More efficient time integration for Fourier pseudo-spectral DNS of incompressible turbulence

David I. Ketcheson[*][†]   Mikael Mortensen[‡]   Matteo Parsani[†]   Nathanael Schilling[§]

October 8, 2019

Abstract

Time integration of Fourier pseudo-spectral DNS is usually performed using the classical fourth-order accurate Runge–Kutta method, or other methods of second or third order, with a fixed step size. We investigate the use of higher-order Runge–Kutta pairs and automatic step size control based on local error estimation. We find that the fifth-order accurate Runge–Kutta pair of Bogacki & Shampine gives much greater accuracy at a significantly reduced computational cost. Specifically, we demonstrate speedups of 2x-10x for the same accuracy. Numerical tests (including the Taylor–Green vortex, Rayleigh–Taylor instability, and homogeneous isotropic turbulence) confirm the reliability and efficiency of the method. We also show that adaptive time stepping provides a significant computational advantage for some problems (like the development of a Rayleigh–Taylor instability) without compromising accuracy.

1 Time integration of Fourier pseudo-spectral DNS

Direct numerical simulation (DNS) is a key tool in improving our understanding of turbulent flows. Simulation of turbulence in the absence of boundaries is essential for understanding the nature of turbulence itself, and Fourier pseudo-spectral methods are usually the tool of choice thanks to their high computational efficiency, scalability, and accuracy. Because they use basis functions with global support, for a given number of degrees of freedom (DOFs), these methods often provide much greater spatial accuracy than would be possible with typical finite element, finite volume, or finite difference methods.

Pseudo-spectral DNS also requires discretization in time. Explicit time integration is generally preferred because small time steps are required in order to satisfy the accuracy requirements of DNS. Most often[1], the Fourier pseudo-spectral space discretization is coupled with the well-known fourth-order accurate Runge–Kutta time discretization [16] (hereafter RK4) see for example [28, 13].

While RK4 is a remarkably useful general purpose integrator, when combined with a spectral method in space it has the potential to become the main source of discretization error, unless used with a very small time step size. Sometimes lower-order accurate Runge–Kutta or multistep methods are used; they require even smaller step sizes in order to provide time accuracy commensurate with the spectral accuracy obtained in space. For example, among the available open source codes, Tarang [19] includes RK methods of orders one, two, and four; Turbo [25] uses a 3rd-order RK method,

---

[*]Authors listed alphabetically.

[†]King Abdullah University of Science and Technology (KAUST), Extreme Computing Research Center (ECRC), Computer Electrical and Mathematical Science and Engineering Division (CEMSE), Thuwal, 23955-6900, Saudi Arabia

[‡]Department of Mathematics, Universityof Oslo Mathematics and Natural Sciences, Oslo, Norway

[§]Center for Mathematics, Technical University of Munich, 85747 Garching, Germany

[1]Unfortunately, the selection of a time integrator is considered such a matter of course that it is not even specified in many publications.

and Philofluid [12] uses a low-storage fourth-order RK method. The use of small time step sizes is especially significant since such simulations are sometimes run on the largest available supercomputers in order to simulate high Reynolds number flows in a reasonable wall-clock time. Switching to a time integrator that allows larger steps without compromising accuracy or parallel scalability is a simple change that could yield significant benefits.

More than a century has passed since Kutta's development of RK4 [16], and in that time a lot of work has gone into developing highly accurate and efficient Runge–Kutta methods. In this work we explore the application of some of those methods to Fourier pseudo-spectral DNS, focusing particularly on the fifth-order method of Bogacki & Shampine (henceforth BS5) [1]. We make use of the open-source code SpectralDNS [20].

We also investigate the usefulness of automatic step size control based on local error estimation. This is a well-established technique for initial value problems. In the context of Runge–Kutta methods, local error estimation is performed using a pair of methods that share a set of common intermediate stages. The two methods have different orders of accuracy (herein we consider pairs of fifth/fourth order), so that the difference between the numerical solutions they provide serves as an estimate of the error. We refer the reader to [11] for details.

In proposing a change to the time stepping algorithm, care must be taken to ensure that the accuracy obtained is at least as good as what would be obtained with RK4. We show that results from adaptive fifth-order integrators agree with what is obtained with traditional methods to very high accuracy, even when considering pointwise differences of turbulent flow fields. We also show that adaptive time stepping does not negatively impact accuracy, even for problems with emergent instabilities. It is essential to validate these results at reasonably high Reynold's number, which requires a scalable parallel code like SpectralDNS and a reasonably large computing resource. Simulations presented here were run on the Shaheen Cray XC40 system at KAUST [10].

2   Pseudo-spectral discretization of incompressible Navier–Stokes

We consider incompressible fluid flow modeled by the Navier–Stokes equations with a divergence-free velocity field. Defining the modified pressure $P = p + u \cdot u/2$ (where p is the regular pressure), the incompressible Navier–Stokes equations can be written (see e.g. [22])

$$\nabla \cdot u = 0, \tag{1a}$$

$$\frac{\partial u}{\partial t} = u \times \omega - \nabla P + \frac{1}{\mathrm{Re}}\nabla^2 u. \tag{1b}$$

Here u is the velocity, $\omega = \nabla \times u$ is the vorticity, and Re is the Reynolds number.

2.1   Spatial Discretisation

The spatial discretization is based on the traditional Fourier pseudo-spectral method; the description here follows that in [21] and is implemented in the SpectralDNS package [20] in the Python programming language. We consider uniform Cartesian grids in two or three dimensions. Let u denote the approximation to the velocity field on the grid, let $\hat{u}$ denote its discrete Fourier transform (DFT), and let k denote a wavenumber. We can eliminate the pressure from (1b) by taking the divergence of both sides. Then transforming equation (1b) to frequency space gives the following system of ordinary differential equations (ODEs) [21]:

$$\partial_t \hat{u}_k = \widehat{(u \times \omega)}_k - \frac{1}{\mathrm{Re}}|k|^2 \hat{u}_k - ik\frac{k \cdot \widehat{(u \times \omega)}_k}{|k|^2}. \tag{2}$$

Here $|k|^2 = k \cdot k$. As is usual in the pseudo-spectral approach, the cross products in equation (2) are evaluated in physical space and the result is then transformed to frequency space (see e.g. [4]).

2.1.1 Dealiasing

Dealiasing is done using the 3/2-rule as described in [4, p. 134]. For evaluation (in physical space) of the cross products of equation (2), the vectors containing u and ω are padded with zeroes so that there are $\frac{3N}{2}$ frequencies in each direction. Applying the inverse Fourier transform (IDFT) gives variables on a mesh with $\left(\frac{3N}{2}\right)^3$ points in physical space. After the operation has been carried out using these variables, the result is transformed back to frequency space. Then the highest frequencies are truncated to leave N frequencies in each direction.

2.2 Time discretizations

The main purpose of this work is to compare different explicit Runge–Kutta methods for the integration of (2). Specifically, we are interested in improvements that can be achieved by using highly optimized fifth-order pairs. We compare the following temporal discretizations:

- AB2: Second-order Adams-Bashforth. This is used only in the first test, to give an idea of its relative inefficiency compared to higher-order methods.

- RK4 [16]: The classical four-stage fourth-order method. The method is used with a fixed step size. We also use this method to generate reference solutions.

- DP5 [8]: The well-known 5(4) pair of Dormand & Prince.

- KCL5 [15, p. 190] Method RK5(4)8[3R+]M from the work of Kennedy, Carpenter, & Lewis.

- BS5 [1]: The 5(4) pair due to Bogacki & Shampine. We use the error estimator $\hat{b}$ (not b^*).

For the embedded pairs, we compare implementations with fixed step size and with variable step size based on local error control.

2.2.1 Automatic step-size control

We use an automatic step-size control method based on [11, p167], the details of which we describe below for the three-dimensional case; the two-dimensional case is almost exactly the same.

Two different Runge–Kutta methods are used for each step, the second being an "embedded" method for automatic step-size control. Let the numerical solution (in frequency space, with a real DFT in one direction) at timestep $t \in \mathbb{N}$ be denoted by $\hat{u}^{(n)}$ for the main Runge–Kutta method and by $\hat{v}^{(n)}$ for the embedded method. As both $\hat{u}^{(n)}$ and $\hat{v}^{(n)}$ are in frequency space, they each have three components at every point. We write $\pi_j : \mathbb{R}^3 \to \mathbb{R}$ for $j = 1, 2, 3$ as the canonical projections onto the components (i.e. $u^{(n)} = (\pi_1 u^{(n)}, \pi_2 u^{(n)}, \pi_3 u^{(n)})^T$). For each wavenumber k considered, the vector $\widehat{sc}_k^{(n)} = (\pi_1 \widehat{sc}_k^{(n)}, \pi_2 \widehat{sc}_k^{(n)}, \pi_3 \widehat{sc}_k^{(n)})$ is defined to have dimensions like that of $\hat{u}$ and that

$$\pi_j \widehat{sc}_k^{(n)} = TOL_{abs} + \max\left\{|\pi_j \hat{u}_k^{(n-1)}|, |\pi_j \hat{u}_k^{(n)}|\right\} TOL_{rel},$$

where TOL_{abs} and TOL_{rel} are absolute and relative tolerances respectively. For all the plots in this document, we used $TOL_{abs} = TOL_{rel}$ with values in the range 10^{-11} to 10^{-2}. The error for each component j is estimated by

$$err_j^{(n)} = \sqrt{\frac{1}{N^2(N/2+1)} \sum_k \left(\frac{\pi_j u_k^{(n)} - \pi_j v_k^{(n)}}{\pi_j \widehat{sc}_k^{(n)}}\right)^2}.$$

Here the $N^2(N/2+1)$ term gives the total number of frequencies k considered. Note that due to the use of the real DFT in the first direction, this treats the zero and Nyquist-frequencies slightly differently than if a complex DFT had been used in each direction. The final error estimate is then formed by taking the maximum over all components, i.e. $err^{(n)} = \max\{err_1^{(n)}, err_2^{(n)}, err_3^{(n)}\}$

Let $h^{(n)}$ be the stepsize used for the n-th step, with $h^{(0)}$ being supplied at the beginning. We set

$$h_{new} = h^{(n)} \cdot \min\left\{\delta_{max}, \max\left\{\delta_{min}, \delta \cdot (1/err^{(n)})^{1/(q+1)}\right\}\right\},$$

with $\delta_{\max} = 2$ and $\delta_{\min} = 0.01$, $\delta = 0.8$ and q being the order of the embedded method. If $err \leq 1$ then we continue to the next step with $h^{(n+1)} = h_{new}$. Else the current step is rejected and we re-run the step with $h^{(n)} = h_{new}$. We set $\delta_{\max} = 1$ for the first step after a rejected step.

3 Numerical comparisons

We test each of the candidate integrators on variations of three classical problems: the Taylor–Green vortex, the Rayleigh–Taylor instability, and homogeneous isotropic turbulence. We calculate (with RK4 and a very small timestep) a reference solution against which other solutions are compared. As a measure for the error, we use the discrete L_2 and maximum norms of the difference (in velocity or density) between a given solution and the reference solution. This is a very stringent test, given that in each case we are computing a (chaotic) turbulent flow field.

It is natural to wonder if the use of much larger time steps (allowed by higher-order methods and by time step adaptivity) might somehow damp out fine features of the flow. In the worst case, there might be a feedback effect in which numerical dissipation from the use of large step size with an adaptive integrator prevents the development of an instability, which in turn causes the integrator to continue with a large step size. We will see that this problem does not arise in practice. The most important point to take away from the results that follow is that the use of high-order, adaptive methods yields essentially the same solution but requires a much smaller number of FFTs.

3.1 Taylor–Green vortex

The Taylor–Green (TG) vortex test case is a widely-used benchmark, see for instance [6]. We solve the system of equations (1) on the periodic cube $[-\pi \leq x, y, z \leq +\pi]$. The initial velocity components are

$$u^{(0)} = \begin{pmatrix} \sin(x)\cos(y)\cos(z) \\ -\cos(x)\sin(y)\cos(z) \\ 0 \end{pmatrix}.$$

The Reynolds number for this flow is defined as $Re = 1/\nu$, where ν is the dynamic viscosity. Starting from the initial condition, the nonlinear interactions of different flow scales yield vortex breakdowns. This nonlinear process is initially laminar, but it subsequently develops into near anisotropic turbulence that decays with the typical spectral energy distribution. We consider three Reynolds numbers: $Re = 280$, $Re = 800$ and $Re = 1,600$.

To measure accuracy, we look at the maximum (over all time-steps) absolute difference (between a computed solution and the reference solutions reported in literature) in the rate of dissipation of kinetic energy. This maximum is obtained by evaluating a piecewise quadratic interpolation of the rate of dissipation of kinetic energy at each timestep. The interpolation is done using the scipy package [14].
We obtained qualitatively very similar results by considering the L^2-error of the velocity field at the final time. To calculate the rate of dissipation in kinetic energy, we first note that on the cube $[-\pi, \pi]^3$ we have that the total kinetic energy per unit volume is given by $E_{kin} = \frac{1}{2N^6}\|\hat{u}\|$ where $\|.\|_2$ is the L_2 norm, i.e. $\|\hat{u}\|_2^2 = \sum_k \sum_{j=1,2,3,i} |\pi_j \hat{u}_k|^2$ where k ranges over all (discrete) frequencies considered and π_j is the canonical projection onto the j-th component so that $\hat{u}_k = (\pi_1 \hat{u}_k, \pi_2 \hat{u}_k, \pi_3 \hat{u}_k)^T$. The rate of dissipation of kinetic energy is given by

$$\epsilon = -\frac{d}{dt}E_{kin} = \frac{-1}{2N^6}\frac{d}{dt}\sum_k \sum_{j=1,2,3} (\pi_j \hat{u}_k)\overline{(\pi_j \hat{u}_k)} = \frac{-1}{N^6}\sum_k \sum_{j=1,2,3} \text{Re}\{\pi_j(\hat{u}_k f(\hat{u}_k))\},$$

where we use the fact that $\frac{d\hat{u}}{dt} = f(\hat{u})$ with the sum ranging over every frequency k considered. Since we already calculate an approximation of $f(\hat{u})$ when applying the time-integration method, this gives a cheap way of calculating the rate of dissipation of kinetic energy in frequency space. Note that in the implementation we used, the method described above had to be adjusted to treat the 0 and Nyquist frequencies for the first real DFT correctly.
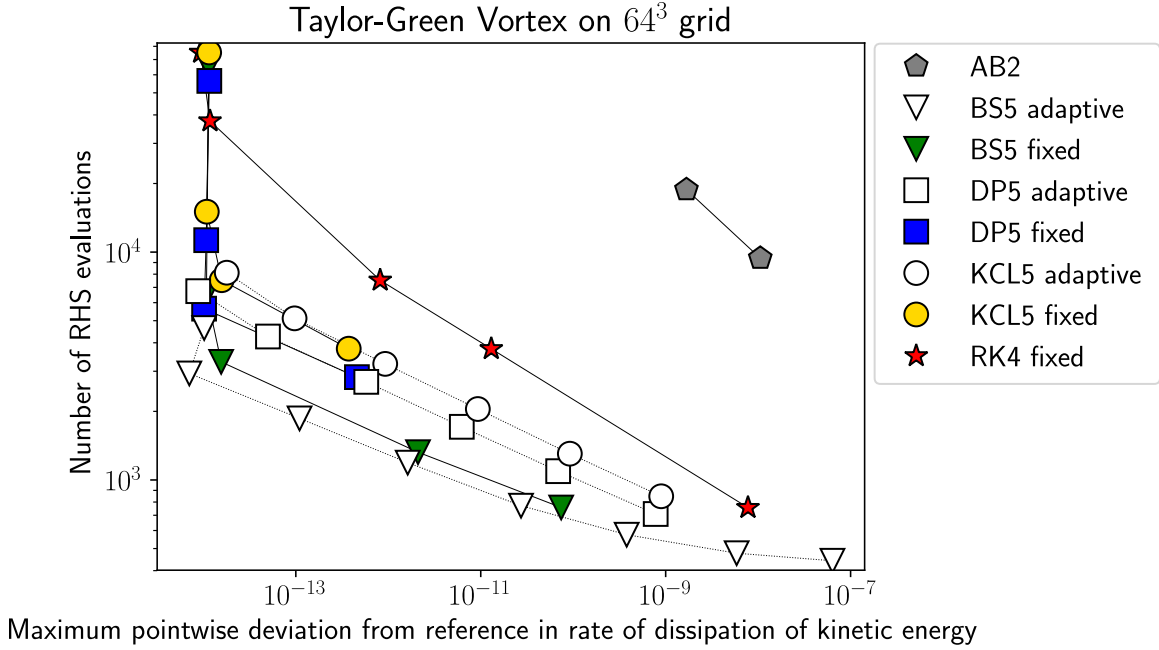
Figure 1: Comparison of time integration methods at Re = 280.

3.1.1 Re=280

Reference values for the rate of dissipation of kinetic energy are available from [5]. Here a 64^3 grid was used, which is sufficient for DNS [5]. Figure 1 shows the error in the calculated rate of dissipation of kinetic energy until $T = 9.4$. The comparison is against a solution generated with RK4 and a timestep of 10^{-4}. A timestep of 10^{-3} was used for the Re=1600 case on a 512^3 grid by [21], and hence we expect a timestep of 10^{-4} to be more than adequate for computing the reference solution.

We used a range of tolerances from 10^{-10} to 10^{-3}. It is clear that that BS5 outperforms the other methods by a wide margin. In particular, the BS5 with adaptive timestep requires 8-10 times less RHS evaluations than the RK4 method for a wide range of error norms.

3.1.2 Re=800

We used a 256^3 grid for the Re=800 runs, as done by [3] which is used as a reference by [9]. A comparison of our results with those reported in [3] using a grid with 256^3 elements is plotted in Figure 2. Our solution did not match exactly with that reported by [3]. However, the discrepancy did not go away even when running on a 512^3 grid. Similar discrepancies appear in [9]; we refer to that work for a more detailed discussion.

We used a reference timestep of 10^{-3} and a range of tolerances from 10^{-9} to 10^{-2}. Results are shown in Figure 3. All 5th-order methods outperform RK4, and the best results are obtained with the adaptive BS5 method, as was found for the TG vortex at Re = 280.

3.1.3 Re=1,600

We used a 512^3 grid, and a reference timestep of 10^{-3} as done in [21]. The TG vortex at this Reynolds number is a well-known test-case, and reference data are available for the the rate of dissipation of kinetic energy, [6].

We used a range of tolerances between 10^{-7} and 10^{-2}. Results are shown in Figure 4. All 5th-order methods outperform RK4 and, once more, the best results are obtained with the adaptive BS5 method.
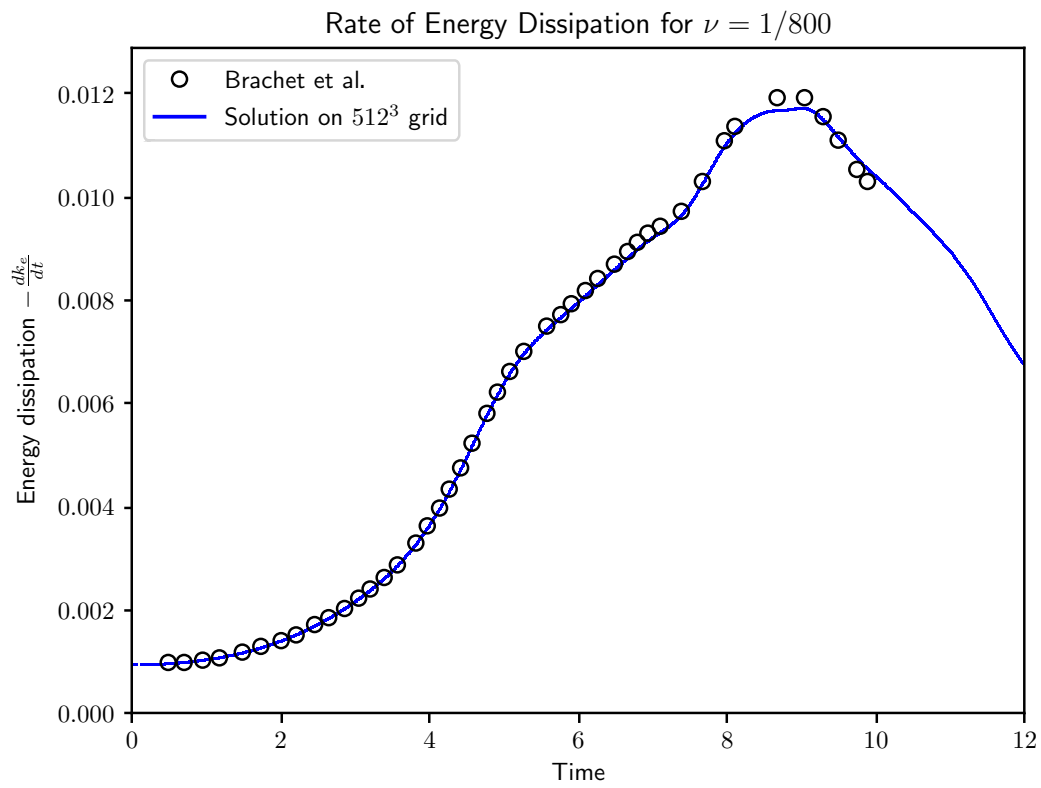
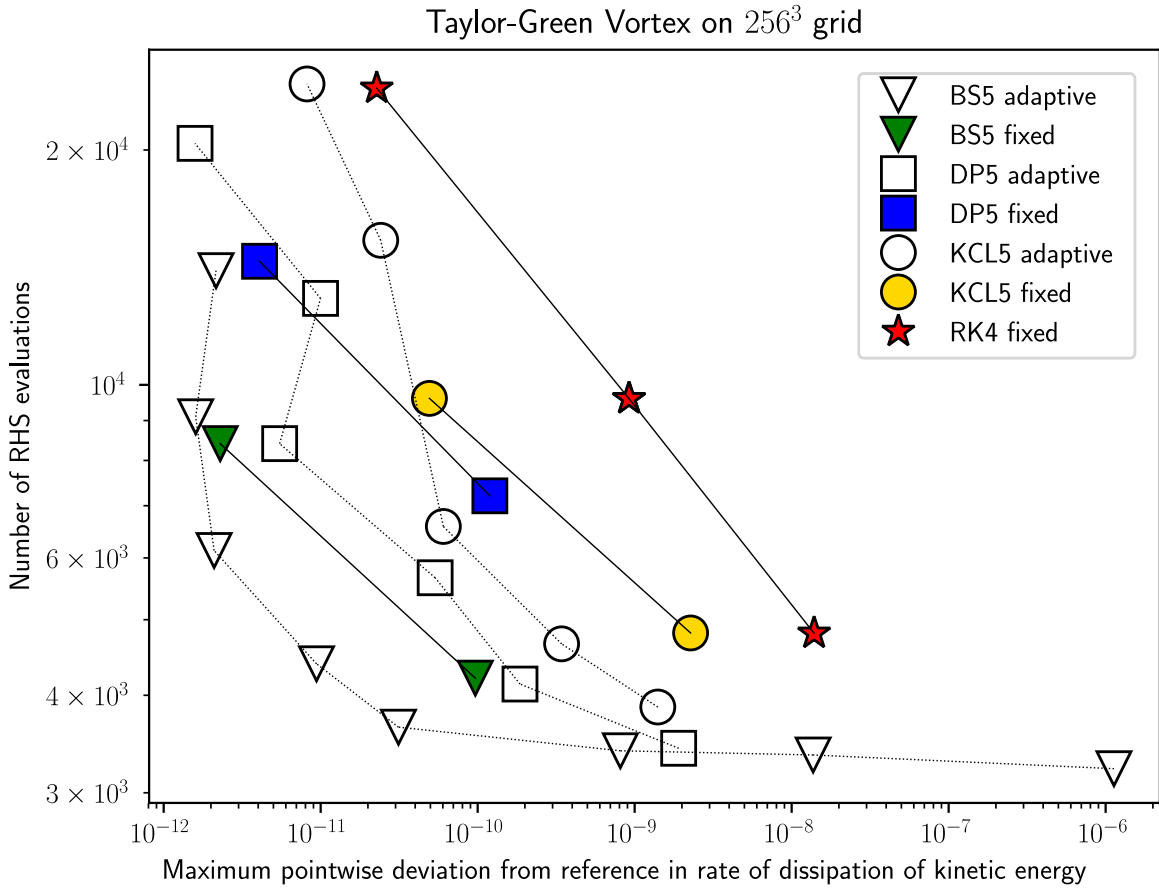Figure 2: Time rate of change of energy dissipation at Re = 800: comparison with results of Brachet et al. [3].

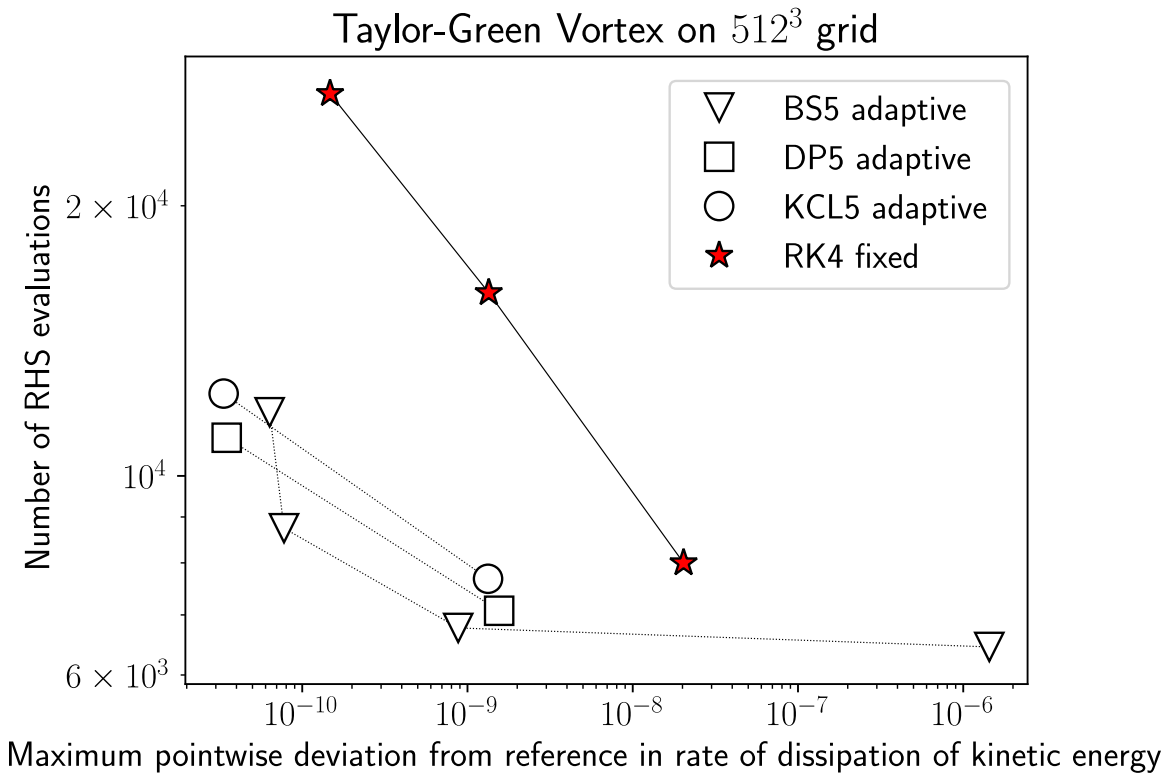Figure 3: Comparison of time integration methods at Re = 800

Figure 4: Comparison of time integration methods at Re = 1600

3.2 Rayleigh–Taylor instability

In this section we solve the incompressible Navier–Stokes equations in the presence of gravity, with variable density, using the Boussinesq approximation:

$$\nabla \cdot u = 0, \tag{3a}$$

$$\frac{\partial u}{\partial t} = u \times \omega - \nabla P + \frac{1}{\text{Re}} \nabla^2 u - \text{Ri}\rho e_z, \tag{3b}$$

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho u) + \frac{1}{\text{RePr}} \nabla^2 \rho. \tag{3c}$$

Here ρ denotes the deviation from ambient density, Ri and Pr denote the Richardson number and the Prandtl number, and e_z indicates the unit vector in the z-direction. In place of (2), the system of ODEs in this case takes the form

$$\partial_t \hat{u}_k = \widehat{(u \times \omega)}_k - \frac{1}{\text{Re}}|k|^2 \hat{u}_k - ik \frac{k \cdot \widehat{(u \times \omega)}_k - \text{Ri}\hat{\rho}_k e_z}{|k|^2} - \text{Ri}\hat{\rho}_k e_z, \tag{4a}$$

$$\partial_t \hat{\rho}_k = -ik \cdot \widehat{(\rho u)}_k - \frac{|k|^2 \hat{\rho}_k}{\text{RePr}}. \tag{4b}$$

The Rayleigh–Taylor instability is a classical fluid-dynamical instability that appears in the presence of gravity when a heavier fluid lies above a lighter fluid. It is of interest here for two reasons. First, it is a widely-simulated phenomenon and serves as a benchmark. Second, it presents an opportunity to greatly improve time-stepping efficiency through automatic step size control. This is because the instability develops slowly at first, so it is expected that very large step sizes can be used initially.

We simulated the single-mode Rayleigh-taylor instability in both 2D and 3D. The initial conditions are based on those of [18], with a slightly smoothed interface between the fluids in order to allow it to be represented accurately in the Fourier basis:

$$u_0 = 0,$$
$$\rho_0 = \frac{1}{2}(\text{erf}(z - z_0 + \zeta(x, y)))\Delta\rho. \tag{5}$$

Here u_0 and ρ_0 are the initial velocity and density fields respectively and erf is the error-function; z_0 is the location of the fluid interface. The $\zeta(x, y)$-term represents a small perturbation to the interface, designed to seed a single-mode instability. The quantity $\Delta\rho$ represents the difference in density between the two fluids; we take $\Delta\rho = 1/10$. This yields an Atwood number of less than 0.05 for which the Boussinesq approximation is reasonable. Both the Richardson number (Ri) and the Prandtl number (Pr) are set to 1.

3.2.1 2D

We use a 512×2048 grid with $\nu = 1/1600$ and take $\zeta = -0.01\cos(x)$. For the reference run, we used a fixed timestep of 10^{-4}. To verify that the requirements for DNS are met, the simulation was also run on a 1024×4096 grid (using BS5 with adaptive timestepping with a tolerance of 10^{-5}) and the results compared to the reference on the coarser grid. The relative L_2 and maximum norm of the difference in the velocity field were both less than 1%. For the density, the (relative) L_2 norm of the difference was less than 1% and in the maximum norm the difference was less than 10%. Comparing the kinetic energy and the rate of dissipation (here negative) of kinetic energy on both grids gives an error of the order 10^{-6} and 10^{-10}, repsectively. Differences in the bubble-height between the two runs are smaller than the resolution of the grid. The simulation was run from $T = 0$ to $T = 70.0$. The density field is plotted for various times in Figure 5.

We used a range of tolerances from 10^{-9} to 10^{-4}. Results for all methods are shown in Figure 6. Clearly all the schemes with adaptive timestepping outperform the schemes with fixed timesteps. Among the methods with adaptive timestep, the DP5 schemes is the most efficient for this problem.
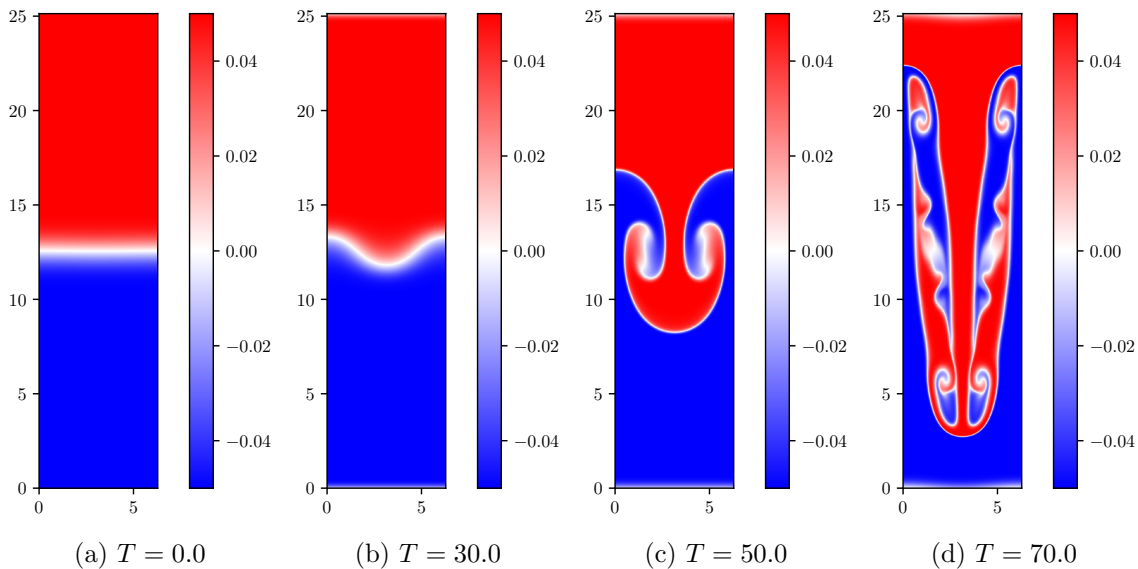
| (a) $T = 0.0$ | (b) $T = 30.0$ | (c) $T = 50.0$ | (d) $T = 70.0$ |

Figure 5: Density field of Rayleigh–Taylor instability on 512×2048 grid

3.2.2   3D

Here a $256 \times 256 \times 1024$ grid was used with $\nu = 10^{-3}$. These parameters are similar to those used in [29], though our simulation differs in that we use a periodic domain: $\zeta(x, y) = 0.01 \cos(x) \cos(y)$. We used a reference timestep of 2×10^{-3}. As in the $2D$-case, we ran a reference simulation on a finer ($512 \times 512 \times 2048$) grid and compared the results to those on the coarser grid. For both density and velocity, the (relative) L_2-norm of the difference was less than 1%. In the maximum-norm, the (relative) norm of the difference was 2.5% and 20% for the velocity field and the density, respectively. Differences of the rate of dissipation of kinetic energy and of the kinetic energy were both on the order of 10^{-10}. Differences in bubble-height were within the resolution of the grid. The simulation was run from time $T = 0$ to $T = 60.0$. A plot of the density field of the solution is plotted at various times in Figure 7.

We used a range of tolerances from 10^{-8} to 10^{-4}. Results for all methods are shown in Figure 8. All the scheme with adaptive timestep outperform the schemes with fixed timestep, except for the BS5 scheme with a fixed timestep whose number of RHS evaluations are close to those of the KCL5. Among the methods with adaptive timestep, the BS5 scheme is the most efficient for this problem.

3.3   Homogeneous isotropic turbulence

Generally speaking, turbulent flows consist of vortices of various scales interacting with each other. Energy is transferred from vortices of larger scale to energy-dissipative vortices of smaller scale. This physical process is known as the turbulent energy cascade. In this context, the study of homogeneous, isotropic turbulence (HIT) is very important for two reasons: i) the smallest turbulent structures in most turbulent flows have an almost isotropic behaviour and therefore it is hoped that these small structures, often not represented in numerical simulations, can be modelled correctly, ii) it is possible to study and understand an important part of HIT analytically. We are particularly interested in proposing improved time integration for HIT since some of the largest DNS runs ever have been devoted to this problem [28, 13, 7, 26, 27].

We simulated homogeneous isotropic turbulence on the cube $[0, 2\pi]^3$. The initial conditions used were based on [24] and were initialized pointwise with each component at each point having random phase and absolute value (i.e., amplitude) given by

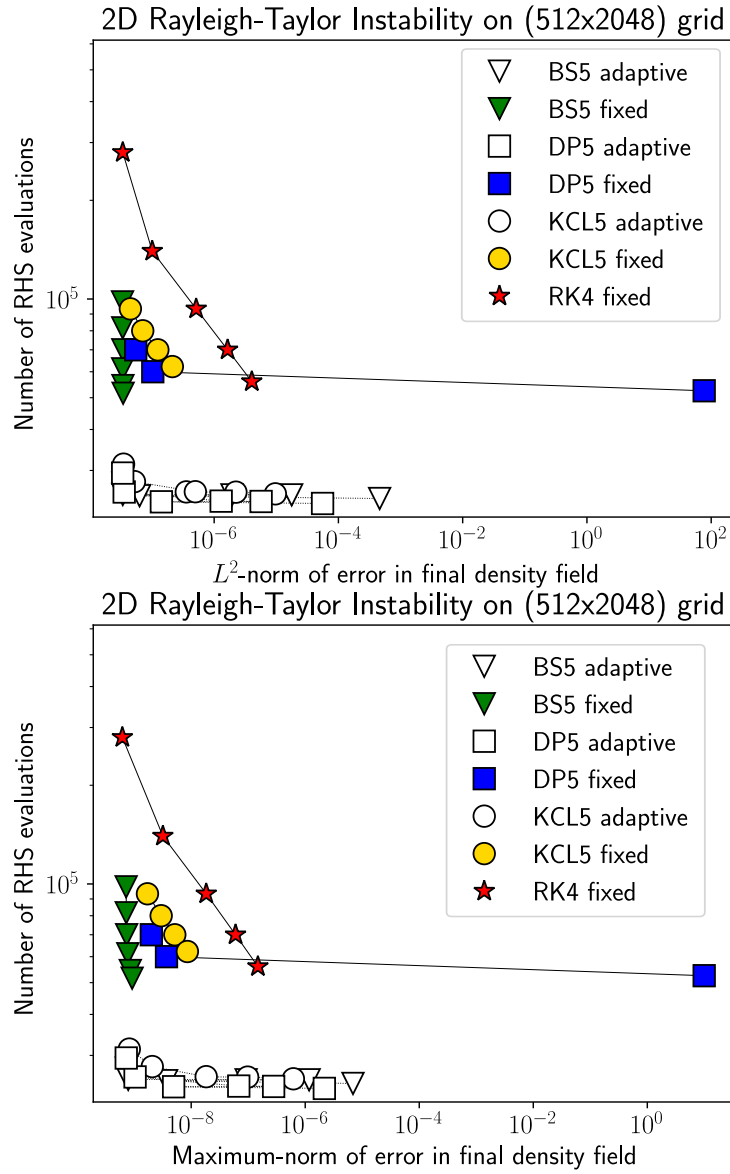$$\frac{C|k|2N^3}{(2\pi)^3} \exp\left(\frac{-|k|^2}{a^2}\right), \tag{6}$$

Figure 6: Comparison of time integration methods at $\nu^{-1} = 1600$

(a) $T = 0.0$ (b) $T = 20.0$ (c) $T = 40.0$ (d) $T = 60.0$
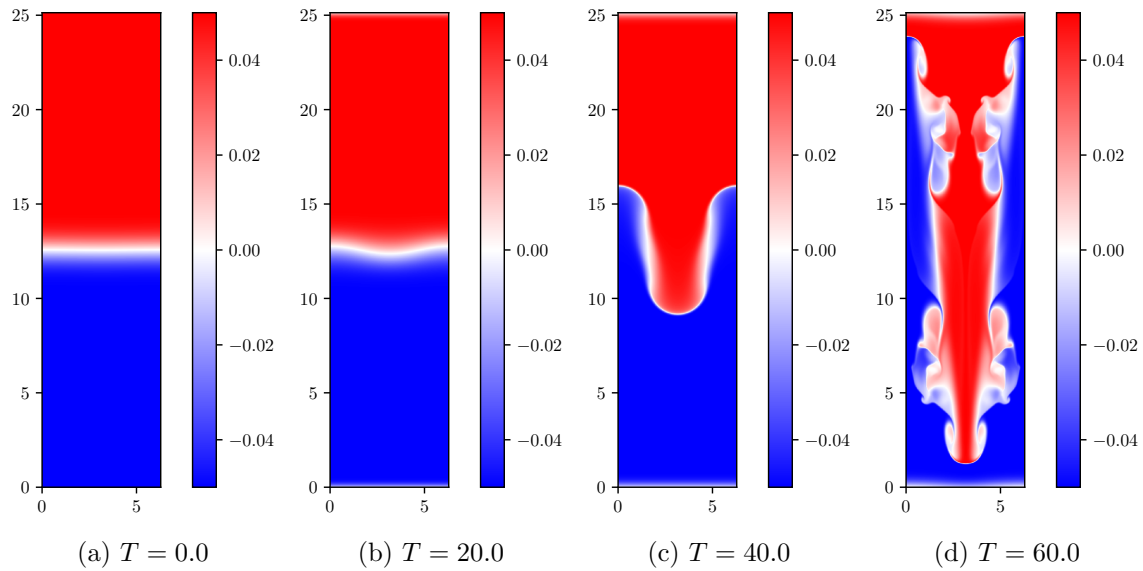
Figure 7: Slice of density-field of Rayleigh–Taylor instability on $256 \times 256 \times 1024$ grid
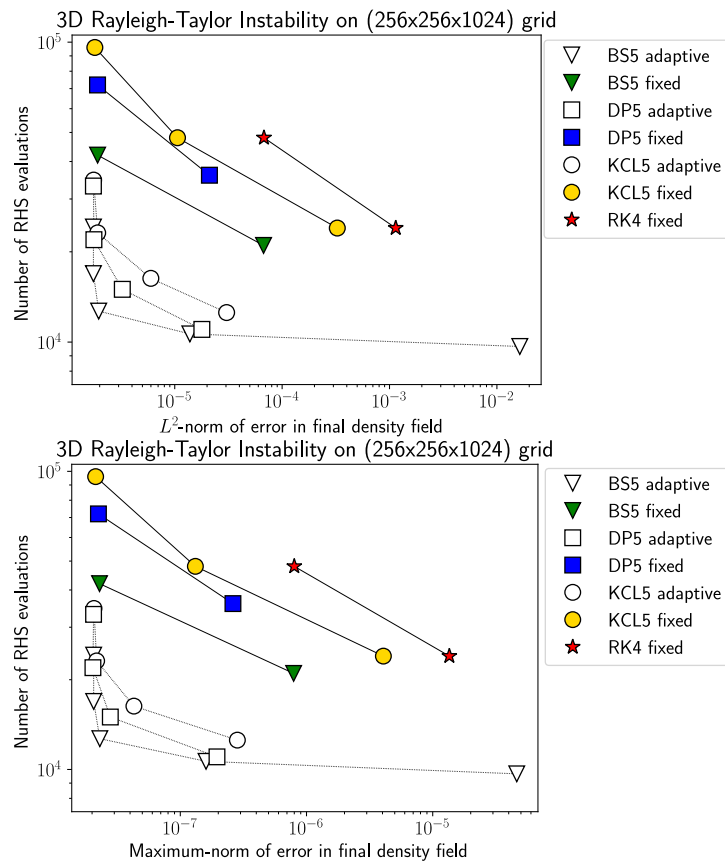

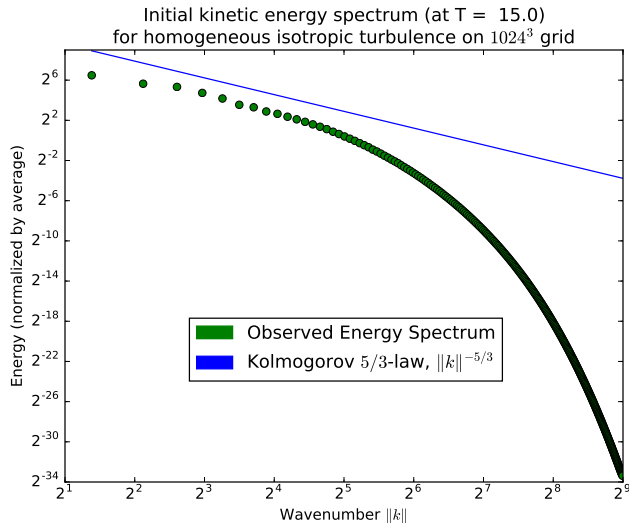Figure 8: Comparison of time integration methods at $\nu^{-1} = 800$

Figure 9: Initial energy spectrum for $\nu^{-1} = 2000$ HIT run

where k is the wavenumber, and $a = 9.5$. The constant C was chosen so that the total kinetic energy was of a similar magnitude as in the Taylor–Green vortex for the 256^3 and 512^3 case. We subtracted $\frac{(k \cdot \hat{u})k}{|k|^2}$ to make the initial conditions divergence-free.

3.3.1 Forcing

In order to replace the energy lost, we used a method of determinstic forcing after each timestep based on the approach presented in [24]. Here the lower wavenumbers (with $0 < |k| \le k_f$) are scaled by an appropriate factor to compensate for energy lost by the system to diffusion.

We calculated the energy spectra of the initial velocity fields like done in [17] . Figure 9 shows the energy spectrum for the 1024^3 run, which had $\nu^{-1} = 2000$ and $k_f = 8$.

To generate an appropriate initial condition, the simulation was run from $T = 0$ to $T = 15.0$ using the BS5 time-integrator with adaptive stepsize (with tolerance 10^{-7} on the 256^3 run and 10^{-6} on the others). The resulting energy spectrum is shown in Figure 9. We see that the high-order adaptive integrator yields the expected statistical state. After reaching this state, the adaptive timestepping gave a step size that corresponds to a CFL number of about 0.6 for the 1024^3 run.

3.3.2 Decaying turbulence test runs

To test each integrator, the simulation was restarted from $t = 15$ and run for a short time without forcing. A slice of the vorticity of the solution at $t = 15$ is shown in Figure 10.

We ran the simulation on various grid sizes with ν chosen based on the value of ν for the Taylor–Green vortex and on [2]. As a very stringent test of the time integrators, we compare pointwise solution values of the resulting turbulent flow field.[2]

Results for a 256^3 grid are shown in Figure 11. Here we used $\nu^{-1} = 800$ and $k_f = 4\sqrt{2}$. The simulation was run from $T = 15.0$ to $T = 16.0$. As a reference we used RK4 with timestep 10^{-3}. We used a range of tolerances between 10^{-8} and 10^{-4}.

Results for a 512^3 grid are shown in Figure 12. Here we used $\nu^{-1} = 1600$ and $k_f = 8$. The simulation was run from $T = 15.0$ to $T = 15.5$. As a reference we used RK4 with timestep 5×10^{-4}. We used a range of tolerances between 10^{-8} and 10^{-4}.

---

[2]After running these (expensive) simulations it was found that the zero and Nyquist frequencies of the first Fourier transform had a non-zero imaginary part. This should not affect the real part of the solution (on which all comparisons are based, and which is evident in the convergence results) but would slightly reduce the step size chosen by adaptive time-stepping, in a uniform way for all methods. The energy spectrum in Figure 9 includes only the real part.

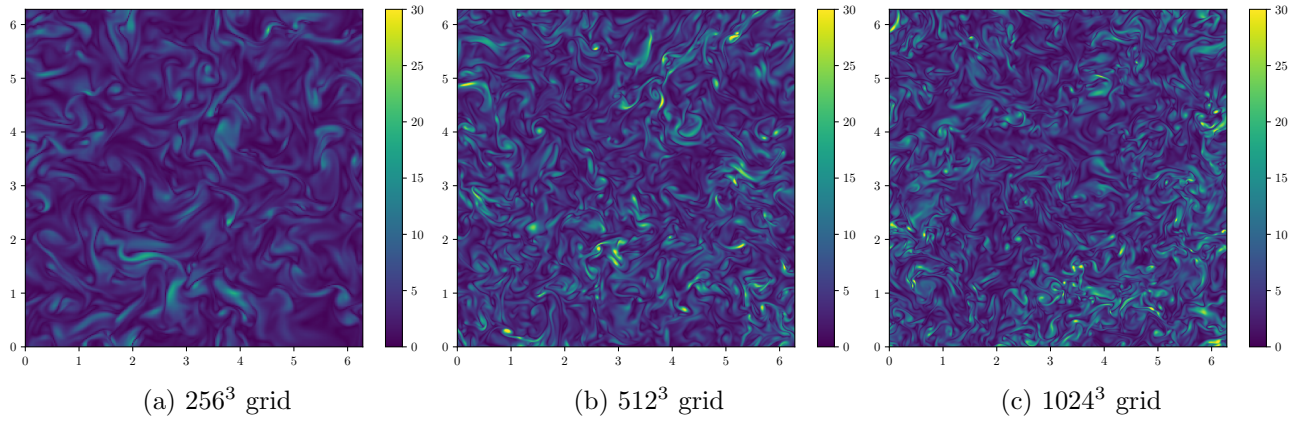(a) 256^3 grid   (b) 512^3 grid   (c) 1024^3 grid

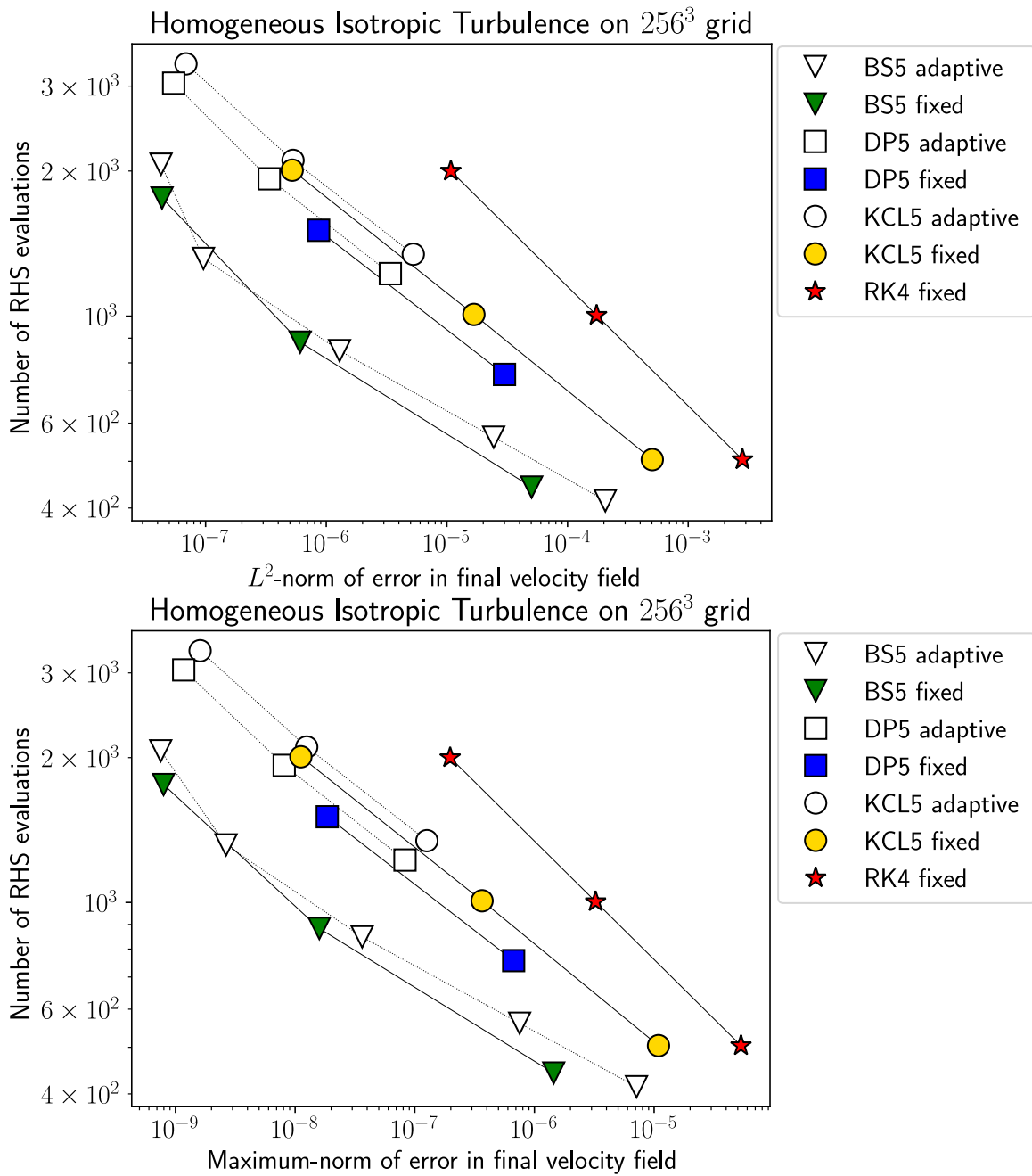Figure 10: Vorticity slice ($x = 0$) at $t = 15$

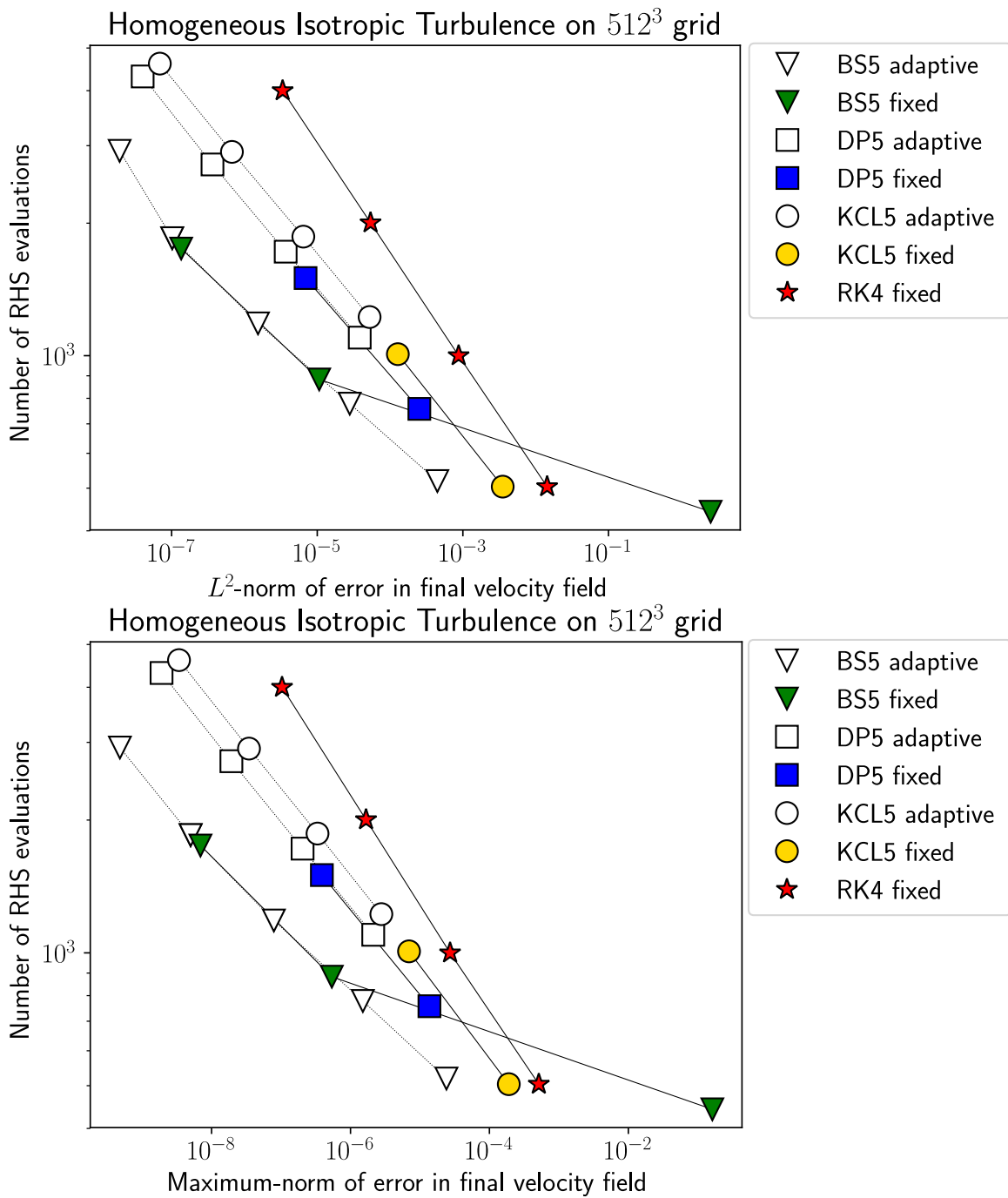Figure 11: Comparison of time integration methods at $\nu^{-1} = 800$

Figure 12: Comparison of time integration methods at $\nu^{-1} = 1600$

Results for a 1024^3 grid are shown in Figure 13. Here we used $\nu^{-1} = 2000$ and $k_f = 8$. The simulation was run from $T = 15.0$ to $T = 15.5$. As a reference we used RK4 with timestep 5×10^{-4}. We used a range of tolerances between 10^{-8} and 10^{-5}.

As for the other test cases presented herein, the BS5 method with adaptive timestep requires less RHS function evaluations. However, for this flow problem, the BS5 scheme with a fixed timestep rqeuires a number of RHS evaluations very close if not better than the BS5 with automatic timestep selection.

4   Discussion

While one should be cautious about drawing general results from a small selection of test problems, our results consistently suggest that:

- High-order, adaptive time stepping yields accurate results for pseudospectral DNS of turbulent flow. The use of significantly larger time step sizes with such methods does not negatively affect the resolution of fine-scale features of the flow when compared with the use of lower-order methods (and correspondingly smaller step sizes).

- Optimized fifth-order methods, especially the BS5 pair, are more efficient than lower-order methods whenever moderate to high accuracy is desired. Specifically, similar accuracy can be obtained at a cost that is reduced by 2x to 10x.

- Error estimation and step size adaptivity can be highly beneficial for problems involving the development of instability from an initially laminar flow. It can also be useful in automatically finding an appropriate step size, even if the characteristics of the flow do not change significantly during the simulation.

Similar work remains to be done for the development of time integrators for incompressible flow in the presence of boundaries. Many questions remain to be investigated regarding the application of modern time discretizations to pseudo-spectral DNS. In particular, we expect that exponential methods (wherein the linear diffusive terms are handled directly via the matrix exponential) may yield even more substantial improvements. Improved step size controllers such as those of [23] will also likely show improvements over the more standard techniques used here.

5   Acknowledgements

References

[1] P. Bogacki and L. F. Shampine. An efficient Runge–Kutta (4, 5) pair. *Computers & Mathematics with Applications*, 32(6):15–28, 1996.

[2] R. Bogdan, P. Hossein, A. Orlando, and Lian-Ping W. Settling velocity of small inertial particles in homogeneous isotropic turbulence from high-resolution DNS. *International Journal of Multiphase Flow*, 83:217–231, 2016.

[3] M. E. Brachet, D. I. Meiron, S. A. Orszag, BG. Nickel, R. H. Morf, and U. Frish. Small-scale structure of the Taylor–Green vortex. *Journal of Fluid Mechanics*, 130:411–452, 1983.
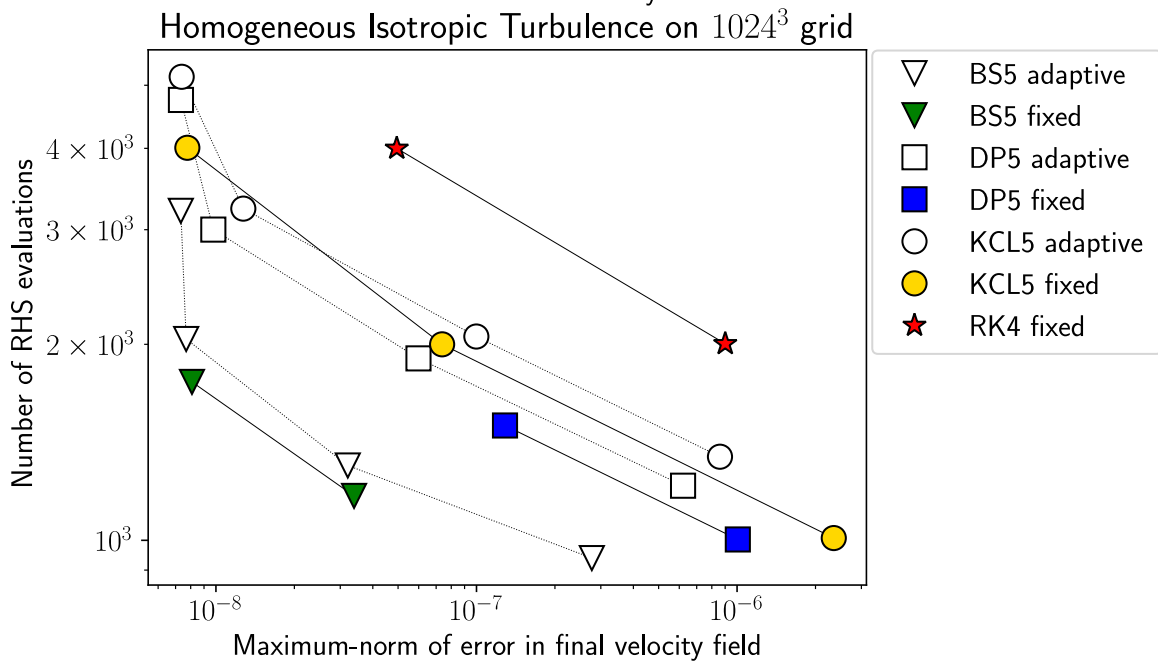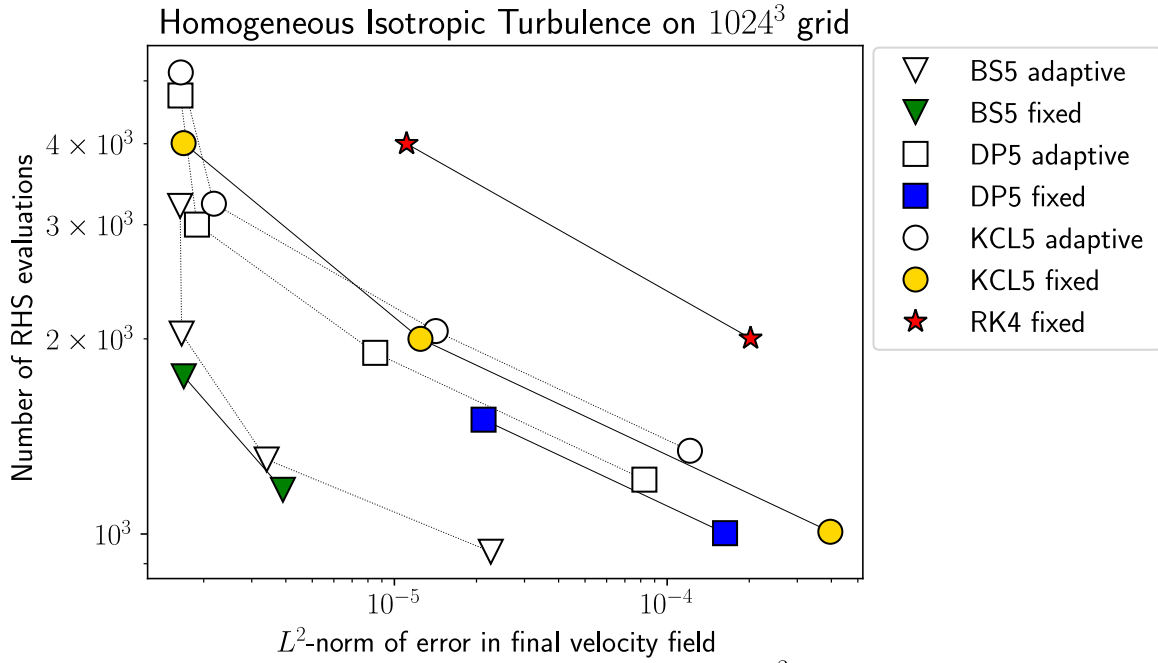
Figure 13: Comparison of time integration methods at $\nu^{-1} = 2000$

[4] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. Basic approaches to constructing spectral methods. *Spectral Methods: Fundamentals in Single Domains*, pages 117–165, 2006.

[5] J. B. Chapelier, M. De La Llave Plata, F. Renac, and E. Martin. Final abstract for ONERA Taylor-Green DG participation. In *Proceedings of 1st International Workshop on High-Order CFD Methods at the 50th AIAA Aerospace Sciences Meeting*, pages 7–8, 2012.

[6] C. de Wiart, K. Hillewaert, M. Duponcheel, and G. Winckelmans. Assessment of a discontinuous Galerkin method for the simulation of vortical flows at high Reynolds number. *International Journal for Numerical Methods in Fluids*, 74(7):469–493, 2014.

[7] D. A. Donzis, P. K. Yeung, and K. R. Sreenivasan. Dissipation and enstrophy in isotropic turbulence: Resolution effects and scaling in direct numerical simulations. *Physics of Fluids*, 20(4):045108, 2008.

[8] J. R. Dormand and P. J. Prince. A family of embedded Runge-Kutta formulae. *Journal of Computational and Applied Mathematics*, 6(1):19–26, 1980.

[9] G. J. Gassner and A. D. Beck. On the accuracy of high-order discretizations for underresolved turbulence simulations. *Theoretical and Computational Fluid Dynamics*, pages 1–17, 2013.

[10] B. Hadri *et al.* Overview of the KAUST's Cray X40 System – Shaheen II. In *Proceedings of the Cray User Group Meeting*, Chicago, USA, May 2015.

[11] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving ordinary differential equations I: Nonstiff Problems*. Springer Series in Computational Mathematics. Springer, second edition, 1993.

[12] M. Iovieno, C. Cavazzoni, and D. Tordella. A new technique for a parallel dealiased pseudospectral Navier–Stokes code. *Computer physics communications*, 141(3):365–374, 2001.

[13] T. Ishihara, Y. Kaneda, M. Yokokawa, K. Itakura, and A. Uno. Small-scale statistics in high-resolution direct numerical simulation of turbulence: Reynolds number dependence of one-point velocity gradient statistics. *Journal of Fluid Mechanics*, 592:335–366, 2007.

[14] Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001–.

[15] C. A. Kennedy, M. H. Carpenter, and R. Michael Lewis. Low-storage, explicit Runge–Kutta schemes for the compressible Navier–Stokes equations. *Applied Numerical Mathematics*, 35(3):177–219, November 2000.

[16] W. Kutta. Beitrag zur näherungsweisen integration totaler differentialgleichungen. *Zeitschrift für Angewandte Mathematik und Physik*, 46:435–453, 1901.

[17] A. G. Lamorgese, D. A. Caughey, and S. B. Pope. Direct numerical simulation of homogeneous turbulence with hyperviscosity. *Physics of Fluids*, 17(1):015106, 2005.

[18] D. Livescu, T. Wei, and M. R. Petersen. Direct numerical simulations of Rayleigh–Taylor instability. In *Journal of Physics: Conference Series*, volume 318, page 082007. IOP Publishing, 2011.

[19] V. K. Mahendra, C. Anando, Y. Rakesh, and R. Sandeep. Object-oriented pseudo-spectral code TARANG for turbulence simulation. In *Proceedings of the ATIP/A* CRC Workshop on Accelerator Technologies for High-Performance Computing: Does Asia Lead the Way?*, page 4. A* STAR Computational Resource Centre, 2012.

[20] M. Mortensen. SpectralDNS code. `http://github.com/spectralDNS/spectralDNS`, 2016.

[21] M. Mortensen and H. P. Langtangen. High performance Python for direct numerical simulations of turbulent flows. *Computer Physics Communications*, 203:53–65, 2016.

[22] S. B. Pope. *Turbulent flows*. Cambridge University Press, 2000.

[23] Gustaf Söderlind. Time-step selection algorithms: Adaptivity, control, and signal processing. *Applied numerical mathematics*, 56(3):488–502, 2006.

[24] N. P. Sullivan, S. Mahalingam, and R. M. Kerr. Deterministic forcing of homogeneous, isotropic turbulence. *Physics of Fluids*, 6(4):1612–1614, 1994.

[25] B. Teaca, M. K. Verma, Bernard Knaepen, and D. Carati. Energy transfer in anisotropic magnetohydrodynamic turbulence. *Physical Review E*, 79(4):046312, 2009.

[26] P. K. Yeung, D. A. Donzis, and K. R. Sreenivasan. Dissipation, enstrophy and pressure statistics in turbulence simulations at high reynolds numbers. *Journal of Fluid Mechanics*, 700:5–15, 2012.

[27] P. K. Yeung, X. M. Zhai, and K. R. Sreenivasan. Extreme events in computational turbulence. *Proceedings of the National Academy of Sciences*, 112(41):12633–12638, 2015.

[28] M. Yokokawa, K. Itakura, A. Uno, T. Ishihara, and Y. Kaneda. 16.4-Tflops direct numerical simulation of turbulence by a Fourier spectral method on the earth simulator. In *Supercomputing, ACM/IEEE 2002 Conference*, pages 50–50. IEEE, 2002.

[29] Y.-N. Young, H. Tufo, A. Dubey, and R. Rosner. On the miscible Rayleigh–Taylor instability: two and three dimensions. *Journal of Fluid Mechanics*, 447:377–408, 2001.

A Coefficients of Runge–Kutta pairs

For the convenience of the reader, we give here the coefficients of each Runge–Kutta pair used in this work.

A.1 Dormand–Prince 5(4) pair (DP5)

$$
\begin{array}{c|ccccccc}
\frac{1}{5} & \frac{1}{5} \\[4pt]
\frac{3}{10} & \frac{3}{40} & \frac{9}{40} \\[4pt]
\frac{4}{5} & \frac{44}{45} & -\frac{56}{15} & \frac{32}{9} \\[4pt]
\frac{8}{9} & \frac{19372}{6561} & -\frac{25360}{2187} & \frac{64448}{6561} & -\frac{212}{729} \\[4pt]
1 & \frac{9017}{3168} & -\frac{355}{33} & \frac{46732}{5247} & \frac{49}{176} & -\frac{5103}{18656} \\[4pt]
1 & \frac{35}{384} & & \frac{500}{1113} & \frac{125}{192} & -\frac{2187}{6784} & \frac{11}{84} \\[4pt]
\hline
& \frac{35}{384} & & \frac{500}{1113} & \frac{125}{192} & -\frac{2187}{6784} & \frac{11}{84} \\[4pt]
& \frac{5179}{57600} & & \frac{7571}{16695} & \frac{393}{640} & -\frac{92097}{339200} & \frac{187}{2100} & \frac{1}{40}
\end{array}
\tag{7}
$$

The first set of weights listed is used to advance the solution. The second set ($\hat{b}$) is used only for error estimation.

A.2 KCL5 low-storage scheme

Because of the special structure and desirable low-storage implementation of this scheme, the Butcher tableau is not the most useful format for its presentation. Instead, we refer the reader to [15, p. 190] where the low-storage coefficients are presented.

A.3   Bogack–Shampine 5(4) pair (BS5)

$$
\begin{array}{c|ccccccc}
\frac{1}{6} & \frac{1}{6} \\[4pt]
\frac{2}{9} & \frac{2}{27} & \frac{4}{27} \\[4pt]
\frac{3}{7} & \frac{183}{1372} & -\frac{162}{343} & \frac{1053}{1372} \\[4pt]
\frac{2}{3} & \frac{68}{297} & -\frac{4}{11} & \frac{42}{143} & \frac{1960}{3861} \\[4pt]
\frac{3}{4} & \frac{597}{22528} & \frac{81}{352} & \frac{63099}{585728} & \frac{58653}{366080} & \frac{4617}{20480} \\[4pt]
1 & \frac{174197}{959244} & -\frac{30942}{79937} & \frac{8152137}{19744439} & \frac{666106}{1039181} & -\frac{29421}{29068} & \frac{482048}{414219} \\[4pt]
1 & \frac{587}{8064} & & \frac{4440339}{15491840} & \frac{24353}{124800} & \frac{387}{44800} & \frac{2152}{5985} & \frac{7267}{94080} \\[4pt]
\hline
 & \frac{587}{8064} & & \frac{4440339}{15491840} & \frac{24353}{124800} & \frac{387}{44800} & \frac{2152}{5985} & \frac{7267}{94080} \\[4pt]
 & \frac{2479}{34992} & & \frac{123}{416} & \frac{612941}{3411720} & \frac{43}{1440} & \frac{2272}{6561} & \frac{79937}{1113912} & \frac{3293}{556956}
\end{array}
\tag{8}
$$

The first set of weights listed is used to advance the solution. The second set $(\hat{b})$ is used only for error estimation.