + 1125\,\Delta t^3\lambda^4\mu^3 \\ &- 2340\,\gamma^2\Delta t^2\lambda^2\mu^4 - 684\,\Delta t^3\lambda^3\mu^4 + 252\,\Delta t^3\lambda^2\mu^5 - 72\,\Delta t^3\lambda\mu^6 + 9\,\Delta t^3\mu^7 \\ &- 900\,\gamma\Delta t^2\lambda^6 + 3960\,\gamma\Delta t^2\lambda^5\mu - 5940\,\gamma\Delta t^2\lambda^4\mu^2 + 3060\,\gamma\Delta t^2\lambda^3\mu^3 + 360\,\gamma\Delta t^2\lambda^2\mu^4 \\ &- 540\,\gamma\Delta t^2\lambda\mu^5 - 360\,\gamma^2\Delta t\lambda^5 + 3960\,\gamma^2\Delta t\lambda^4\mu - 6840\,\gamma^2\Delta t\lambda^3\mu^2 + 3240\,\gamma^2\Delta t\lambda^2\mu^3 \\ &+ 2160\,\gamma\Delta t\lambda^5 - 11880\,\gamma\Delta t\lambda^4\mu + 22680\,\gamma\Delta t\lambda^3\mu^2 - 18360\,\gamma\Delta t\lambda^2\mu^3 + 5400\,\gamma\Delta t\lambda\mu^4 \\ &+ 8640\,\gamma\lambda^4 - 25920\,\gamma\lambda^3\mu + 25920\,\gamma\lambda^2\mu^2 - 8640\,\gamma\lambda\mu^3\Big] + \mathcal{O}(\Delta t^6).\end{aligned} \tag{46}$$

For $\gamma = 0$, $err_{\lambda,\mathrm{ODD}} \sim \mathcal{O}(\Delta t^5)$ and $err_{\mu,\mathrm{ODD}} \sim \mathcal{O}(\Delta t^5)$, giving global fourth-order accuracy. However, for $\gamma \neq 0$, $err_{\lambda,\mathrm{ODD}} \sim \mathcal{O}(\Delta t^2)$ and $err_{\mu,\mathrm{ODD}} \sim \mathcal{O}(\Delta t^2)$, corresponding to global first-order accuracy.

The characteristic polynomial for the SG1 scheme is

$$\begin{aligned}P_{\mathrm{SG1}}(E) = E - 1 - \Delta t\lambda + \frac{1}{2}\,(\gamma - 1)\Delta t^2\lambda^2 - \frac{1}{6}\left(\gamma^2 - 2\,\gamma + 1\right)\Delta t^3\lambda^3 \\ + \frac{1}{24}\left(\gamma^3 - 3\,\gamma^2 + 3\,\gamma - 1\right)\Delta t^4\lambda^4,\end{aligned} \tag{47}$$

and the particular polynomial is

$$
\begin{aligned}
Q_{\mathrm{SG1}}(E) = & -\frac{1}{24}\left(\gamma^3 - 3\,\gamma^2 + 3\,\gamma - 1\right)\Delta t^4 \lambda^3 + \frac{1}{12}\left(\gamma^2 - 2\,\gamma + 1\right)\Delta t^3 \lambda^2 - \frac{1}{6}\left(\gamma - 1\right)\Delta t^2 \lambda \\
& + \frac{1}{6}\left(E + 1\right)\Delta t + \frac{1}{12}\left(\left(\gamma^2 - 2\,\gamma + 1\right)\Delta t^3 \lambda^2 - 4\left(\gamma - 1\right)\Delta t^2 \lambda + 8\,\Delta t\right)\sqrt{E}.
\end{aligned} \tag{48}
$$

In this case, the characteristic polynomial only has a single $\sigma$-root given by

$$
\begin{aligned}
\sigma_{1,\mathrm{SG1}} = & 1 + \Delta t \lambda - \frac{1}{2}\left(\gamma - 1\right)\Delta t^2 \lambda^2 + \frac{1}{6}\left(\gamma^2 - 2\,\gamma + 1\right)\Delta t^3 \lambda^3 \\
& - \frac{1}{24}\left(\gamma^3 - 3\,\gamma^2 + 3\,\gamma - 1\right)\Delta t^4 \lambda^4,
\end{aligned} \tag{49}
$$

from which the stability region may be inferred. Figure 1b shows plots of the SG1 stability region for several values of $\gamma$. Similar to the ODD scheme, $\gamma = 0$ results in the same stability region as the ERK4 scheme (it can be shown that $\sigma_{1,\mathrm{SG1}} = \sigma_{1,\mathrm{ERK4}}$ if $\gamma = 0$). In addition, $\gamma = 1$ results in the same stability region as the forward Euler (FE) scheme.

The local errors in the transient and particular solutions for the SG1 scheme are

$$
\begin{aligned}
err_{\lambda,\mathrm{SG1}} = & \frac{1}{120}\,\Delta t^5 \lambda^5 + \frac{1}{24}\left(\gamma^3 \lambda^4 - 3\,\gamma^2 \lambda^4 + 3\,\gamma \lambda^4\right)\Delta t^4 + \frac{1}{2}\,\gamma \Delta t^2 \lambda^2 - \frac{1}{6}\left(\gamma^2 \lambda^3 - 2\,\gamma \lambda^3\right)\Delta t^3 \\
& + \mathcal{O}(\Delta t^6)
\end{aligned} \tag{50}
$$

and

$$
\begin{aligned}
err_{\mu,\mathrm{SG1}} = & \frac{\Delta t^2 \mu}{2880\left(\lambda - \mu\right)^4}\Big[20\,\gamma^4 \Delta t^3 \lambda^7 - 40\,\gamma^4 \Delta t^3 \lambda^6 \mu + 200\,\gamma^4 \Delta t^3 \lambda^5 \mu^2 - 60\,\gamma^3 \Delta t^3 \lambda^7 \\
& + 100\,\gamma^3 \Delta t^3 \lambda^6 \mu - 320\,\gamma^3 \Delta t^3 \lambda^5 \mu^2 + 220\,\gamma^3 \Delta t^3 \lambda^4 \mu^3 + 60\,\gamma^3 \Delta t^3 \lambda^3 \mu^4 + 60\,\gamma^2 \Delta t^3 \lambda^7 \\
& - 50\,\gamma^2 \Delta t^3 \lambda^6 \mu - 20\,\gamma^2 \Delta t^3 \lambda^5 \mu^2 - 40\,\gamma^2 \Delta t^3 \lambda^4 \mu^3 + 40\,\gamma^2 \Delta t^3 \lambda^3 \mu^4 + 10\,\gamma^2 \Delta t^3 \lambda^2 \mu^5 \\
& - 20\,\gamma \Delta t^3 \lambda^7 + 240\,\gamma^3 \Delta t^2 \lambda^5 \mu - 40\,\gamma \Delta t^3 \lambda^6 \mu - 120\,\gamma^3 \Delta t^2 \lambda^4 \mu^2 + 240\,\gamma \Delta t^3 \lambda^5 \mu^2 \\
& - 120\,\gamma^3 \Delta t^2 \lambda^3 \mu^3 - 280\,\gamma \Delta t^3 \lambda^4 \mu^3 + 100\,\gamma \Delta t^3 \lambda^3 \mu^4 - 360\,\gamma^2 \Delta t^2 \lambda^5 \mu + 30\,\Delta t^3 \lambda^6 \mu \\
& + 600\,\gamma^2 \Delta t^2 \lambda^4 \mu^2 - 100\,\Delta t^3 \lambda^5 \mu^2 - 120\,\gamma^2 \Delta t^2 \lambda^3 \mu^3 + 125\,\Delta t^3 \lambda^4 \mu^3 - 120\,\gamma^2 \Delta t^2 \lambda^2 \mu^4 \\
& - 76\,\Delta t^3 \lambda^3 \mu^4 + 28\,\Delta t^3 \lambda^2 \mu^5 - 8\,\Delta t^3 \lambda \mu^6 + \Delta t^3 \mu^7 + 120\,\gamma \Delta t^2 \lambda^5 \mu - 360\,\gamma \Delta t^2 \lambda^4 \mu^2 \\
& + 360\,\gamma \Delta t^2 \lambda^3 \mu^3 - 120\,\gamma \Delta t^2 \lambda^2 \mu^4 + 240\,\gamma^2 \Delta t \lambda^5 - 720\,\gamma^2 \Delta t \lambda^3 \mu^2 + 480\,\gamma^2 \Delta t \lambda^2 \mu^3 \\
& - 240\,\gamma \Delta t \lambda^5 + 480\,\gamma \Delta t \lambda^4 \mu - 480\,\gamma \Delta t \lambda^2 \mu^3 + 240\,\gamma \Delta t \lambda \mu^4 + 1440\,\gamma \lambda^4 - 4320\,\gamma \lambda^3 \mu \\
& + 4320\,\gamma \lambda^2 \mu^2 - 1440\,\gamma \lambda \mu^3\Big] + \mathcal{O}(\Delta t^6),
\end{aligned} \tag{51}
$$

respectively. For $\gamma = 0$, the leading terms in $err_{\lambda,\mathrm{SG1}}$ and $err_{\mu,\mathrm{SG1}}$ are of order $\mathcal{O}(\Delta t^5)$, giving fourth-order global accuracy. However, if $\gamma \neq 0$, the leading error terms are of order $\mathcal{O}(\Delta t^2)$, giving first-order global accuracy.

It is difficult to theoretically estimate the value of $\gamma$ for two reasons. First, $\mathbf{D}$ and $\mathbf{V}$ involve quantities that are interpolated to tracer position $q$. Second, the velocity field is calculated from the stream function as the result of solving equation 8. Accordingly, the explicit dependence of $\mathbf{V}$ on $\mathbf{y}$ is not known. However, we did not encounter any apparent instabilities using a time step size corresponding to 99% of the ERK4 scheme Courant limit along the negative real axis (diffusion dominance was presumed) with the ODD and SG1 schemes. Based upon this observation, the value of $\mathrm{Im}[\gamma]$ may be small, while $\mathrm{Re}[\gamma]$ may lie in the approximate interval $(0, 0.5]$ because the stability region along the negative real axis does not deteriorate. In this situation, the theoretical expectation is that the ERK4 scheme is fourth-order accurate, while the ODD and SG1 schemes are first-order accurate. In the following sections, all Courant factors refer to the stability region of the ERK4 scheme along the negative real axis.

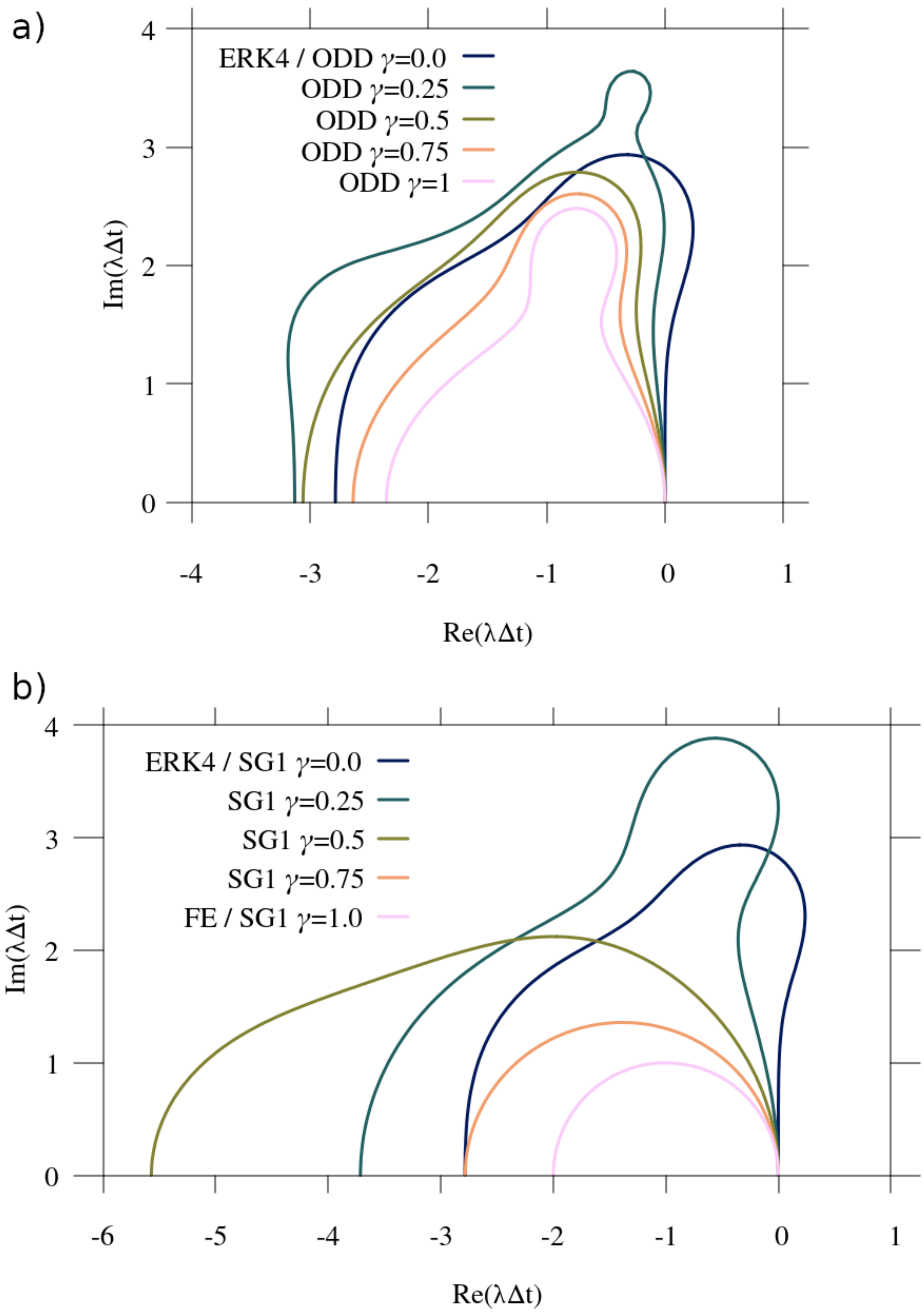



**Figure 1.** Plot showing the stability region for a) the ODD scheme and b) the SG1 scheme for different values of $\gamma$. The stability curves for a and b are determined using $|\sigma_{1,\mathrm{ODD}}| = 1$ and $|\sigma_{1,\mathrm{SG1}}| = 1$, respectively (see equations 44 and 49). For $\gamma = 0$, the stability region in parts a and b is equivalent to that of the ERK4 scheme. For part b, $\gamma = 1$ results in a stability region equivalent to that of the forward Euler (FE) scheme.

#### 2.4.2 Analysis in the context of additive Runge–Kutta methods

Runge–Kutta methods may be summarized in the form of Butcher tables, which allow scheme details to be notated in a compact tabular form (Butcher, 2016). The Butcher tables for elemental RK methods take the following form:

$$\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^\intercal \end{array}, \tag{52}$$

where $\mathbf{A} \in \mathbb{R}^{s\times s}$ is the RK matrix, $\mathbf{c} \in \mathbb{R}^s$ is a vector of nodes (or abscissae), and $\mathbf{b} \in \mathbb{R}^s$ is a vector of quadrature weights, with $s$ the total number of stages. Methods can be formed from the tableau entries using

$$\mathbf{y}^{n+1} = \mathbf{y}^n + \Delta t \sum_{i=1}^{s} b_i \mathbf{k}_i, \tag{53}$$

with

$$\mathbf{k}_i = \mathbf{F}\left(t^n + c_i \Delta t, \mathbf{y}^n + \Delta t \sum_{j=1}^{s} a_{ij}\mathbf{k}_i\right), \; i = 1, 2, \dots, s. \tag{54}$$

The Butcher tableau for ERK4 is

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}, \tag{55}$$

where blank entries take a value of zero. ERK methods are characterized by strictly lower triangular table matrices $\mathbf{A}$. Accordingly for such methods, the upper bound of the sum in equation 54 can be replaced by $i-1$.

The ODD and SG1 schemes fall under the umbrella of 2–additive Runge–Kutta ($\text{ARK}_2$) methods, which involve a combination of two elemental RK schemes. $\text{ARK}_2$ methods may be summarized using the partitioned Butcher tableau given by

$$\begin{array}{c|cc|c} \mathbf{c}^{[1]} & \mathbf{A}^{[1]} & \mathbf{c}^{[2]} & \mathbf{A}^{[2]} \\ \hline & \mathbf{b}^{[1]\intercal} & & \mathbf{b}^{[2]\intercal} \end{array}, \tag{56}$$

where $\mathbf{A}^{\{[1],[2]\}}$ are the Runge–Kutta matrices, $\mathbf{c}^{\{[1],[2]\}}$ are node vectors, $\mathbf{b}^{\{[1],[2]\}}$ are the weight vectors, and bracketed superscripts correspond to each elemental scheme. $\text{ARK}_2$ schemes are defined using

$$\mathbf{y}^{n+1} = \mathbf{y}^n + \widetilde{\Delta t} \sum_{i=1}^{s} \left[b_i^{[1]}\mathbf{k}_i^{[1]} + b_i^{[2]}\mathbf{k}_i^{[2]}\right], \tag{57}$$

with

$$\mathbf{k}_i^{\{[1],[2]\}} = \mathbf{F}^{\{[1],[2]\}}\left(t^n + c_i^{\{[1],[2]\}}\widetilde{\Delta t}, \mathbf{y}^n + \widetilde{\Delta t}\sum_{i=1}^{s} a_{ij}^{\{[1],[2]\}}\mathbf{k}_j^{\{[1],[2]\}}\right), \; i = 1, 2, \dots, s, \tag{58}$$

where $\widetilde{\Delta t}$ is the time step size. In this study, $\mathbf{F}^{[1]} = \mathbf{D}$ and $\mathbf{F}^{[2]} = \mathbf{V}$.

$\text{ARK}_2$ methods are most commonly used as implicit-explicit methods, where one term on the right side is considered stiff and treated implicitly, whereas the other term is considered non-stiff and treated explicitly (Ascher et al., 1997). Such a strategy is employed to enhance the overall feasibility or efficiency of performing a given time integration.

The Butcher tables corresponding to the ODD and SG1 schemes are

$$
\begin{array}{c|cccccccc}
0 & & & & & & & & \\
\frac{1}{4} & \frac{1}{4} & & & & & & & \\
\frac{1}{4} & 0 & \frac{1}{4} & & & & & & \\
\frac{1}{2} & 0 & 0 & \frac{1}{2} & & & & & \\
\frac{1}{2} & \frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{12} & & & & \\
\frac{3}{4} & \frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{12} & \frac{1}{4} & & & \\
\frac{3}{4} & \frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{12} & 0 & \frac{1}{4} & & \\
1 & \frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{12} & 0 & 0 & \frac{1}{2} & \\
\hline
 & \frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{12} & \frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{12}
\end{array}
\quad
\begin{array}{c|cccccccc}
0 & & & & & & & & \\
0 & & & & & & & & \\
0 & & & & & & & & \\
0 & & & & & & & & \\
0 & & & & & & & & \\
\frac{3}{4} & \frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{12} & \frac{1}{4} & & & \\
\frac{3}{4} & \frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{12} & 0 & \frac{1}{4} & & \\
1 & \frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{12} & 0 & 0 & \frac{1}{2} & \\
\hline
 & \frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{12} & \frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{12}
\end{array}
\tag{59}
$$

and

$$
\begin{array}{c|cccc}
0 & & & & \\
\frac{1}{2} & \frac{1}{2} & & & \\
\frac{1}{2} & 0 & \frac{1}{2} & & \\
1 & 0 & 0 & 1 & \\
\hline
 & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6}
\end{array}
\quad
\begin{array}{c|cccc}
0 & & & & \\
0 & & & & \\
0 & & & & \\
0 & & & & \\
\hline
 & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6}
\end{array}
\; . \tag{60}
$$

Accordingly, both the ODD and SG1 schemes correspond to explicit-explicit $\mathrm{ARK}_2$ methods.

To simplify the theory that follows, the ODD tableau has been scaled so that nodal values range from zero to unity. However, the ODD scheme as presented in equations 21–28 can be recovered by applying a time step size of $\widetilde{\Delta t} = 2\Delta t$. In contrast, we have $\widetilde{\Delta t} = \Delta t$ for SG1. We also note that the Butcher tables in equations 55, 59, and 60 do not require $\mathbf{V}$ to be autonomous as in equation 13. However, we may apply these Butcher tables for autonomous $\mathbf{V}$ without difficulty.

The accuracy of elemental RK schemes depends on the satisfaction of certain order conditions (Butcher, 2016). We list the order conditions up to fourth order in table 2. The ERK4 scheme satisfies all order conditions in table 2 and accordingly is a fourth-order method.

The order of accuracy of an ARK method depends on the order conditions of its constituent elemental schemes $\mathbf{A}^{[\ell]}$, $\mathbf{b}^{[\ell]}$, and $\mathbf{c}^{[\ell]}$, $\ell = 1, 2, \ldots, N$, as in equation 56 (for $N = 2$), as well as additional coupling conditions (Kennedy & Carpenter, 2003). We may apply the conditions in table 2 to the constituent Butcher tables in equations 59–60 to determine the orders of the elemental schemes for ODD and SG1. The left tables in equations 59–60 satisfy order conditions up to fourth order. This is expected because these elemental schemes correspond to ERK4. However, the right tables in equations 59–60 only satisfy order conditions up to first order. It can further be shown that an $N$-additive RK method will have at most second-order coupling error if $\mathbf{b}^{[1]} = \mathbf{b}^{[2]} = \cdots = \mathbf{b}^{[N]}$ or $\mathbf{c}^{[1]} = \mathbf{c}^{[2]} = \cdots = \mathbf{c}^{[N]}$ (Kennedy & Carpenter, 2003). The ODD and SG1 schemes satisfy $\mathbf{b}^{[1]} = \mathbf{b}^{[2]}$ (and $\mathbf{c}^{[1]} \neq \mathbf{c}^{[2]}$), which permits second-order coupling. However, the accuracies of the ODD and SG1 schemes are limited by their first-order elemental schemes, which dominate their leading error terms. Accordingly, both ODD and SG1 are first-order accurate $\mathrm{ARK}_2$ methods.

Formally, the order conditions are derived using the test equation $d\mathbf{y}/dt = \lambda\mathbf{y}$. This differs from the analysis in section 2.4.1, where a non-linear time dependence was

**Table 2.** Order conditions of Runge–Kutta methods[a]

| Order | Condition |
|---|---|
| 1 | $\sum_j b_j = 1$ |
| 2 | $\sum_j b_j c_j = \frac{1}{2}$ |
| 3 | $\sum_j b_j c_j^2 = \frac{1}{3}$ |
| | $\sum_{jl} b_j A_{jl} c_l = \frac{1}{6}$ |
| 4 | $\sum_j b_j c_j^3 = \frac{1}{4}$ |
| | $\sum_{jl} b_j c_j A_{jl} c_l = \frac{1}{8}$ |
| | $\sum_{jl} b_j A_{jl} c_l^2 = \frac{1}{12}$ |
| | $\sum_{jlm} b_j A_{jl} A_{lm} c_m = \frac{1}{24}$ |

[a] Order conditions for elemental Runge–Kutta methods up to fourth order.

included. This distinction may be of importance for problems 3 and 4 due to a non-linear time-dependence for $H$.

The stability function of an elemental RK method is

$$\sigma(\mathfrak{z}) = \frac{\det(\mathbf{I} - \mathfrak{z}\mathbf{A} + \mathfrak{z}\mathbf{e}\mathbf{b}^\intercal)}{\det(\mathbf{I} - \mathfrak{z}\mathbf{A})}, \tag{61}$$

where $\mathfrak{z} = \lambda\Delta t$ and $\mathbf{e}$ is a vector of ones (Wanner & Hairer, 1996). For the ERK4 scheme, it can be shown that equation 61 produces a result equivalent to equation 39.

The stability function for an ARK$_2$ method is given by

$$\sigma(\mathfrak{z}^{[1]}, \mathfrak{z}^{[2]}) = \frac{\det(\mathbf{I} - \mathfrak{z}^{[1]}\mathbf{A}^{[1]} - \mathfrak{z}^{[2]}\mathbf{A}^{[2]} + \mathfrak{z}^{[1]}\mathbf{e}\mathbf{b}^{[1]\intercal} + \mathfrak{z}^{[2]}\mathbf{e}\mathbf{b}^{[2]\intercal}))}{\det(\mathbf{I} - \mathfrak{z}^{[1]}\mathbf{A}^{[1]} - \mathfrak{z}^{[2]}\mathbf{A}^{[2]})}, \tag{62}$$

where $\mathfrak{z}^{[1],[2]} = \lambda^{[1],[2]}\Delta t$, and $\lambda^{[1],[2]}$ are the eigenvalues of each elemental scheme (Kennedy & Carpenter, 2003). In this study, we have $\lambda^{[1]} = \lambda_D$ and $\lambda^{[2]} = \lambda_V$. For the ODD scheme, equation 62 gives the square of equation 44 because the stability function determined via ARK theory corresponds to one iteration of the scheme (over two successive time steps). For the SG1 scheme, equation 62 gives a result equivalent to equation 49. Accordingly, the stability functions from ARK analysis are consistent with those of the polynomial analysis in section 2.4.1.

### 2.5 Test Problem Setup

We examine the performance of the ERK4, ODD, and SG1 velocity update schemes on four test problems. All the problems are governed by equations 1–4. They differ in the choices of parameter values, boundary conditions, and initial conditions, as described in this section and corresponding to different physical situations. Details of these choices are now given.

#### *2.5.1 Parameters*

Relevant physical parameters for all test problems are summarized in table 3, where the domain is given by $[0, \alpha]\times[0, 1]$.

**Table 3.** Physical parameters for test problems[a]

| Problem | $Ra_T$ | $Ra_C$ | $\alpha$ | $H$ |
|---|---|---|---|---|
| 1 | 0 | $-1$ | 0.9142 | N/A |
| 2 | $10^5$ | 0 | 0.9142 | 0 |
| 3 | $10^5$ | 0 | 1 | $H(x, z, t)$ |
| 4 | $10^5$ | $5\times10^4$ | 1 | $H(x, z, t)$ |

[a] Problems 3 and 4 involve variable internal heating rates given by equation 76 and available via GitHub/Zenodo (see section 3.4.1), respectively.

#### *2.5.2 Initial Conditions*

For problem 1, the initial composition field (van Keken et al., 1997) for tracer particle $q$ located at position $\mathbf{r} = (x, z)$ is

$$C_q(x, z, t = 0) = \begin{cases} 0, & \text{if } z > 0.2 + 0.02\cos(\pi x/\alpha), \\ 1, & \text{otherwise.} \end{cases} \tag{63}$$

For problem 2, the initial temperature field for tracer particle $q$ is

$$T_q(x, z, t = 0) = \frac{1}{2}\left[1 + \tanh\left(\frac{0.2 + 0.04\cos(\pi x/\alpha) - z}{0.04}\right)\right]. \tag{64}$$

This temperature function is similar to the function for composition in equation 63, but the interface between 0 and 1 is smoothed (Bangerth et al., 2020). Including a smooth interface in the initial temperature allows us to mitigate against numerical error when computing the thermal diffusion term.

For problem 3, the initial temperature field corresponds to a conductive profile given by

$$T_q(x, z, t = 0) = -z + 1. \tag{65}$$

In this problem, the lateral temperature gradients required to initiate covection are due to a variable internal heating rate.

For problem 4, we compare our numerical solutions to an exact solution (S. J. Trim et al., 2023) that requires an initial composition field based upon the logistic function. Specifically, we have a two-layer initial composition field given by $C_q(x, z, t = 0) = C_0(z)$ with

$$C_0(z) = \left[1 + \exp\left[-2k\left(z_I - z\right)\right]\right]^{-1}, \tag{66}$$

where $k$ and $z_I$ control the thickness and vertical position of the material interface between compositions. We note that $C_q$ is defined using fractional values between zero and unity. This required some additional programming in the convection code used, which originally only allowed $C_q$ values of 0 or 1. However, this update permitted precise matching of the gradient within the compositional interface.

The initial temperature field for problem 4 is given by

$$T_q(x,z,t=0) = \frac{1}{Ra_T}\Bigg[ -\frac{\pi^3\left(\alpha^2+1\right)^2}{\alpha^3}\cos\left(\pi x/\alpha\right)\sin\left(\pi z\right) f\left(t=0\right) + Ra_C C_0(z) \\ + (Ra_T - Ra_C)(1-z)\Bigg], \tag{67}$$

where $f(t)$ is a function that influences the time dependence of the solution (see equation 77).

#### 2.5.3 Boundary Conditions

For each problem, boundary conditions for temperature and velocity are required and are summarized in table 4. For problem 4, we have isothermal top and bottom boundaries, with a basal temperature of

$$T_{bot} = \frac{Ra_C}{Ra_T}[C_0(0) - 1] + 1 \tag{68}$$

and a surface temperature of

$$T_{top} = \frac{Ra_C}{Ra_T} C_0(1). \tag{69}$$

However, for the parameters used in this study, $T_{bot} \approx 1$ and $T_{top} \approx 0$. Accordingly, we apply basal and surface temperatures of 1 and 0 in our calculations.

**Table 4.** Boundary conditions for test problems[a]

| Problem | $z=0$ | $z=1$ | $x=0$ and $x=\alpha$ |
|---|---|---|---|
| 1 | $\mathbf{v}=\mathbf{0}$ | $\mathbf{v}=\mathbf{0}$ | $u=0,\ \frac{\partial w}{\partial x}=0$ |
| 2 | $T=1,\ \mathbf{v}=\mathbf{0}$ | $T=0,\ \mathbf{v}=\mathbf{0}$ | $\frac{\partial T}{\partial x}=0,\ u=0,\ \frac{\partial w}{\partial x}=0$ |
| 3 | $T=1,\ \frac{\partial u}{\partial z}=0,\ w=0$ | $T=0,\ \frac{\partial u}{\partial z}=0,\ w=0$ | $\frac{\partial T}{\partial x}=0,\ u=0,\ \frac{\partial w}{\partial x}=0$ |
| 4 | $T=T_{bot},\ \frac{\partial u}{\partial z}=0,\ w=0$ | $T=T_{top},\ \frac{\partial u}{\partial z}=0,\ w=0$ | $\frac{\partial T}{\partial x}=0,\ u=0,\ \frac{\partial w}{\partial x}=0$ |

[a]Summary of boundary conditions for temperature and velocity for all test problems. As in section 2.1, the velocity field is expressed by the components $u$ and $w$ such that $\mathbf{v}=[u,w]$.

Figure 2 shows snapshots of $\psi$, $C$, and $T$, where applicable, for each test problem. The times shown were selected in order to demonstrate the character of evolution for each problem. Exact solutions were not available for problems 1–2, so ProjecTracer was used with the ERK4 scheme to produce the data used in the plots. For problem 1, results were obtained using a 300×300 Eulerian mesh with 60TPCV using $\Delta t = 0.5$. For problem 2, a 100×100 Eulerian mesh was used with 120 TPCV with a Courant factor of 0.495. For problems 3–4, exact solutions were used to generate the plots (see sections 3.3.1 and 3.4.1).

#### 2.5.4 Diagnostics

The root-mean-square (RMS) velocity is defined as

$$v_{\mathrm{RMS}} = \sqrt{\frac{1}{\alpha}\int_0^1\int_0^\alpha [u^2+w^2]\,dxdz}, \tag{70}$$

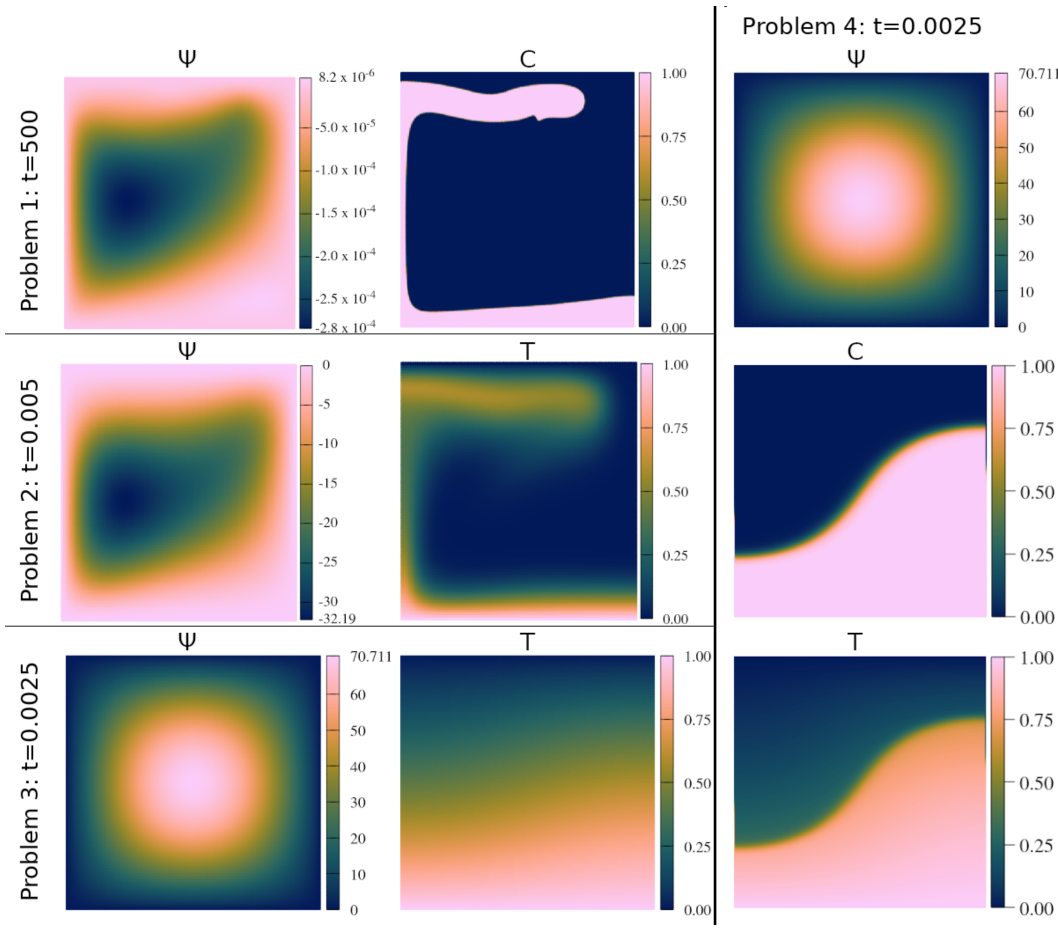



**Figure 2.** Snapshots of $\psi$, $C$, and $T$, where applicable, for test problems 1–4. Test problems are separated by black lines. Time values are also indicated for each problem.

and characterizes the overall vigor of convection.

# 3 Results

## 3.1 Problem 1: compositional RT

### *3.1.1 Generation of the reference solution*

Our goal is to quantify the efficiency of the velocity update schemes. Accordingly, we require a time-accurate reference solution for comparison with our numerical solutions. Efficiency can then be measured in terms of the CPU time required to produce a solution with a given accuracy. From the literature, we have a theoretical value for the growth rate of the primary diapir at $t = 0$ (Ramberg, 1981; van Keken et al., 1997). Because this value does not involve time evolution, however, we cannot use it to quantify time accuracy. In the absence of an exact solution for $t > 0$, the next best alternative is a numerical solution with a small error. We achieve such a solution by fixing a sufficiently fine spatial mesh and gradually refining the time step size using the ERK4 scheme until convergence of the RMS velocity at $t = 208$ is achieved. We select $t = 208$ because the peak RMS velocity, corresponding to the rise of the primary diapir, occurs around that time. We compare RMS velocities at a fixed $t$ value rather than the observed peak time because the precise peak time likely occurs between time steps, thus requiring interpolation and introducing a confound. We isolate the effect of temporal discretization by using a fixed spatial resolution while varying the time step size. Specifically, we use a 300×300 Eulerian mesh with TPCV = 60 at $t = 0$ for a total of 5 436 060 tracers.

RMS velocity data at $t = 208$ for all velocity update schemes are shown in table 5. Time step sizes were tested in decreasing factors of two ranging from 16 down to 0.0078125. Examining the $v_{\mathrm{RMS}}$ results for the ERK4 scheme reveals convergence of approximately 9 digits at the smallest time step size used. We use this value as our reference solution.

### *3.1.2 Convergence properties*

The order of accuracy of a time-stepping method refers to the rate of convergence to a solution with perfect time accuracy as the time step size is decreased. Accordingly, the order of accuracy of each velocity update scheme can be estimated by using the reference solution developed in section 3.1.1.

Corresponding to the data in table 5, figure 3 shows a logarithmic plot of the RMS velocity error at $t = 208$ relative to the reference solution versus the time step size. The slope of a best-fit line can be used as an estimate of the rate of convergence to the reference solution. For the ERK4 scheme, the theoretical convergence rate is expected to be of order four. A best-fit line through all data points gives a slope of about 2.2. The reduction in observed convergence may be due to the presence of statistical noise and roundoff error at smaller time step sizes. If we consider just the three largest time step sizes, a best-fit line has a slope of approximately 4.3.

For the ODD scheme, a best-fit line through all data points results in a slope of 0.9. However, oscillations in the $v_{\mathrm{RMS}}(t = 208)$ value are present for larger time step sizes (see table 5). This may be due to limited temporal resolution in the velocity field at larger step sizes. Specifically, because the velocity field is only updated on odd-numbered time steps, there is effectively half the temporal resolution in the RMS velocity time series compared to the ERK4 and SG1 schemes. A best-fit line through data points with $\Delta t \leq 1$ results in a slope of about 0.97. Overall, the observed order of accuracy is near unity for the ODD scheme, consistent with the theoretical results of section 2.4.

**Table 5.** Problem 1: $v_{\mathrm{RMS}}(t = 208)$ [a]

| $\Delta t$ | ERK4 | ODD | SG1 |
|---|---|---|---|
| 16 | 0.00309362096182427 | 0.00252405820744709 | 0.00277785683067033 |
| 8 | 0.00309303181796976 | 0.00309651423432836 | 0.00305160939785673 |
| 4 | 0.00309300102933752 | 0.00311164980347686 | 0.00310960787560614 |
| 2 | 0.00309300290188762 | 0.00310676676305127 | 0.00311091126105452 |
| 1 | 0.00309299863963776 | 0.00310094169178837 | 0.00310430901766157 |
| 0.5 | 0.00309299951606417 | 0.00309725893445424 | 0.00309918734358765 |
| 0.25 | 0.00309299945897165 | 0.00309519284649142 | 0.00309625268180455 |
| 0.125 | 0.00309299946739831 | 0.00309411400482291 | 0.00309466244897793 |
| 0.0625 | 0.00309299947117256 | 0.00309356192051190 | 0.00309384183391344 |
| 0.03125 | 0.00309299947117070 | 0.00309328268354388 | 0.00309342309407799 |
| 0.015625 | 0.00309299947156048 | 0.00309314150993983 | 0.00309321219426920 |
| 0.0078125 | 0.00309299947143122 | 0.00309307078015074 | 0.00309310626632360 |
| reference | 0.00309299947143122 | 0.00309299947143122 | 0.00309299947143122 |

[a]Summary of RMS velocities at $t = 208$ for different time step sizes using the velocity update schemes of table 1. The reference value used for absolute difference error calculations is indicated by “reference” in the first column.

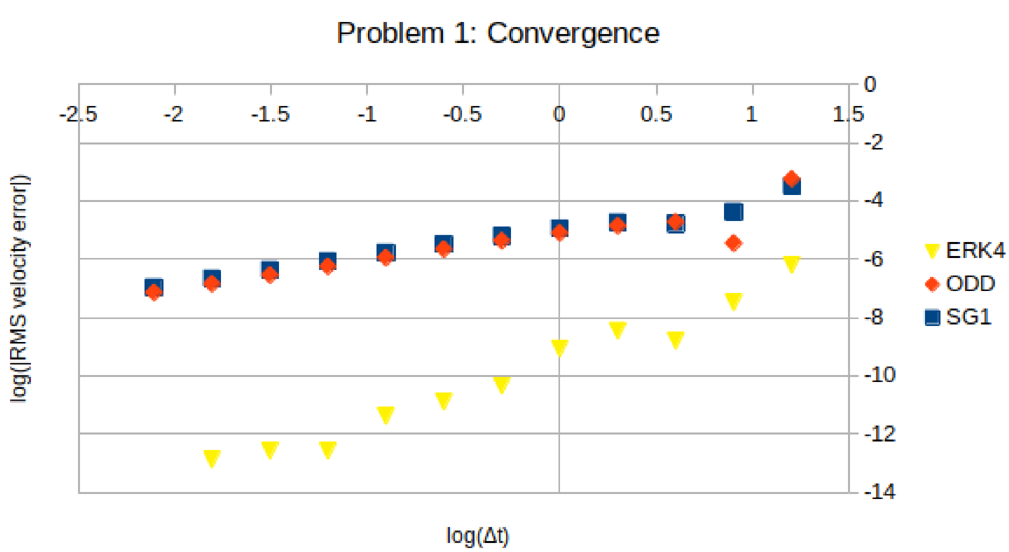



**Figure 3.** Base-10 logarithmic plot showing the RMS velocity error magnitude versus the time step size for problem 1. The RMS velocity error is the difference between numerical and reference values for $v_{\mathrm{RMS}}$ at $t = 208$. The ERK4, ODD, and SG1 schemes are shown using triangles, diamonds, and squares, respectively.

For the SG1 scheme, using all data points results in a best-fit slope of about 0.92. For $\Delta t \leq 1$, the best-fit slope is about 0.97. Similar to the ODD scheme, the SG1 scheme has an observed order of accuracy near unity, which is consistent with our theoretical findings.

#### *3.1.3 Accuracy versus computational expense*

The primary factors for computational expense in particle mantle convection models are the costs of solving for the velocities and advecting the tracer particles. We discuss the accuracy of the different velocity update schemes in both of these contexts.

Figure 4a shows a plot of $v_{\mathrm{RMS}}$ error relative to the reference solution versus the total number of velocity updates (i.e., over the entire model evolution) for all schemes. We see that the ERK4 scheme produces smaller errors compared to the other schemes for a given number of total velocity updates. The improvement becomes significant for a velocity update count greater than 100, with a maximum improvement of about 6 digits toward the reference solution.

For a velocity update count greater than 100, the ODD and SG1 schemes have similar performance. Specifically, the SG1 produces slightly smaller errors. For a velocity update count less than 100, the ODD scheme has oscillatory behavior due to reduced temporal resolution in the $v_{\mathrm{RMS}}$ time series.

Figure 4b shows a plot of the $v_{\mathrm{RMS}}$ error compared to the total number of time integration stages (i.e., over the entire model duration), used to quantify the cost of particle advection, in logarithmic space. We observe that the ERK4 scheme offers between 2–6 digits of improvement toward the reference solution compared to the performance of the other schemes. As before, the ODD and SG1 schemes give similar errors. However, the ODD scheme has a slight advantage for larger stage counts.

### 3.2 Problem 2: thermal RT

#### *3.2.1 Reference solution*

Similar to problem 1, we require a time-accurate reference solution in order to analyze the accuracy of the velocity update schemes. An exact solution is not available, so we use a solution with a small temporal discretization error. As before, we use the ERK4 scheme and gradually decrease the time step size until convergence. For this problem, we use a 100×100 Eulerian grid with 120 TPCV. Fixing the spatial resolution allows us to isolate the effect of varying the time step size and velocity update schemes.

Due to thermal diffusion, the time step size is stability limited. Accordingly, we choose our largest time step size at 99% of the value predicted by stability analysis for the ERK4 scheme, corresponding to a Courant number of 0.99. To find the reference solution, we reduce the time step size by factors of two until the RMS velocity at a specific time converges over multiple digits. We select $t = 0.001987650065548$, which is slightly after the first peak in the $v_{\mathrm{RMS}}$ time series. This particular value of $t$ is available without interpolation for all time step sizes tested given the Courant limit for this spatial discretization. Results for all update schemes are shown in table 6. For the ERK4 scheme, we observe 6 or 7 digits of convergence for the RMS velocity for a Courant number as low as 0.007734375 and from which we take our reference solution.

The reference solution should also not be sensitive to perturbations in spatial resolution. Tests with a 200×200 Eulerian grid with 120 TPCV produced a $v_{\mathrm{RMS}}$ value within about 0.01% of the reference value in table 6.

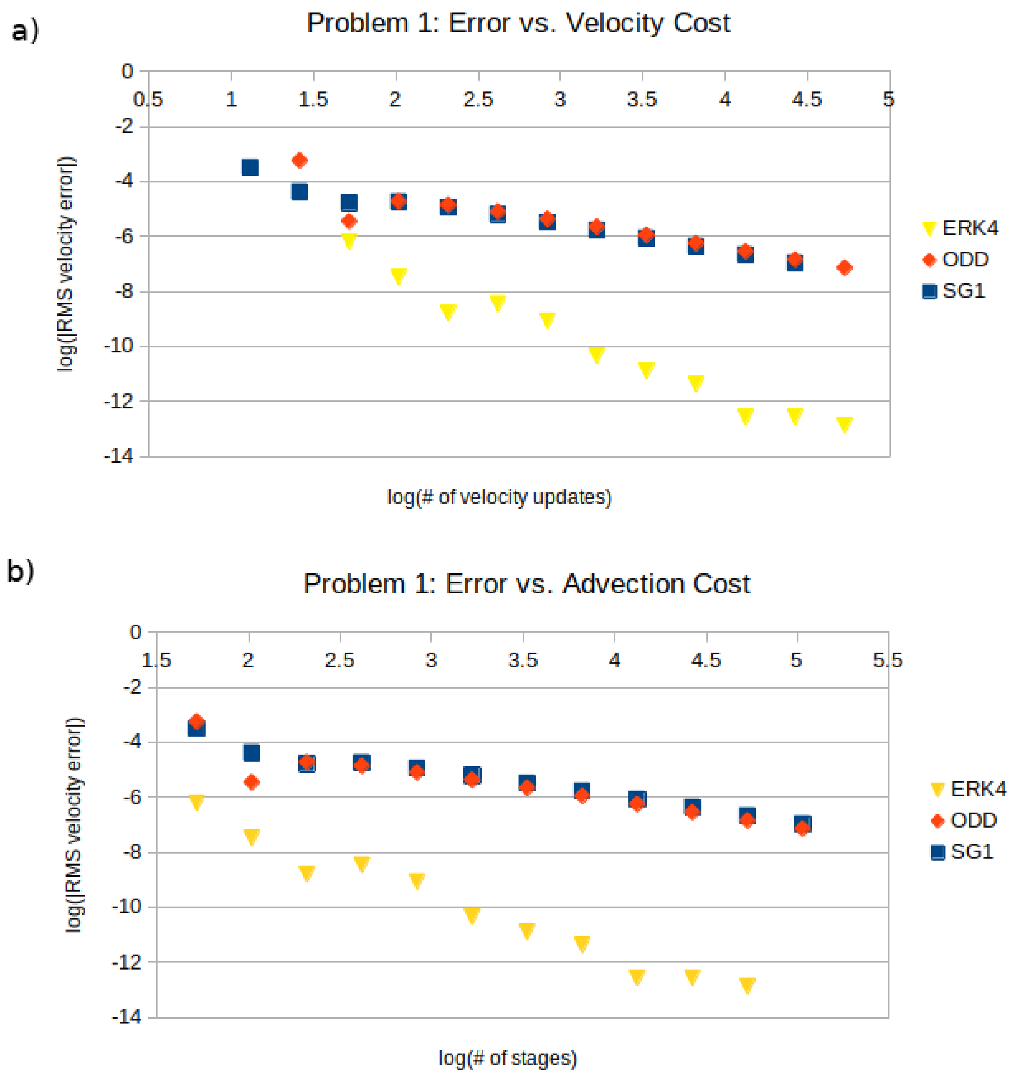



**Figure 4.** Base-10 logarithmic plots of RMS velocity error magnitude versus a) the total number of velocity updates and b) the total number of time integration stages for problem 1. The RMS velocity error is the difference between numerical and reference values for $v_{\mathrm{RMS}}$ at $t = 208$. The ERK4, ODD, and SG1 schemes are shown using triangles, diamonds, and squares, respectively.

**Table 6.** Problem 2: $v_{\mathrm{RMS}}(t = 0.001987650065548)$ [a]

| Courant Number | ERK4 | ODD | SG1 |
|---|---|---|---|
| 0.99 | 292.634869358049 | 297.882135309721 | 300.507923082747 |
| 0.495 | 292.634020001879 | 295.267843308958 | 296.587811356537 |
| 0.2475 | 292.633722486234 | 293.952636305863 | 294.613137896606 |
| 0.12375 | 292.633905954233 | 293.293840802325 | 293.624206364686 |
| 0.061875 | 292.633863109873 | 292.963909646566 | 293.129065316948 |
| 0.0309375 | 292.633862447062 | 292.798905424038 | 292.881467444705 |
| 0.01546875 | 292.633843171016 | 292.716374993738 | 292.757651607947 |
| 0.007734375 | 292.633851343382 | 292.675119792523 | 292.695752487351 |
| 0.0038671875 | – | 292.654483453064 | 292.664799916902 |
| 0.00193359375 | – | 292.644165807179 | 292.649324777993 |
| reference | 292.633851343382 | 292.633851343382 | 292.633851343382 |

[a]Summary of RMS velocities at $t = 0.001987650065548$ for different Courant numbers using the velocity update schemes of table 1. The reference value for error calculations is indicated by "reference" in the first column. Courant numbers are relative to the ERK4 scheme.

#### *3.2.2 Convergence rates*

Figure 5 shows a plot of the $v_{\mathrm{RMS}}$ error compared to the reference value versus the time step size in logarithmic space for all velocity update schemes. The slope of a best-fit line in this plot can be used to estimate the observed convergence rate toward the reference solution.

The ERK4 scheme shows superlinear convergence. Specifically, considering the entire range of time step sizes, the observed order convergence is about 1.15. Unlike problem 1, time step sizes in this problem are stability limited due to the presence of thermal diffusion. At the stability limit, the RMS velocity appears to be near a stagnant regime where noise begins to limit convergence to the reference solution. This difficulty is reduced at larger time step sizes. For instance, if we only consider the two largest time step sizes, the estimated order of convergence for the ERK4 scheme increases to about 2.6. However, we cannot test larger time step sizes due to numerical instability.

For this problem, we observe an order of convergence of approximately unity for both the ODD and SG1 schemes, consistent with our theoretical results.

#### *3.2.3 Accuracy and computational expense*

Figure 6a shows the RMS velocity error compared to the reference solution versus the total number of velocity updates used in logarithmic space for all velocity updates schemes. We observe that for a fixed number of velocity updates, the ERK4 scheme outperforms the other schemes by about four digits of accuracy compared to the reference solution. Also, the ODD and SG1 schemes achieve a similar level of accuracy given the

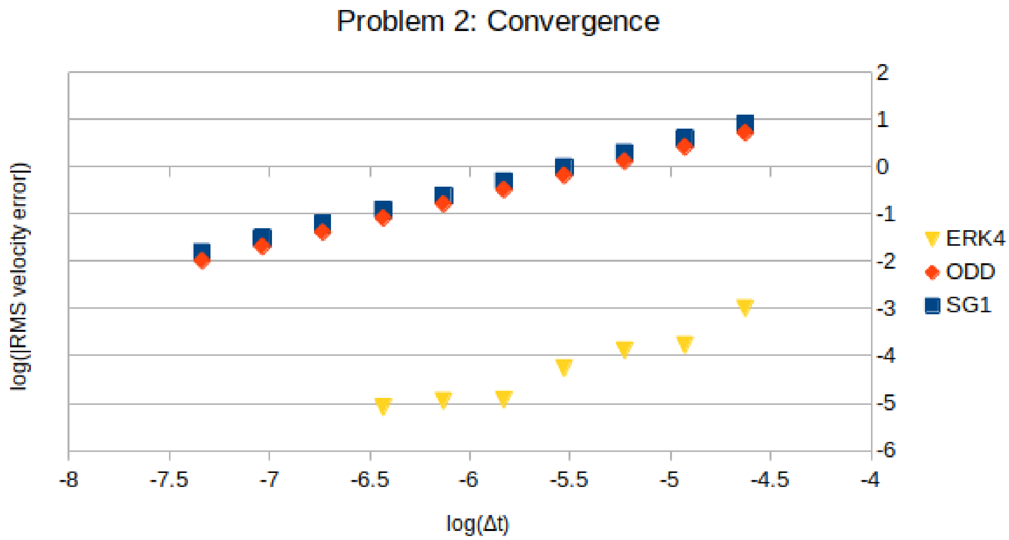



**Figure 5.** Base-10 logarithmic plot showing RMS velocity error magnitude versus the time step size for problem 2. The RMS velocity error is the difference between numerical and reference values for $v_{\mathrm{RMS}}$ at $t = 0.001987650065548$. The ERK4, ODD, and SG1 schemes are shown using triangles, diamonds, and squares, respectively.

same number of velocity updates. The SG1 scheme performs slightly better than the ODD scheme for all velocity update counts.

Figure 6b shows the RMS velocity error against the total number of time integration stages using a logarithmic scale. In the context of particle advection cost, the ERK4 scheme is also superior to the other schemes. In this case, the ODD scheme produces slightly smaller errors compared to the SG1 scheme for a given advection cost.

## 3.3 Problem 3: thermal convection

### *3.3.1 Deriving an exact solution*

Our strategy for deriving a problem with an exact solution is to select a reasonable stream function and then use a corresponding variable internal heating rate to satisfy the governing equations. We begin by selecting a function for $\psi$ that satisfies the boundary conditions for this problem (impermeable, free-slip boundaries), given by

$$\psi(x, z, t) = 100 \sin(100\pi t) \sin(\pi x) \sin(\pi z). \tag{71}$$

This function is based on the particle advection test described in Appendix C of van Keken et al. (1997). However, we have added a time-dependent amplitude with a period of 0.01 nondimensional time units in order to establish a time-dependent flow. Using equation 9, the corresponding components of velocity are

$$u(x, z, t) = 100\pi(100\pi t) \sin(\pi x) \cos(\pi z) \tag{72}$$

and

$$w(x, z, t) = -100\pi \sin(100\pi t) \cos(\pi x) \sin(\pi z). \tag{73}$$

Accordingly, the RMS velocity is

$$v_{\mathrm{RMS}}(t) = 50\pi\sqrt{2}|\sin(100\pi t)| \tag{74}$$

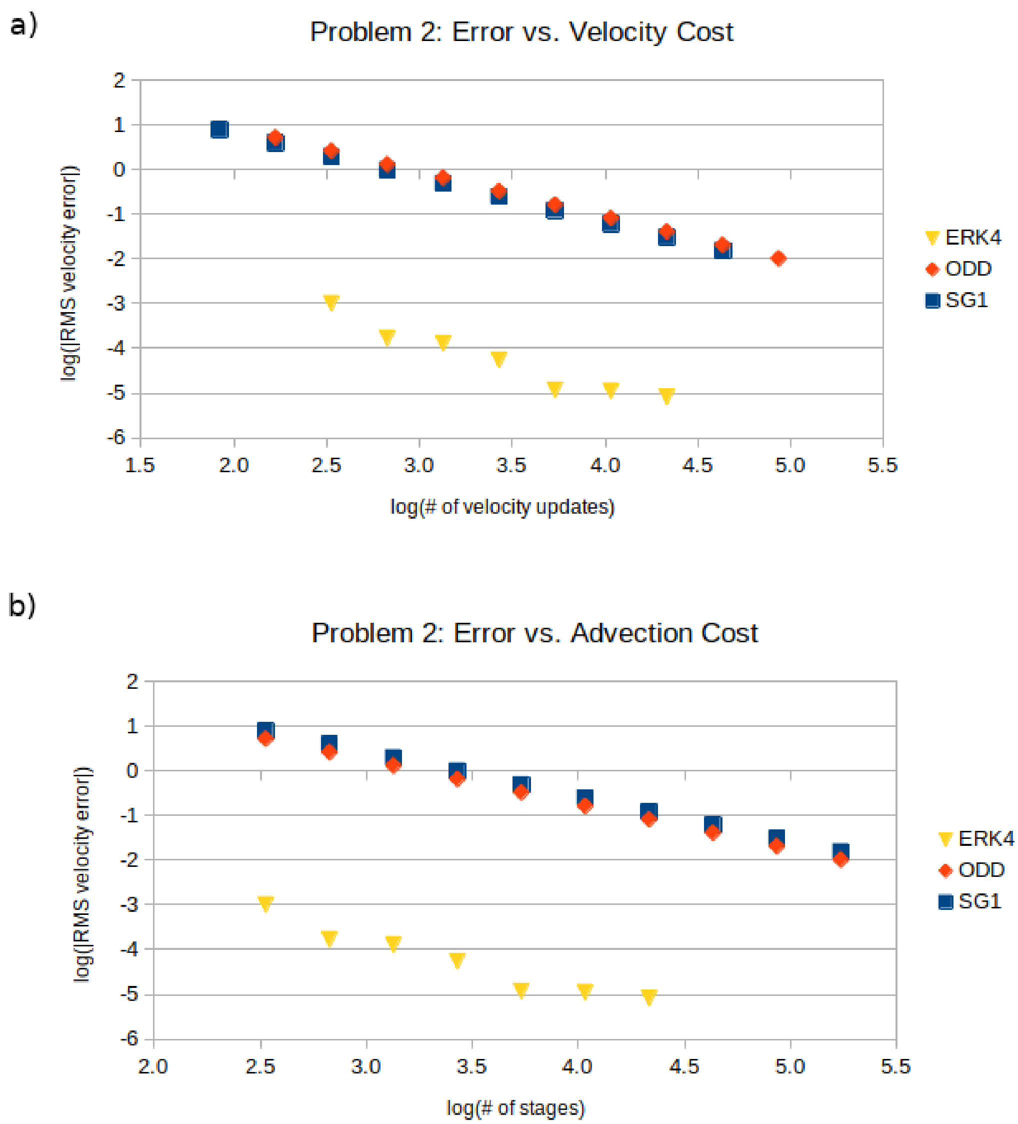



**Figure 6.** Base-10 logarithmic plots of RMS velocity error magnitude versus a) the total number of velocity updates and b) the total number of time integration stages for problem 2. The RMS velocity error is the difference between numerical and reference values for $v_{\mathrm{RMS}}$ at $t = 0.001987650065548$. The ERK4, ODD, and SG1 schemes are shown using triangles, diamonds, and squares, respectively.

Next, we substitute equation 71 into equation 8, solve for $\partial T/\partial x$, and integrate to get an expression for $T$:

$$T(x,z,t) = -\frac{\pi^3}{250}\sin(100\pi t)\cos(\pi x)\sin(\pi z) - z + 1. \tag{75}$$

Because the variable of integration is $x$, it is permissible to add an arbitrary function of $z$ (and possibly $t$). In this case, we choose to add $-z+1$, corresponding to a conductive profile, in order to satisfy the initial and boundary conditions for $T$.

Now, we may substitute equations 72, 73, and 75 into equation 3 and solve for $H$, giving

$$\begin{aligned} H(x,z,t) = \frac{1}{125}\Big\{&50\pi^5\sin^2(100\pi t)\cos(\pi z) - \big[50\pi^4\cos(100\pi t) \\ &- (12500\pi - \pi^5)\sin(100\pi t)\big]\cos(\pi x)\Big\}\sin(\pi z). \end{aligned} \tag{76}$$

Numerical models for this problem were computed using a 300×300 Eulerian mesh with 240 TPCV. Simulations were run until $t = 9.8175\times10^{-5}$, which was selected because it was available for all Courant numbers tested (0.0309375 to 0.99) without interpolation. We found that this simulation time was sufficient to observe the difference between the velocity update schemes.

#### *3.3.2 Convergence Rates*

For this problem (and problem 4), we define two reference values for RMS velocity with which to compare model results: "exact" and "numerical". The exact value is computed using equation 74 and incurs error only due to the use of finite-precision arithmetic. This error is negligible compared to discretization errors present in numerical models. The numerical reference value is the most accurate value obtained with our convection code, which was obtained using the ERK4 scheme with the smallest available time step size (corresponding to a Courant number of 0.0309375). RMS velocity values for $t = 9.8175\times10^{-5}$ for all velocity update schemes, with Courant numbers ranging from 0.0309375 to 0.99, and the reference values are included in table 7.

For the ERK4 scheme, the numerical RMS velocity values agree to about 8 significant figures as the Courant factor is reduced. However, the ODD and SG1 RMS velocity values only agree to about three significant figures at the lowest Courant numbers considered because of larger temporal discretization errors. Comparing the lowest Courant number (0.0309375) values to the exact value shows an agreement of 4 digits for the ERK4 scheme and 2–3 digits for the ODD and SG1 schemes. The ODD and SG1 schemes show similar agreement to the numerical reference value. Spatial discretization error ultimately limits the observed agreement between the model results and the exact value, even for small time step sizes.

Figure 7 shows logarithmic plots of RMS velocity errors relative to a) exact and b) numerical reference values versus time. As before, the slopes of best fit lines may be used to compute rates of convergence to the reference values.

For the ODD and SG1 schemes, the convergence rates are slightly less than unity (0.93 and 0.95 respectively) for the exact value and near unity for the numerical reference value. Similar to the previous problems, near first-order convergence is observed for the ODD and SG1 schemes, consistent with section 2.4.

For the ERK4 scheme, the convergence rates are near zero and 0.88 relative to the exact and numerical reference values, respectively. These rates are much lower than the theoretical fourth-order convergence because the time step sizes considered are so small that the temporal discretization error is insignificant compared to the spatial discretization error, giving stagnant behavior. As in problem 2, the maximum time step size is stability limited due to the spatial discretization of thermal diffusion (see section 4.2), so

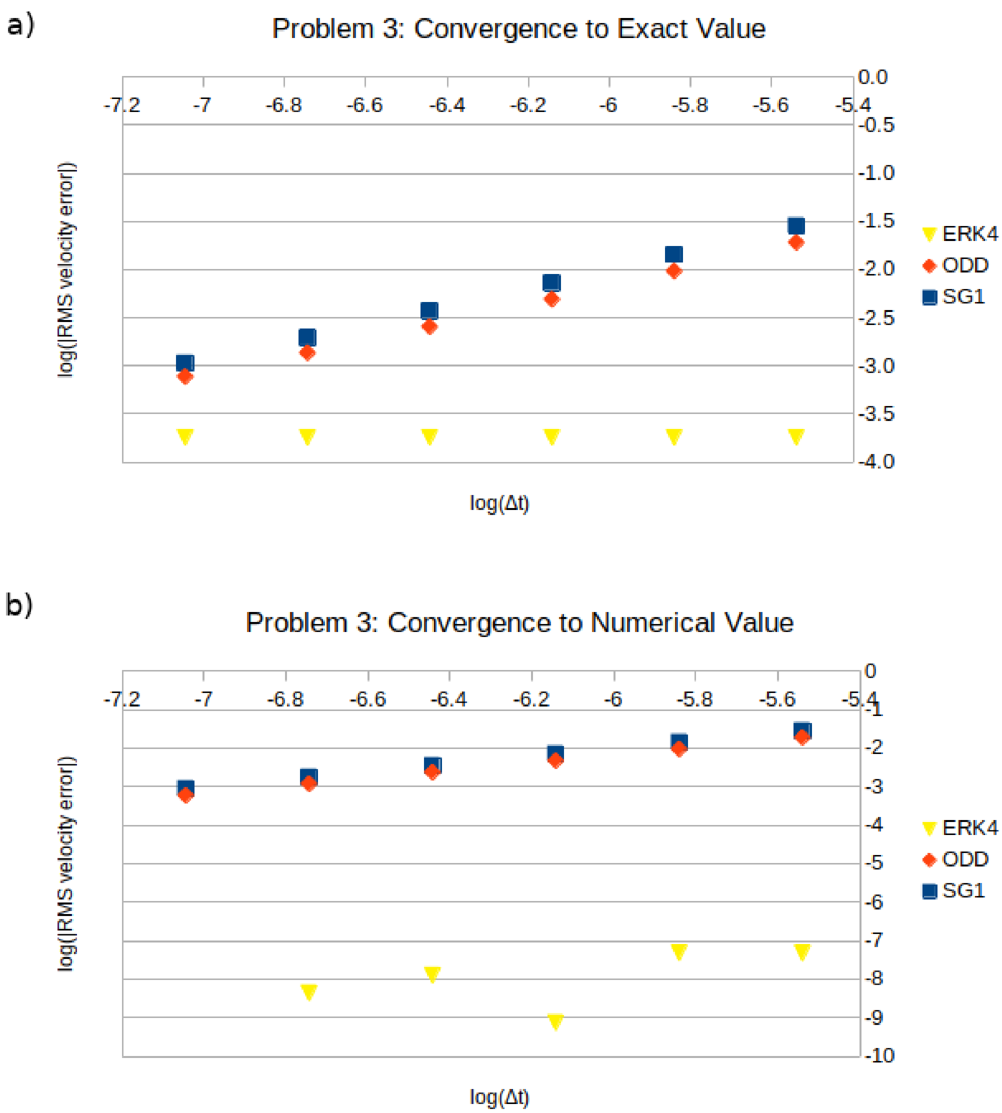



**Figure 7.** Base-10 logarithmic plot showing RMS velocity error magnitude versus the time step size for problem 3. The RMS velocity error is the difference between $v_{\mathrm{RMS}}$ model values and either a) exact or b) numerical reference values for $t = 9.8175 \times 10^{-5}$. The ERK4, ODD, and SG1 schemes are shown using triangles, diamonds, and squares, respectively.

**Table 7.** Problem 3: $v_{\mathrm{RMS}}(t = 9.8175 \times 10^{-5})^a$

| Courant Number | ERK4 | ODD | SG1 |
|---|---|---|---|
| 0.99 | 6.85023365134287 | 6.83131605759924 | 6.82193840707678 |
| 0.495 | 6.85023365052789 | 6.84076912701060 | 6.83605707015817 |
| 0.2475 | 6.85023369947449 | 6.84549992958774 | 6.84313811102065 |
| 0.12375 | 6.85023368616665 | 6.84786644620770 | 6.84668407594190 |
| 0.061875 | 6.85023369439422 | 6.84904998094178 | 6.84845843178892 |
| 0.0309375 | 6.85023369873922 | 6.84964181526977 | 6.84934595239669 |
| numerical | 6.85023369873922 | | |
| exact | 6.85041371419587 | | |

[a]Summary of RMS velocities at $t = 9.8175 \times 10^{-5}$ for different Courant numbers using the velocity update schemes of table 1. The numerical reference value for error calculations is indicated by "numerical" in the first column. The corresponding exact value is indicated as "exact". Courant numbers are relative to the ERK4 scheme.

we are unable to examine convergence rates for larger time steps. The time step size is significantly more restricted in this problem due to the higher spatial resolution required. Nonetheless, we observe that the ERK4 scheme provides a more accurate solution at its maximum stable time step size compared to the ODD and SG1 schemes at their smallest time step sizes.

We note that it is expected that convergence rates relative to the exact value are less than those relative to the numerical reference values. As $\Delta t \to 0$, the numerical models converge to the exact solution of the method-of-lines ODEs given in equation 13, which deviates from the solution of equations 10–12 due to spatial discretization. Accordingly, the numerical models converge more precisely to the numerical reference value, which contains the effect of spatial discretization.

#### 3.3.3 Accuracy and computational expense

Figure 8 shows logarithmic plots of RMS velocity error relative to the exact value versus a) the number of velocity updates and b) the number of time integration stages for the ERK4, ODD, and SG1 schemes. For an equivalent number of velocity updates, the SG1 scheme has a slight advantage in RMS velocity error over the ODD scheme, while the ERK4 scheme has the smallest error magnitudes. Also, the least expensive model possible in terms of velocity updates is that with the SG1 scheme (although the error is the largest).

For an equal number of time integration stages (proportional to particle advection cost), the ODD scheme has a slight advantage over the SG1 scheme. As before, the ERK4 scheme results in the lowest errors for any number of stages. We note that there is no significant advantage of decreasing the time step size (i.e., increasing the number of velocity updates or stages) for the ERK4 scheme because all stable time step sizes fall within a stagnant regime.

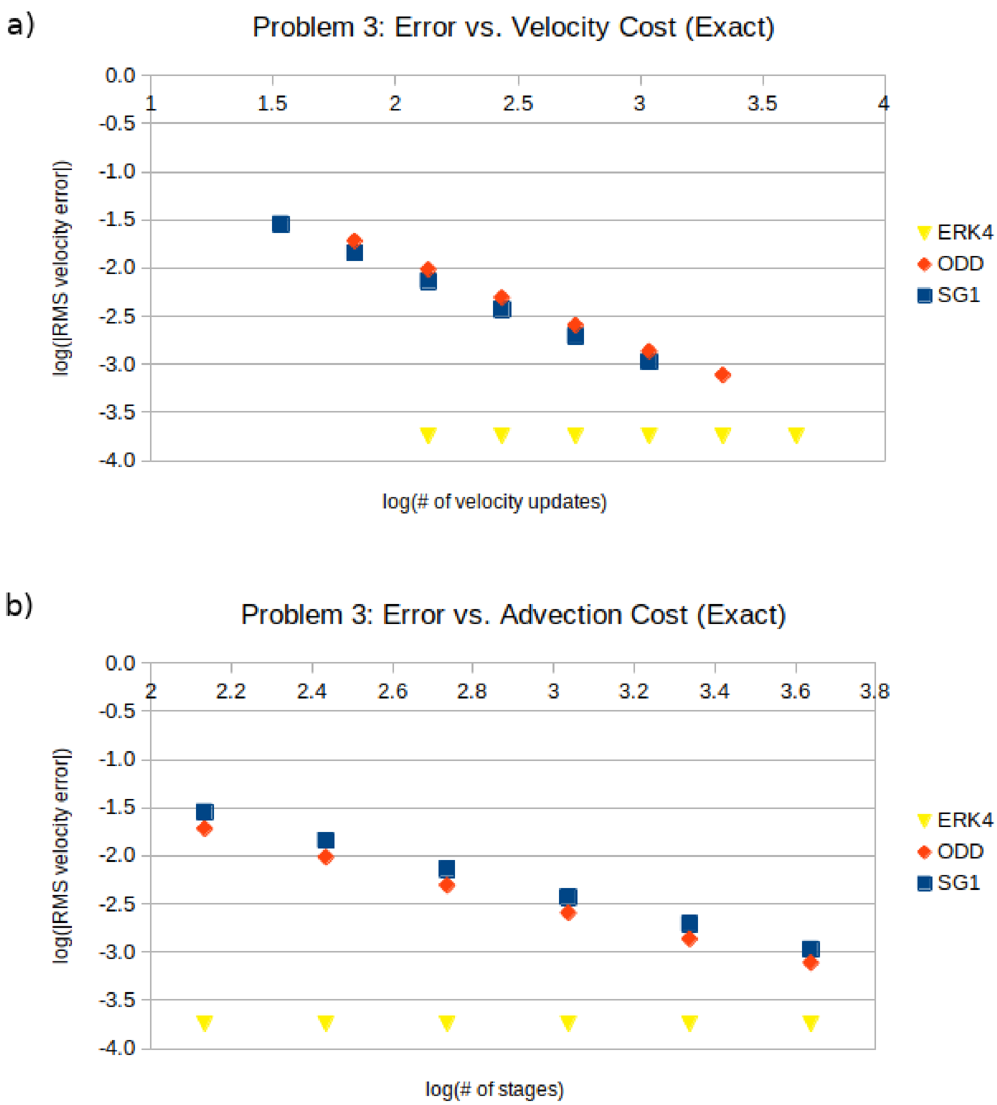



**Figure 8.** Base-10 logarithmic plots of RMS velocity error magnitude versus a) the total number of velocity updates and b) the total number of time integration stages for problem 3. The RMS velocity error is the difference between model and exact $v_{\mathrm{RMS}}$ values for $t = 9.8175 \times 10^{-5}$. The ERK4, ODD, and SG1 schemes are shown using triangles, diamonds, and squares, respectively.

Figure 9 is similar to figure 8 except that the RMS velocity error is computed relative to the numerical reference value. The behaviors of the ODD and SG1 schemes do not change significantly from that of figure 8 because the RMS velocity values are sufficiently far from both the exact and numerical reference values. Accordingly, trends for these schemes observed in figure 8 also hold for figure 9. However, the behavior of the ERK4 scheme relative to the numerical reference value differs from that relative to the exact value. This is because spatial discretization error is included in model results and the numerical reference value, allowing for more precise agreement.

As in figure 8, we observe that the ERK4 scheme in figure 9 produces significantly smaller errors compared to the ODD and SG1 schemes for equal velocity cost or equal advection cost. However, the error magnitudes for the ERK4 scheme in figure 9 are at least three orders of magnitude lower than in figure 8. We also note that the ERK4 scheme has a greater minimum cost for velocity computation than the ODD and SG1 schemes. Additionally, we observe that there is a slight advantage in ERK4 scheme error for a larger number of velocity updates or stages.

## 3.4 Problem 4: thermochemical convection

### *3.4.1 The exact solution*

We employ the exact solution described in S. J. Trim et al. (2023). The derivation of the exact solution for the thermochemical case follows the same general strategy as that used for problem 3. However, the addition of equation 4 must be taken into account and adds considerable difficulty. Similar to problem 3, a stream function is first presumed for the flow. Next, a solution for composition is found using the method of characteristics. Subsequently, equation 8 is used to solve for temperature. Lastly, equation 3 is used to solve for the internal heating rate. Due to a large number of terms, computer algebra systems were used for the symbolic computation of $H$, which is available via GitHub (`https://github.com/seantrim/exact-thermochem-solution`) and Zenodo (S. Trim, 2023a). Note that using the routines provided, $H$ can be calculated for $x \in (-\alpha/2, 3\alpha/2)$ and $z \in (-1, 2)$. This allows $H$ to be calculated on the domain boundaries and at ghost point positions if necessary.

Construction of the exact solution may be summarized as follows. The presumed stream function is given by

$$\psi(x,z,t) = \sin(\pi x/\alpha)\sin(\pi z) f(t), \tag{77}$$

where $f(t)$ describes the time dependence. Applying equations 9 and 70, we have

$$v_{\mathrm{RMS}}(t) = \frac{\pi\sqrt{\alpha^2+1}}{2\,\alpha}|f(t)|. \tag{78}$$

The initial composition field is given by $C(x,z,t=0) = C_0(z)$ (see equation 66). Applying the method of characteristics, the time-dependent composition field can be expressed as a transformation of its initial condition, giving

$$C(x,z,t) = C_0(z_0) = \left[1+\exp\left[-2k\left(z_I - z_0\right)\right]\right]^{-1}, \tag{79}$$

where $z_0$ is the initial vertical position of the fluid parcel located at $(x,z)$ at time $t$. It can be shown that

$$z_0 = \begin{cases} z, & \text{if } z=\{0,1\} \text{ or } (x,z)=(\alpha/2,1/2), \\ \frac{1}{\pi}\operatorname{arccot}\left(Z_0\right), & \text{if } x=\{0,\alpha\} \text{ and } Z_0 \geq 0, \\ 1+\frac{1}{\pi}\operatorname{arccot}\left(Z_0\right), & \text{if } x=\{0,\alpha\} \text{ and } Z_0 < 0, \\ \frac{1}{\pi}\arccos\left[\operatorname{cn}\left\{F\left(\pi z\middle|\frac{1}{D^2}\right) - S(x)\frac{\mathrm{i}\pi^2 D}{\alpha}\int_0^t f(t')dt'\middle|\frac{1}{D^2}\right\}\right], & \text{otherwise,} \end{cases} \tag{80}$$

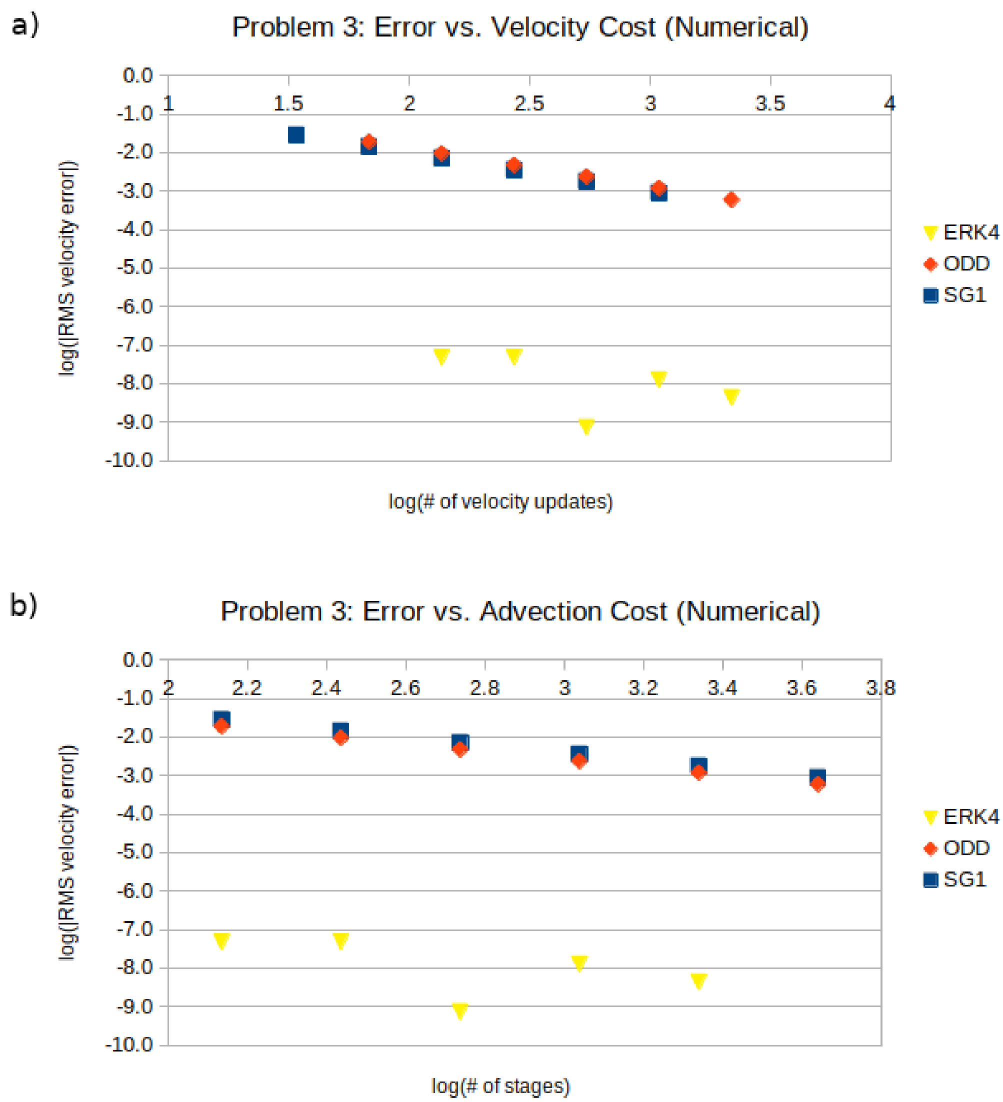



**Figure 9.** Base-10 logarithmic plots of RMS velocity error magnitude versus a) the total number of velocity updates and b) the total number of time integration stages for problem 3. The RMS velocity error is the difference between model and numerical reference $v_{\mathrm{RMS}}$ values at $t = 9.8175 \times 10^{-5}$. The ERK4, ODD, and SG1 schemes are shown using triangles, diamonds, and squares, respectively.

where

$$Z_0 = -\frac{1}{2}\left[e^{-Q} - e^{Q}\right] \tag{81}$$

with

$$Q(z,t) = \ln|\csc(\pi z) + \cot(\pi z)| - \frac{\pi}{\alpha}S(x_b)\int_0^t f(t')dt', \tag{82}$$

$$S(x) = \begin{cases} +1 & \text{if } x \leq \alpha/2 \\ -1 & \text{if } x > \alpha/2 \end{cases}, \tag{83}$$

$$D = |\sin(\pi x/\alpha)\sin(\pi z)|, \tag{84}$$

and $\mathfrak{i}$ is the imaginary unit. We note the use of the Jacobi elliptic function cn and the incomplete elliptic integral of the first kind $F$ in equation 80. Additionally, the temperature is given by

$$T(x,z,t) = \frac{1}{Ra_T}\left[-\frac{\pi^3\left(\alpha^2+1\right)^2}{\alpha^3}\cos\left(\pi x/\alpha\right)\sin\left(\pi z\right)f\left(t\right) + Ra_C C + (Ra_T - Ra_C)(1-z)\right]. \tag{85}$$

We take $f(t) = 100\sin(100\pi t)$, which results in the same stream function used in problem 3. Parameters are given in table 3, with $H$ values computed using the routines provided in the GitHub and Zenodo repositories described above.

For this problem, we use a 600×600 Eulerian mesh with 240 TPCV. Due to sharp changes near the boundaries in the exact solution, we placed approximately half of the tracers in boundary adjacent cells directly on the boundaries. This allowed boundary behavior to be more effectively resolved. We investigate evolution until $t = 9.961875\times10^{-5}$, which is a similar time interval considered for problem 3.

#### *3.4.2 Convergence Rates*

Table 8 shows the RMS velocity values for the ERK4, ODD, and SG1 velocity update schemes for Courant numbers ranging between 0.061875 and 0.99. Numerical reference and exact values are also shown. For the ERK4 scheme, it was found that the RMS velocity converged to 9 significant figures over the three smallest Courant numbers. Accordingly, we take the 0.061875 Courant number value using the ERK4 scheme as the numerical reference value. Conversely, the ODD and SG1 schemes converged to within four digits of the numerical reference value. The ERK4, ODD, and SG1 schemes match the exact value to within three significant figures. There is a larger discrepancy relative to the exact value due to spatial discretization error, which is not alleviated by decreasing the time step size. However, we note that the RMS velocity value of the ERK4 scheme is closer to the exact value compared to the other schemes for all Courant numbers considered.

Figure 10 shows logarithmic plots of RMS velocity error relative to a) exact and b) numerical reference values versus time step size. Convergence rates relative to the exact value are relatively low due to spatial discretization error. Specifically, the observed orders of convergence for the ERK4, ODD, and SG1 schemes are 0.0067, 0.2818, and 0.3608, respectively. Relative to the exact value, the ERK4 scheme does not significantly benefit from decreasing the time step size below the stability limit because spatial discretization errors are larger than temporal discretization errors. However, the ERK4 scheme at the largest stable time step size produces a smaller error than the other schemes at the smallest time step sizes considered. In contrast, the ODD and SG1 schemes do benefit from decreased time step sizes because the temporal discretization error is significant for larger Courant numbers.

Relative to the numerical reference value, the ODD and SG1 schemes have observed orders of accuracy of 0.99987 and 0.99979, respectively. For the ERK4 scheme, the observed order of accuracy relative to the numerical reference value is approximately 4.8.

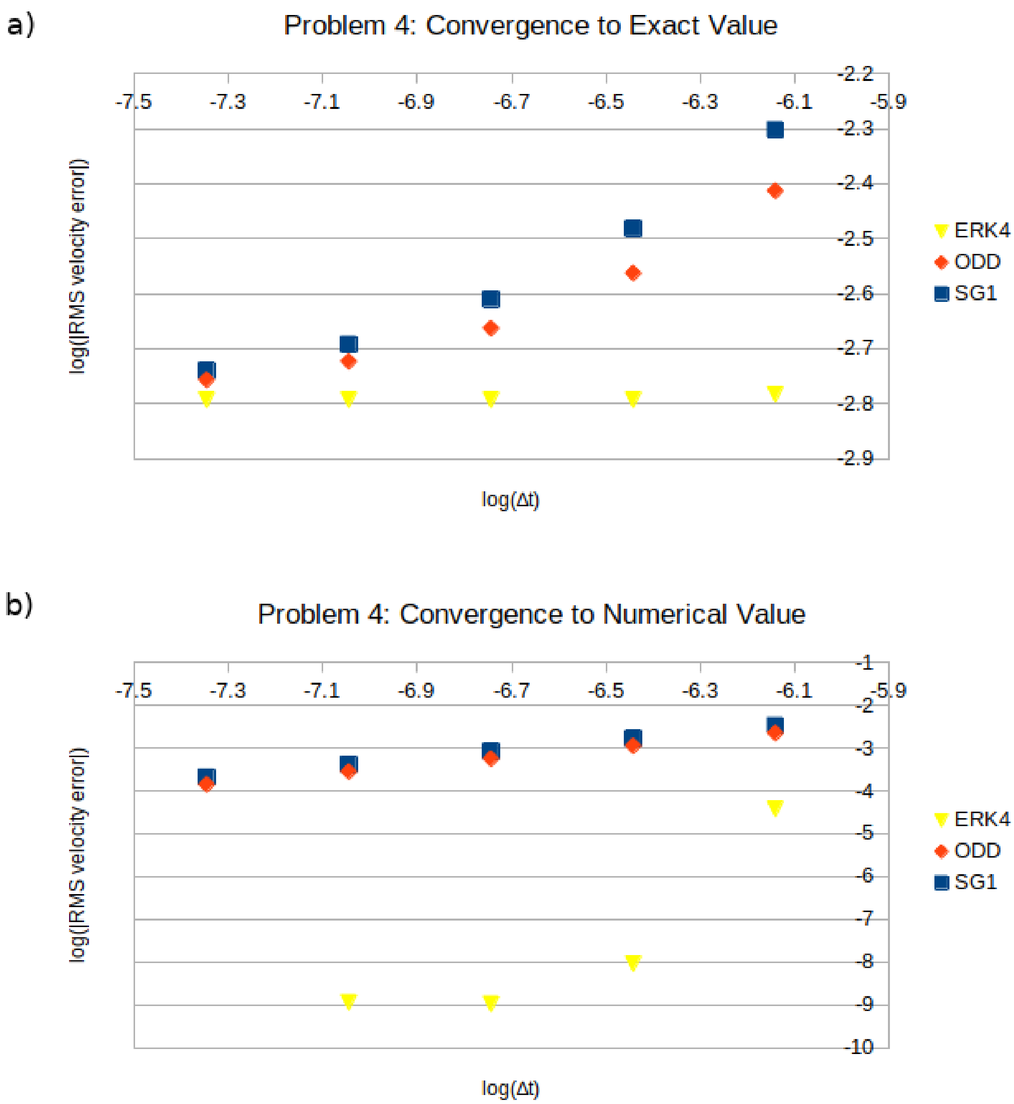



**Figure 10.** Base-10 logarithmic plot showing RMS velocity error magnitude versus the time step size for problem 4. The RMS velocity error is the difference between model $v_{\mathrm{RMS}}$ values and either a) exact or b) numerical reference values for $t = 9.961875 \times 10^{-5}$. The ERK4, ODD, and SG1 schemes are shown using triangles, diamonds, and squares, respectively.

**Table 8.** Problem 4: $v_{\mathrm{RMS}}(t = 9.961875 \times 10^{-5})^a$

| Courant Number | ERK4 | ODD | SG1 |
|---|---|---|---|
| 0.99 | 6.94947596496731 | 6.94725994063769 | 6.94613422117755 |
| 0.495 | 6.94951362394697 | 6.94838669871076 | 6.94782350918055 |
| 0.2475 | 6.94951363193589 | 6.94895012736312 | 6.94866844863008 |
| 0.12375 | 6.94951363184592 | 6.94923185260553 | 6.94909099944699 |
| 0.061875 | 6.94951363296964 | 6.94937272486854 | 6.94930229335591 |
| numerical | 6.94951363296964 | | |
| exact | 6.95112243830231 | | |

[a]Summary of RMS velocities at $t = 9.961875 \times 10^{-5}$ for different Courant numbers using the velocity update schemes of table 1. The numerical reference value for error calculations is indicated by "numerical" in the first column. The corresponding exact value is indicated as "exact". Courant numbers are relative to the ERK4 scheme.

However, the rate of convergence slows as the time step size is reduced. These findings are consistent with those of section 2.4. Compared to both reference values, the ODD scheme is more accurate than SG1 for a given Courant number (although the ODD scheme involves twice as many velocity updates).

#### *3.4.3 Accuracy and computational expense*

Figure 11 shows logarithmic plots of the RMS velocity error relative to the exact value versus a) the total number of velocity updates and b) the total number of time integration stages for all velocity update schemes. Relative to the exact value, there is little improvement for the ERK4 scheme as the total number of velocity updates or stages increase. However, the ERK4 scheme results in the smallest error for any number of velocity solves or stages. Accordingly, the ERK4 scheme is the most efficient scheme in terms of both velocity and advection costs for small error tolerances.

In contrast, the ODD and SG1 schemes do show reduced errors as the total number of velocity updates or stages increases. We observe that the SG1 scheme produces smaller errors than the ODD scheme for a fixed number of velocity updates. Conversely, the ODD scheme gives smaller errors than the SG1 scheme for a fixed number of stages.

Figure 12 is similar to figure 11 except that the RMS velocity error is computed relative to the numerical reference value. The main difference compared to figure 11 is the behavior of the ERK4 scheme, which more closely approaches the numerical reference value as the velocity or advection costs increase. Accordingly, low temporal discretization error tolerances can be met using the ERK4 scheme compared to the other schemes for equivalent velocity or advection costs.

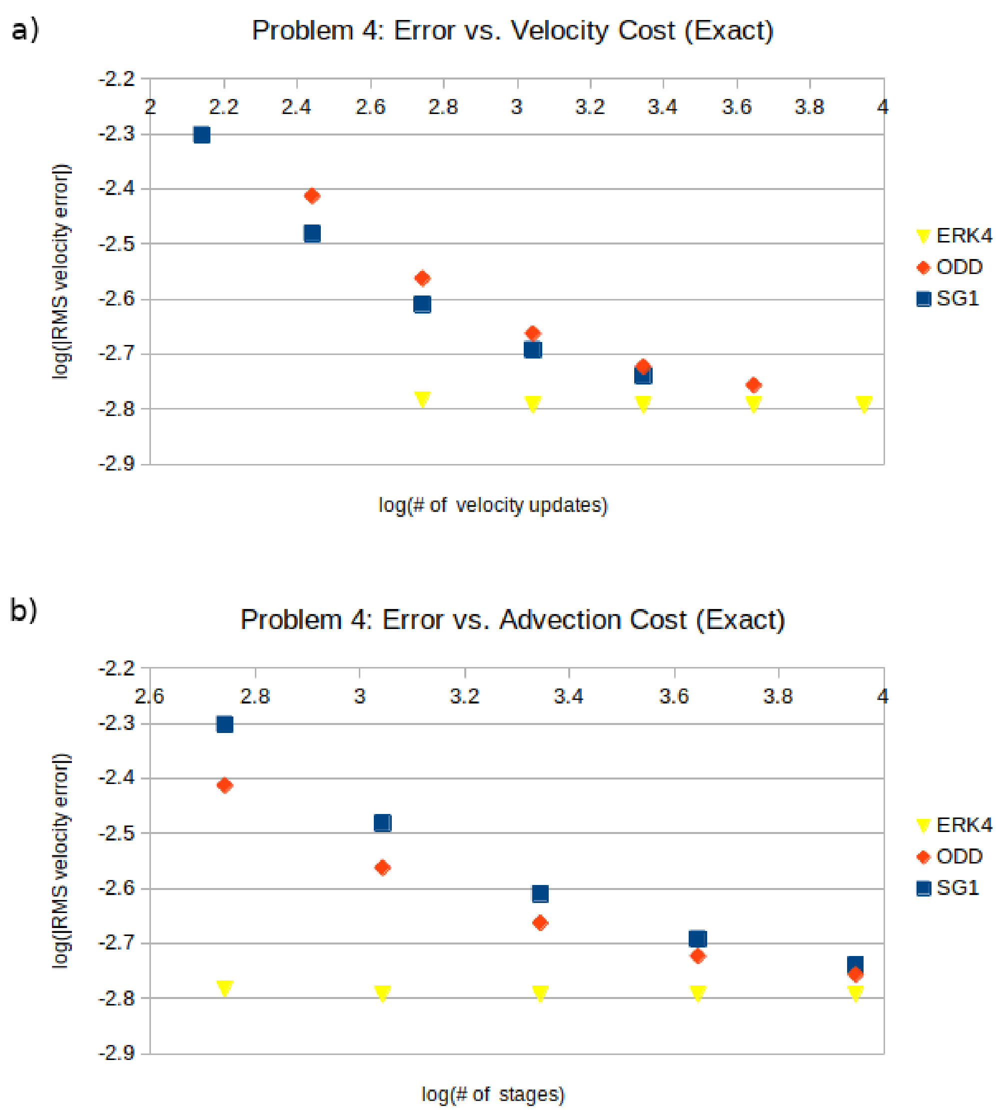
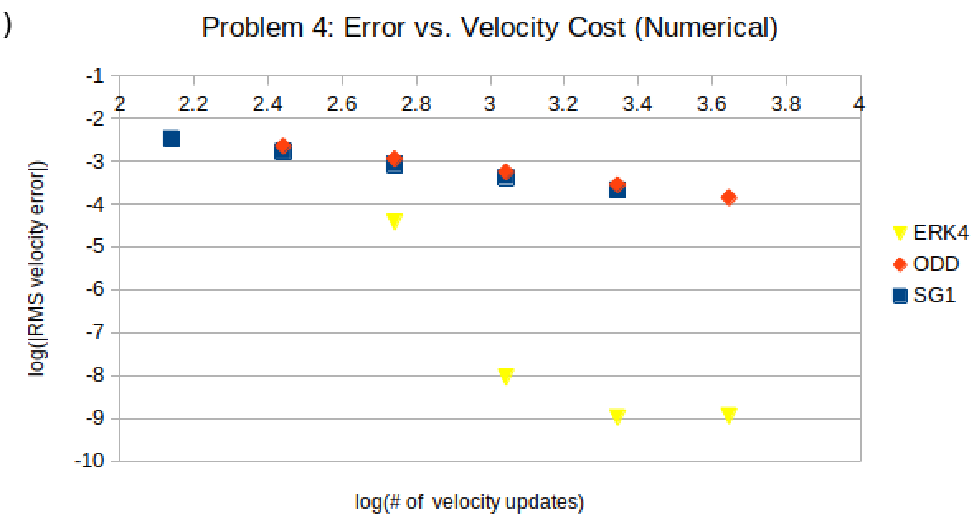



**Figure 11.** Base-10 logarithmic plots of RMS velocity error magnitude versus a) the total number of velocity updates and b) the total number of time integration stages for problem 4. The RMS velocity error is the difference between model and exact $v_{\mathrm{RMS}}$ values for $t = 9.961875 \times 10^{-5}$. The ERK4, ODD, and SG1 schemes are shown using triangles, diamonds, and squares, respectively.

a)




b)

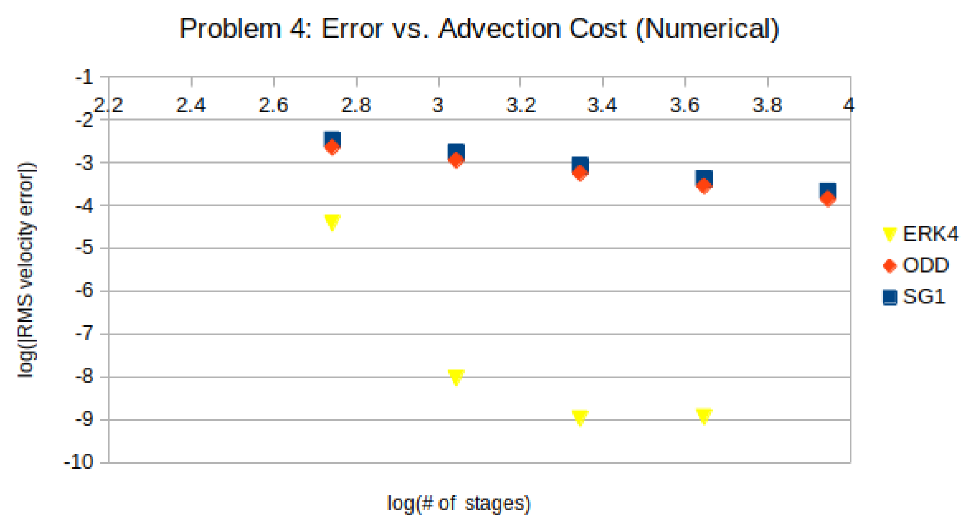



**Figure 12.** Base-10 logarithmic plots of RMS velocity error magnitude versus a) the total number of velocity updates and b) the total number of time integration stages for problem 4. The RMS velocity error is the difference between model and numerical reference $v_{\mathrm{RMS}}$ values for $t = 9.961875 \times 10^{-5}$. The ERK4, ODD, and SG1 schemes are shown using triangles, diamonds, and squares, respectively.

# 4 Discussion

## 4.1 Observed computational cost

Figure 13a shows the particle advection and velocity compute times as a fraction of the total run time for each test problem using the ERK4 scheme. Specifically, the timing for particle advection includes the floating-point operations in equations 17–20 but does not include evaluations of $\nabla^2 T + H$ and $\mathbf{v}$ on the Eulerian grid and the interpolation of $T$ and $C$ from particle positions to the Eulerian grid between stages. The velocity compute times include the evaluation of $\mathbf{v}$ on the Eulerian mesh. This evaluation corresponds to the numerical solution of equation 8 via LU factorization and subsequent solutions of the resulting triangular systems (i.e., forward and back substitution). The ratio of the particle advection to the velocity compute time is shown in figure 13b.

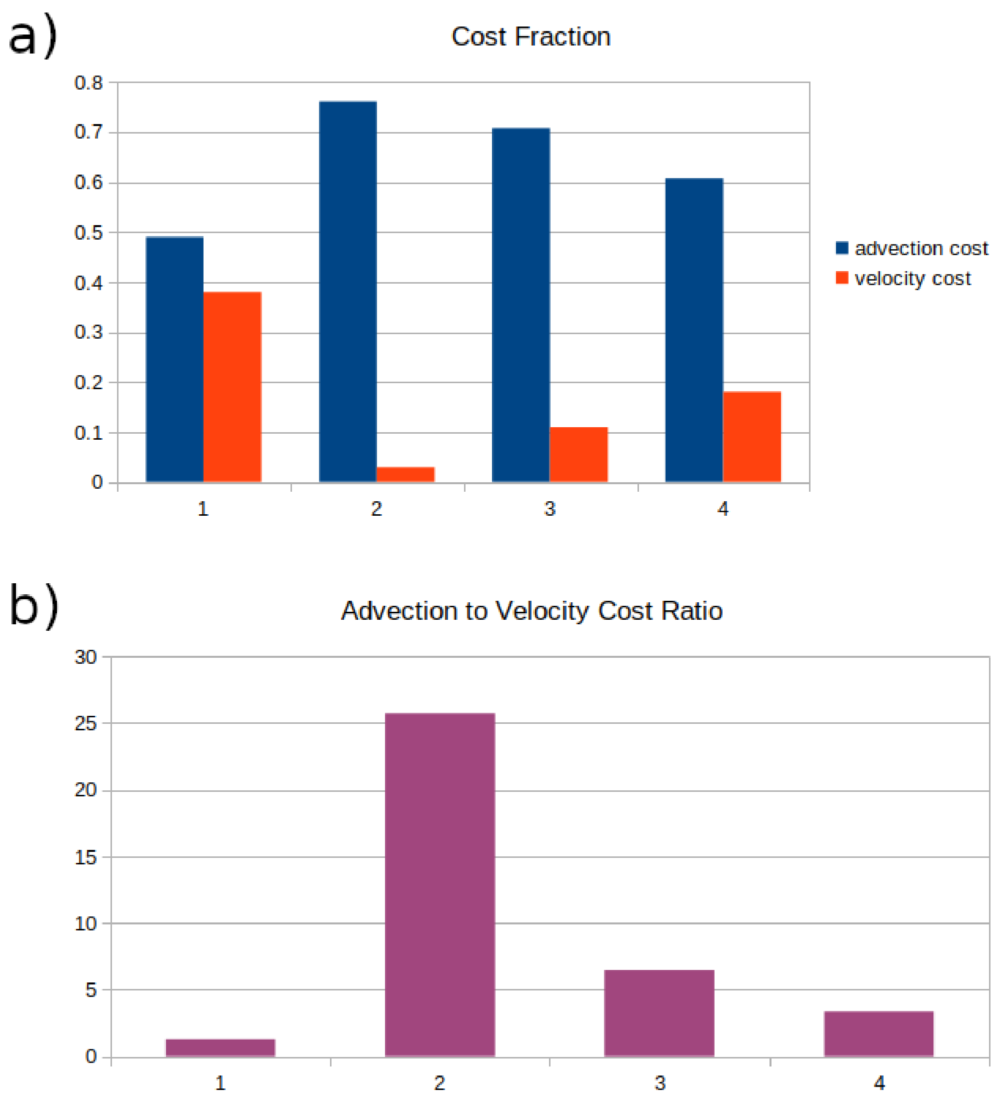



**Figure 13.** Plots of a) particle advection and velocity compute times as a fraction of the total compute time, and b) the ratio of advection to velocity compute times for all test problems using the ERK4 scheme. The test problem number is given on the horizontal axes.

For problem 1, we observe that the particle advection cost is slightly greater than that of velocity. For problems 2–4, we observe that particle advection cost dominates that of velocity. Specific behavior depends on both the Eulerian grid resolution and tracer particle density. For instance, the calculations for problem 1 required the lowest particle density of the test problems, reducing the advection cost. Problem 2 required a greater particle density at a lower Eulerian resolution. These requirements resulted in an increase in advection cost and a reduction in velocity cost, giving rise to a large ratio in figure 13b. Problem 3 featured double the particle density and one-third of the Eulerian mesh spacing of problem 2. These settings increased the advection cost in absolute terms but reduced it relative to the total computation time (due to a higher velocity cost). Problem 4 featured the same particle density as problem 2 but with half the Eulerian mesh spacing. The high Eulerian resolution led to a significant increase in the velocity computation cost, reducing the ratio observed in figure 13.

Due to the relatively high advection cost in these conditions, the ODD scheme produces a given error tolerance a bit more efficiently than the SG1 scheme (see section 4.3). However, the ERK4 scheme is the most efficient for achieving low error tolerances. It is important to note that the cost of particle advection relative to velocity computation depends on the numerical method used. Additionally, the cost of velocity computations increases significantly when considering variable viscosity flows.

### 4.2 Suboptimal convergence of ERK4 scheme for $Ra_T > 0$

For $Ra_T > 0$, the time step size is stability limited due to the presence of thermal diffusion. The theoretical temporal discretization error for the ERK4 scheme is $E_t \sim \mathcal{O}((\Delta t)^4)$. However, for a uniform grid, numerical stability requires that $\Delta t \leq A(\Delta x)^2$ where $A$ depends on the order of accuracy of the finite differences used. Accordingly, we have $E_t \lesssim \mathcal{O}((\Delta x)^8)$. Thus, our spatial discretization error must satisfy $E_s \lesssim \mathcal{O}((\Delta x)^8)$ if we wish to see significant improvement in simulation accuracy relative to the exact solution as we decrease $\Delta t$ below the stability limit.

We have the ability to increase the order of accuracy of the finite difference used to approximate spatial derivatives and the interpolants used to approximate Lagrangian values from neighboring Eulerian values. However, the interpolation from Lagrangian to Eulerian values is governed by bilinear shape functions, which we believe limits the convergence rate of the spatial discretization (Gassmöller et al., 2019). In future studies, high-order shape functions should be explored to see whether the theoretical convergence rate of the ERK4 scheme can be fully achieved.

If higher-order interpolants are not used for $Ra_T > 0$, ERK4 may not be as computationally efficient as ERK3 or ERK2. For thermochemical calculations with constant cell shape functions, ERK2 was found to be more efficient than ERK4 (S. Trim et al., 2020).

### 4.3 Pros and cons of the different schemes

For a given level of accuracy, the ERK4 scheme is the most efficient method in terms of both velocity solver and particle advection costs. The only limiting factor to the ERK4 scheme is that its minimum cost is greater than the other schemes. For extremely large problems, this may make the ODD or SG1 schemes more attractive. Additionally, the choice of velocity update scheme may depend on the error level required. If high accuracy is required, the ERK4 scheme is the winning choice. However, if it is acceptable to have larger errors, the ODD or SG1 schemes may achieve them more efficiently than the ERK4 scheme.

The ODD and SG1 schemes show similar performance for a given computational cost. However, for a fixed velocity solver cost, the SG1 scheme produces smaller errors than the ODD scheme. This observation may be useful if the velocity solver is the bottleneck. Conversely, for a given particle advection cost, the ODD scheme gives smaller errors than the SG1 scheme. This observation may be useful if particle advection is the bottleneck. The velocity computations may be a bottleneck for large spatial meshes (particularly for direct solvers) or if viscosity variations are present. Tracer advection tends to be the bottleneck for isoviscous calculations at modest spatial resolutions.

Unlike the ERK4 and SG1 schemes, the ODD scheme does not give a distinct velocity field for every time step. For larger time steps, this can cause oscillations in the $v_{\mathrm{RMS}}$ value. However, for situations where particle advection is the bottleneck, the ODD scheme gives slightly smaller errors than the SG1 scheme.

For steady-state flows, time accuracy is not necessarily required (Pulliam & Zingg, 2014). In that case, using the ODD or SG1 schemes may reduce the computational cost involved in achieving a steady state.

Additionally, the stability regions of the ODD and SG1 schemes differ from that of the ERK4 scheme for $\gamma \neq 0$ (see figure 1). For $\gamma$ near zero, the ODD and SG1 schemes have a similar stability region to the ERK4 scheme and at reduced computational cost. For $\gamma$ near 0.5, ODD and SG1 have superior stability along the real axis compared to the ERK4 scheme. In this case, the SG1 scheme is the most stable along the real axis by a large margin. However, the stability regions of the ODD and SG1 schemes do not cover the imaginary axis for $\gamma \neq 0$, unlike the ERK4 scheme. This is important for Eulerian solvers for $T$ and $C$, which generally require greater stability along the imaginary axis due to the spatial discretization of advection terms.

# 5 Conclusions

## 5.1 Summary

1. Computing the velocity field is computationally expensive in mantle convection models.
2. To reduce expense, velocity fields can be computed for a subset of high-order Runge–Kutta stages, potentially resulting in a useful time integration scheme for steady-state problems or if larger error tolerances are acceptable.
3. Theoretical stability regions and orders of accuracy were developed for the time integration schemes considered (ERK4, ODD, and SG1).
4. In agreement with theory, computing velocities for a subset of time integration stages was observed to:
   (a) not adversely impact numerical stability for the problems considered
   (b) result in at most first-order time accuracy
5. Computing velocities for every time integration stage resulted in:
   (a) substantially improved efficiency for small error tolerances
   (b) high-order convergence for isothermal and thermochemical flows
6. For isocompositional flows, the theoretical high-order convergence of Runge–Kutta integration with velocities computed at every stage was difficult to observe due to the stability limit (see section 4.2)

## 5.2 Future work

It would be of interest to explore other velocity update schemes for particle advection such as that used for the second-order Runge–Kutta method. We have focused on isoviscous models in 2D. However, the efficiency gains of the ERK4 scheme for time-accurate calculations are likely to be even greater for variable viscosity and 3D flows due to the increased computational cost of solving for the velocity field. Direct testing of this hypothesis is a worthy direction for future effort.

## 6 Open Research

Source code and related files for ProjecTracer are available on GitHub (`https://github.com/seantrim/ProjecTracer`) and Zenodo (S. Trim, 2023b). The routines used to calculate $H$ for problem 4 are also available on GitHub (`https://github.com/seantrim/exact-thermochem-solution`) and Zenodo (S. Trim, 2023a).

Figures were created using LibreOffice (The Document Foundation, 2022), GIMP (GIMP: GNU Image Manipulation Program, 2021), and Gnuplot (Williams & Kelley, 2021). LibreOffice (`https://www.libreoffice.org/`), GIMP (`https://www.gimp.org`) and Gnuplot (`http://www.gnuplot.info/`) are available via free licenses. Figures 1-2 utilize the perceptually uniform batlow color scale (Crameri et al., 2020) available on Zenodo (Crameri, 2021).

**Acknowledgments**

S.J.T and R.J.S are grateful for funding from the Natural Sciences and Engineering Research Council of Canada under Discovery Grant [RGPN-2020-04467]. S.L.B is also grateful for funding from the Natural Sciences and Engineering Research Council of Canada [RGPN-2020-06332].

## References

Ascher, U. M., Ruuth, S. J., & Spiteri, R. J. (1997). Implicit-explicit Runge–Kutta Methods for Time-dependent Partial Differential Equations. *Applied Numerical Mathematics*, *25*(2-3), 151–167.

Bangerth, W., Dannberg, J., Gassmoeller, R., & et al., T. H. (2020, 6). ASPECT: Advanced Solver for Problems in Earth's ConvecTion, User Manual [Computer software manual]. Retrieved from `https://figshare.com/articles/journal_contribution/ASPECT_Advanced_Solver_for_Problems_in_Earth_s_ConvecTion_User_Manual/4865333` doi: 10.6084/m9.figshare.4865333.v7

Batchelor, G. (1967). An introduction to fluid dynamics. 1967. *Cambridge,: UP xviii*, *615*.

Brandenburg, J., & Van Keken, P. (2007). Deep storage of oceanic crust in a vigorously convecting mantle. *Journal of Geophysical Research: Solid Earth*, *112*(B6).

Butcher, J. C. (2016). *Numerical methods for ordinary differential equations.* John Wiley & Sons.

Crameri, F. (2021, September). *Scientific colour maps.* Zenodo. Retrieved from `https://doi.org/10.5281/zenodo.5501399` (The development of the Scientific colour maps is not funded any longer, but will continue as a pro bono project for the scientific community. - Fabio) doi: 10.5281/zenodo.5501399

Crameri, F., Schmeling, H., Golabek, G., Duretz, T., Orendt, R., Buiter, S., . . . Tackley, P. (2012). A comparison of numerical surface topography calculations in geodynamic modelling: an evaluation of the 'sticky air'method. *Geophysical Journal International*, *189*(1), 38–54.

Crameri, F., Shephard, G. E., & Heron, P. J. (2020). The misuse of colour in science communication. *Nature communications*, *11*(1), 5444.

Fornberg, B. (1988). Generation of finite difference formulas on arbitrarily spaced grids. *Mathematics of computation*, *51*(184), 699–706.

Gable, C. W., O'Connell, R. J., & Travis, B. J. (1991). Convection in three dimensions with surface plates: Generation of toroidal flow. *Journal of Geophysical Research: Solid Earth*, *96*(B5), 8391–8405.

Gassmöller, R., Lokavarapu, H., Bangerth, W., & Puckett, E. G. (2019). Evaluating the accuracy of hybrid finite element/particle-in-cell methods for modelling incompressible stokes flow. *Geophysical Journal International*, *219*(3), 1915–1938.

Gassmöller, R., Lokavarapu, H., Heien, E., Puckett, E. G., & Bangerth, W. (2018). Flexible and scalable particle-in-cell methods with adaptive mesh refinement for geodynamic computations. *Geochemistry, Geophysics, Geosystems*, *19*(9), 3596-3604. Retrieved from `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2018GC007508` doi: https://doi.org/10.1029/2018GC007508

Gerya, T. V., & Yuen, D. A. (2003). Characteristics-based marker-in-cell method with conservative finite-differences schemes for modeling geological flows with strongly variable transport properties. *Physics of the Earth and Planetary Interiors*, *140*(4), 293–318.

GIMP: GNU Image Manipulation Program. (2021). *version 2.10.30.* `https://www.gimp.org`.

Kennedy, C. A., & Carpenter, M. H. (2003). Additive runge–kutta schemes for convection–diffusion–reaction equations. *Applied numerical mathematics*, *44*(1-2), 139–181.

Larsen, T. B., Yuen, D. A., Moser, J., & Fornberg, B. (1997). A high-order finite-difference method applied to large rayleigh number mantle convection. *Geophysical & Astrophysical Fluid Dynamics*, *84*(1-2), 53–83.

Lenardic, A., Moresi, L.-N., & Mühlhaus, H. (2003). Longevity and stability of cratonic lithosphere: insights from numerical simulations of coupled mantle convection and continental tectonics. *Journal of Geophysical Research: Solid Earth*, *108*(B6).

Li, M., & McNamara, A. K. (2013). The difficulty for subducted oceanic crust to accumulate at the earth's core-mantle boundary. *Journal of Geophysical Research: Solid Earth*, *118*(4), 1807–1816.

Li, M., McNamara, A. K., Garnero, E. J., & Yu, S. (2017). Compositionally-distinct ultra-low velocity zones on earth's core-mantle boundary. *Nature communications*, *8*(1), 1–9.

Lomax, H., Pulliam, T. H., & Zingg, D. W. (2001). *Fundamentals of computational fluid dynamics* (Vol. 246). Springer.

McNamara, A. K., & Zhong, S. (2004). Thermochemical structures within a spherical mantle: Superplumes or piles? *Journal of Geophysical Research: Solid Earth*, *109*(B7). Retrieved from `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2003JB002847` doi: https://doi.org/10.1029/2003JB002847

McNamara, A. K., & Zhong, S. (2005). Thermochemical structures beneath Africa and the Pacific Ocean. *Nature*, *437*(7062), 1136–1139.

Moresi, L., Dufour, F., & Mühlhaus, H.-B. (2002). Mantle convection modeling with viscoelastic/brittle lithosphere: Numerical methodology and plate tectonic modeling. *Pure and applied Geophysics*, *159*, 2335–2356.

Moresi, L., Dufour, F., & Mühlhaus, H.-B. (2003). A lagrangian integration point finite element method for large deformation modeling of viscoelastic geomaterials. *Journal of computational physics*, *184*(2), 476–497.

Puckett, E. G., Turcotte, D. L., He, Y., Lokavarapu, H., Robey, J. M., & Kellogg, L. H. (2018). New numerical approaches for modeling thermochemical convection in a compositionally stratified fluid. *Physics of the Earth and Planetary Interiors*, *276*, 10–35.

Pulliam, T. H., & Zingg, D. W. (2014). *Fundamental algorithms in computational fluid dynamics* (Vol. 940). Springer.

Ramberg, H. (1981). *Gravity, deformation, and the earth's crust: In theory, experiments, and geological application.* Academic press.

Rolf, T., & Tackley, P. (2011). Focussing of stress by continents in 3d spherical mantle convection with self-consistent plate tectonics. *Geophysical Research Letters*, *38*(18).

Schubert, G., Turcotte, D. L., & Olson, P. (2001). *Mantle convection in the earth and planets.* Cambridge University Press.

Tackley, P. J. (2008). Modelling compressible mantle convection with large viscosity contrasts in a three-dimensional spherical shell using the yin-yang grid. *Physics of the Earth and Planetary Interiors*, *171*(1), 7-18. Retrieved from `https://www.sciencedirect.com/science/article/pii/S0031920108002276` (Recent Advances in Computational Geodynamics: Theory, Numerics and Applications) doi: https://doi.org/10.1016/j.pepi.2008.08.005

Tackley, P. J., & King, S. D. (2003). Testing the tracer ratio method for modeling active compositional fields in mantle convection simulations. *Geochem. Geophys. Geosys.*, *4*(4).

The Document Foundation. (2022). *Libreoffice community.* Retrieved from `https://www.libreoffice.org`

Trim, S. (2017). *The dynamic interaction of mantle compositional heterogeneities at the surface and the core-mantle boundary.* University of Toronto (Canada).

Trim, S. (2023a, February). *seantrim/exact-thermochem-solution: Release 1.2.0.* Zenodo. Retrieved from `https://doi.org/10.5281/zenodo.7671498` doi: 10.5281/zenodo.7671498

Trim, S. (2023b, February). *seantrim/projectracer: Release 1.2.0.* Zenodo. Retrieved from `https://doi.org/10.5281/zenodo.7671666` doi: 10.5281/zenodo.7671666

Trim, S., Heron, P., Stein, C., & Lowman, J. (2014). The feedback between surface mobility and mantle compositional heterogeneity: Implications for the Earth and other terrestrial planets. *Earth and Planetary Science Letters*, *405*, 1–14.

Trim, S., Lowman, J., & Butler, S. (2020). Improving mass conservation with the tracer ratio method: application to thermochemical mantle flows. *Geochemistry, Geophysics, Geosystems*, *21*(2), e2019GC008799.

Trim, S. J., Butler, S. L., McAdam, S. S. C., & Spiteri, R. J. (2023). Manufacturing an exact solution for 2d thermochemical mantle convection models. *Geochemistry, Geophysics, Geosystems*, *24*(4), e2022GC010807. Retrieved from `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2022GC010807` (e2022GC010807 2022GC010807) doi: https://doi.org/10.1029/2022GC010807

Trim, S. J., & Lowman, J. P. (2016). Interaction between the supercontinent cycle and the evolution of intrinsically dense provinces in the deep mantle. *Journal of Geophysical Research: Solid Earth*, *121*(12), 8941–8969.

van Keken, P., King, S., Schmeling, H., Christensen, U., Neumeister, D., & Doin, M.-P. (1997). A comparison of methods for the modeling of thermochemical convection. *Journal of Geophysical Research: Solid Earth*, *102*(B10), 22477–22495.

Wanner, G., & Hairer, E. (1996). *Solving ordinary differential equations ii* (Vol. 375). Springer Berlin Heidelberg New York.

Williams, T., & Kelley, C. (2021, June). *Gnuplot 5.4: an interactive plotting program.* `http://www.gnuplot.info/`.

Zhong, S., & Hager, B. H. (2003). Entrainment of a dense layer by thermal plumes. *Geophysical Journal International*, *154*(3), 666-676. Retrieved from `https://onlinelibrary.wiley.com/doi/abs/10.1046/j.1365-246X.2003.01988.x` doi: https://doi.org/10.1046/j.1365-246X.2003.01988.x