



# Time-parallel simulation of the decay of homogeneous turbulence using Parareal with spatial coarsening

Thibaut Lunet<sup>1,2</sup> · Julien Bodart<sup>1</sup> · Serge Gratton<sup>3</sup> · Xavier Vasseur<sup>1</sup>

Received: 8 May 2017 / Accepted: 29 September 2017 / Published online: 30 May 2018  
© Springer-Verlag GmbH Germany, part of Springer Nature 2018

## Abstract

Direct Numerical Simulation of turbulent flows is a computationally demanding problem that requires efficient parallel algorithms. We investigate the applicability of the time-parallel PARAREAL algorithm to an instructional case study related to the simulation of the decay of homogeneous isotropic turbulence in three dimensions. We combine a PARAREAL variant based on explicit time integrators and spatial coarsening with the space-parallel HYBRID Navier–Stokes solver. We analyse the performance of this space–time parallel solver with respect to speedup and quality of the solution. The results are compared with reference data obtained with a classical explicit integration, using an error analysis which relies on the energetic content of the solution. We show that a single PARAREAL iteration is able to reproduce with high fidelity the main statistical quantities characterizing the turbulent flow field.

**Keywords** Direct numerical simulation · Explicit time integrator · High-order method in space · Navier–Stokes equations · PARAREAL · Space–time parallelism

## 1 Introduction

The numerical solution of time dependent partial differential equations is of interest in many applications in Computational Science and Engineering. The recent advent of computing platforms with more, but not faster, processors currently requires the design of new parallel algorithms that must be able to exploit more concurrency to provide a fast time-to-solution. In this respect, parallel-in-time and space–time methods are considered as promising candidates [10]. Indeed, such methods enable the exploitation of possibly substantially more computational

resources than purely space-parallel methods with sequential time stepping.

A popular parallel-in-time method based on a decomposition of the space–time domain in time is the PARAREAL algorithm introduced by Lions, Maday and Turinici [28]. PARAREAL relies on the availability of a cheap coarse time integrator that provides guesses of the solution at several instants. Given these starting values, a fine time integrator is applied concurrently. The results are then used to propagate a correction to the guesses by using the coarse integrator serially over the time slices. As shown in [18], PARAREAL can be derived as both a multigrid method in time or as a multiple shooting method along the time axis; see [16] for further comments and details on the classification of time parallel methods. Due to its non-intrusiveness, PARAREAL is one of the most widely used time-parallel algorithms. Successful applications have been considered in Computational Fluid Dynamics (CFD) [6,14], neutron transport [30], plasma physics [39] and skin permeation [23] to name a few. While a relative efficiency has been obtained on diffusive problems [14,23], the applicability of PARAREAL to hyperbolic or advection-dominated problems is still an open issue; see [35] for a recent analysis and [3,41] for early attempts. Although modifications of PARAREAL for hyperbolic problems have been proposed [9,12,13,15,37], these enhancements gener-

---

This research is partly funded by “Région Occitanie/ Pyrénées-Méditerranée” as part of the “Développement de nouvelles stratégies pour le calcul massivement parallèle à l’échelle exa en mécanique des fluides numérique (DENSE)” project.

---

✉ Thibaut Lunet  
Thibaut.Lunet@unige.ch

<sup>1</sup> ISAE-SUPAERO, 10 avenue Edouard Belin, BP 54032, 31055 Toulouse Cedex 4, France

<sup>2</sup> CERFACS, 42 avenue Gaspard Coriolis, 31057 Toulouse Cedex 1, France

<sup>3</sup> INPT-IRIT, University of Toulouse and ENSEEIHT, 2 rue Camichel, BP 7122, 31071 Toulouse Cedex 7, France

ally imply a significant overhead that leads to a degradation of the parallel efficiency. Hence this seems to prevent the use of PARAREAL for a certain class of important problems in Computational Fluid Dynamics.

In this manuscript we investigate this critical issue by applying PARAREAL to the direct numerical simulation of complex turbulent compressible flows in three dimensions. We specifically focus on the simulation of the decay of homogeneous isotropic turbulence, a canonical test case known as the simplest configuration which incorporates fundamental and relevant turbulence mechanisms [31]. As identified in [45], regarding parallel-in-time integration, interesting features of turbulent flows are their chaotic nature and their strongly unsteady behavior, which typically requires explicit integration methods [7]. This test case incorporates these issues at a very moderate cost, making this model problem an interesting instructional candidate for time-parallel algorithms. To the best of our knowledge, the performance of PARAREAL on such a test case has not yet been studied. In the following, we concentrate on PARAREAL with spatial coarsening [14] and explicit time integrators. We employ the HYBRID Navier–Stokes solver for the spatial discretization, for which excellent parallel scaling properties have been obtained [4]. With this setting, we aim at investigating if this application could benefit from time parallelization. For such a purpose, we perform a numerical study on the role of various parameters on the convergence of PARAREAL with spatial coarsening. This extensive numerical study is the main contribution of the manuscript.

The manuscript is organized as follows. In Sect. 2 we briefly present the PARAREAL time-parallel time integration method. In Sect. 3 we describe the set of governing time-dependent partial differential equations and explain the relevance of the canonical test case concerning the direct numerical simulation of turbulent flows. Then we present detailed numerical experiments to understand the convergence of PARAREAL in Sect. 4. Finally, as a conclusion, we draw first lessons in Sect. 5.

## 2 Time parallelization using Parareal

We briefly introduce PARAREAL [28], a popular method for the time parallel solution of nonlinear initial value problems. Then we describe a variant based on spatial coarsening first proposed in [14]. Finally, we include a theoretical model for the expected speedup and parallel efficiency of both algorithms.

### 2.1 General setting

PARAREAL aims at solving the initial value problem of the form

$$\frac{dU}{dt} = f(U(t), t), \quad U(0) = U_0, \quad t \in [0, T], \quad (1)$$

with  $f : \mathbb{R}^p \times \mathbb{R}^+ \rightarrow \mathbb{R}^p$ ,  $U(t) \in \mathbb{R}^p$ ,  $U_0 \in \mathbb{R}^p$ ,  $p$  being the total number of degrees of freedom and  $T$  a positive real value. Here, the problem (1) arises from the spatial discretization of a nonlinear system of partial differential equations (PDEs) through the “method-of-lines” [40]. We decompose the global time interval  $[0, T]$  into  $N$  uniform time slices  $[t_{n-1}, t_n]$ ,  $n = 1, \dots, N$ , where  $N$  is the number of processes to be considered for the time parallelization only. In the following, we denote by  $U_n$  the approximation of  $U$  at time  $t_n$ , i.e.,  $U_n \approx U(t_n)$ . Let  $\mathcal{F}_{t_{n-1} \rightarrow t_n}^{\delta_t}(U_{n-1})$  denote the result of approximately integrating (1) on the time slice  $[t_{n-1}, t_n]$  from a given starting value  $U_{n-1}$  using a fine time integrator  $\mathcal{F}$  with time increment  $\delta_t$ . Similarly, we introduce a second time integrator  $\mathcal{G}$  (referred to as the coarse propagator, with time increment  $\Delta_t$ ), which has to be much cheaper than  $\mathcal{F}$  in terms of computational time, with possible reduced accuracy. Finally, for ease of exposition, we assume in this analysis that an integer number of both  $\delta_t$  and  $\Delta_t$  covers a time slice exactly.

The prediction step of PARAREAL consists of computing a first guess of the starting values  $U_n^0$  at the beginning of each time slice by

$$U_n^0 = \mathcal{G}_{t_{n-1} \rightarrow t_n}^{\Delta_t}(U_{n-1}^0), \quad U_0^0 = U_0, \quad (2)$$

with  $n = 1, \dots, N$ . A correction iteration is then applied concurrently on each time slice:

$$U_n^k = \mathcal{F}_{t_{n-1} \rightarrow t_n}^{\delta_t}(U_{n-1}^{k-1}) + \mathcal{G}_{t_{n-1} \rightarrow t_n}^{\Delta_t}(U_{n-1}^k) - \mathcal{G}_{t_{n-1} \rightarrow t_n}^{\Delta_t}(U_{n-1}^{k-1}), \quad (3)$$

where  $U_n^k$  denotes the approximation of  $U$  at time  $t_n$  at the  $k$ -th iteration of PARAREAL ( $k = 1, \dots, K$ ,  $n = 1, \dots, N$ ). While the application of  $\mathcal{F}$  can be performed independently for each time slice PARAREAL remains limited by the sequential nature of the coarse integration (3). Hence, PARAREAL will bring a reduction of the total computational time with respect to a direct time-serial integration, only if the application of  $\mathcal{G}$  is cheap enough and if the total number of iterations  $K$  of PARAREAL is small. We recall that PARAREAL converges after  $N$  iterations to the approximation of the exact solution [17].

### 2.2 Variant of Parareal based on explicit time integrators and on spatial coarsening

The PARAREAL parallel in time shooting method [18] is generally used in combination with implicit time integrators; see, e.g., [8, 12, 23, 42] for applications related to time-dependent PDEs. In this context procedure, the coarse integrator  $\mathcal{G}$  is

obtained by simply choosing  $\Delta_t > \delta_t$ . However, this strategy is usually not applicable when explicit time integrators are favored, since the time step is notably bounded by numerical stability conditions (such as the Courant–Friedrichs–Levy (CFL) condition).

As a cure, Ruprecht and Krause [37] have considered a coarse propagator  $\mathcal{G}$  with lower accuracies in both time and space. Using fewer degrees of freedom at the coarse level is also possible as proposed in [14] for the numerical solution of the two-dimensional incompressible Navier–Stokes equations. This has been later investigated in [12] for the Detached Eddy Simulation of the three-dimensional incompressible Navier–Stokes equations and for the simulation of two-dimensional plasma turbulence [39], respectively. In this setting, we denote by  $\widehat{\mathcal{G}}$  the corresponding propagator on the coarse spatial grid (which now involves  $\widehat{p}$  degrees of freedom with  $\widehat{p} < p$ ), while  $\mathcal{R}$  and  $\mathcal{I}$  represent the spatial restriction and interpolation operators, respectively. The prediction step of PARAREAL with spatial coarsening is now

$$U_n^0 = \mathcal{I} \widehat{\mathcal{G}}^{\Delta_t} (\mathcal{R}(U_{n-1}^0)), \quad U_0^0 = U_0, \quad (4)$$

with  $n = 1, \dots, N$ . Similarly, the correction iteration of the PARAREAL algorithm with spatial coarsening then becomes

$$U_n^k = \mathcal{F}^{\delta_t} (U_{n-1}^{k-1}) + \mathcal{I} \left( \widehat{\mathcal{G}}^{\Delta_t} (\mathcal{R}(U_{n-1}^k)) - \widehat{\mathcal{G}}^{\Delta_t} (\mathcal{R}(U_{n-1}^{k-1})) \right), \quad (5)$$

with  $k = 1, \dots, \widehat{K}$  and  $n = 1, \dots, N$ . As noted in [36], the convergence of PARAREAL with spatial coarsening will not only depend on the fine and coarse time propagators but also on the restriction and interpolation operators. Hence, we expect (5) to obtain a different convergence rate than (3). In this manuscript, we mostly focus on the variant of PARAREAL based on (4) and (5) with explicit time integration for both the fine and coarse propagators. We next address the expected performance of such a variant.

### 2.3 Expected parallel performance of Parareal

In our setting, parallel-in-time integration is considered as a possibility for additional fine grain parallelism on top of an existing coarse grain spatial decomposition. In a preliminary phase, we have decided to simulate the parallelization in time, whereas the parallelization in space is truly implemented on a distributed memory passing system using the Message Passing Interface (MPI) [19]. This allows us to predict at a very moderate cost if the time parallelization can be relevant in our study. Hence, modelling the expected performance of PARAREAL is of utmost importance. We first analyse the parallel speedup, defined as the ratio of the sequential to the

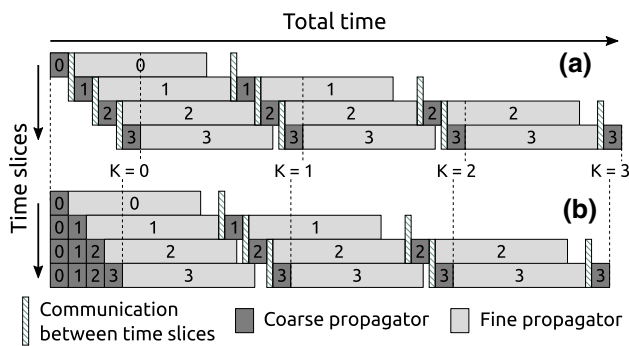
parallel execution time for a given number of processes. As pointed out in [1,2,5,34], the PARAREAL algorithm is flexible enough to accommodate various implementations based on different programming paradigms. In the modeling, we consider a distributed memory implementation to handle the parallelization in time. We refer the reader to [34] for a discussion and analysis of other strategies. We consider a total of  $N_{proc}$  processes for the space–time parallelism with  $N$  processes being devoted to the parallelization in time. In this setting, each time slice associated to a spatial subdomain is assigned to a process.

We first consider the standard PARAREAL algorithm and denote by  $C_{\mathcal{F}}$  the cost of integrating over a given time slice using  $\mathcal{F}^{\delta_t}$  and by  $C_{\mathcal{G}}$  the corresponding cost when using  $\mathcal{G}^{\Delta_t}$ . Since explicit time-integration schemes are used with uniform time steps, we expect both  $C_{\mathcal{F}}$  and  $C_{\mathcal{G}}$  to be proportional to the time slice length. Hence  $C_{\mathcal{F}}$  is equal to  $T_{\mathcal{F}} N_{\delta_t, \mathcal{F}}$ , where  $T_{\mathcal{F}}$  and  $N_{\delta_t, \mathcal{F}}$  are the computational time related to the application of the fine integrator over one time step and the number of time steps done for one time slice, respectively. A similar expression can be found for  $C_{\mathcal{G}}$  (i.e.,  $C_{\mathcal{G}} = T_{\mathcal{G}} N_{\Delta_t, \mathcal{G}}$  with similar notation).

As advocated in [34], we concentrate on an efficient implementation of PARAREAL discussed in both [29, Sec. 5] and [2, Sect. 4] making use of pipelining, i.e., reducing the costs of the coarse propagation in each correction iteration (3) from  $N C_{\mathcal{G}}$  to  $C_{\mathcal{G}}$ . The estimate of the theoretical speedup of the pipelined PARAREAL using  $K$  iterations is then given by [29]

$$\mathcal{S}(N) = \frac{N C_{\mathcal{F}}}{N C_{\mathcal{G}} + K(C_{\mathcal{F}} + C_{\mathcal{G}})} = \frac{1}{\left(1 + \frac{K}{N}\right) \frac{C_{\mathcal{G}}}{C_{\mathcal{F}}} + \frac{K}{N}}. \quad (6)$$

The projected parallel speedup (6) has been derived by neglecting the time spent communicating between each time slice, later referred to as communication in time. For an increased accuracy of the performance model, we have decided to include this cost in the analysis and to propose a modification of pipelined PARAREAL, which slightly reduces the cost induced by the communications in time. Figure 1a sketches the execution diagram of pipelined PARAREAL. Since the prediction step (2) is sequential,  $N - 1$  communications between time slices are required during this phase (see the thin rectangles on the left of Fig. 1a). To avoid this offset, we have considered an implementation shown in Fig. 1b, in which the solution of the prediction step is computed *concurrently* on each time slice. This removes unnecessary communications between time slices in the prediction step at the cost of redundancy in the computation. Let  $C_T^t$  denote the cost to communicate a single global (fine) solution from a time slice to the next. Hence, the total cost spent in com-



**Fig. 1** Execution diagram of PARAREAL with pipelining (top) and with improved pipelining (bottom). Three iterations of PARAREAL ( $K = 3$ ) on four time slices ( $N = 4$ ) are considered. The end of each Parareal iteration is indicated by a dotted line. Time-slices are numbered from 0 to 3, each line corresponds to one process of PARAREAL

munications in time for  $K$  iterations of PARAREAL is then given by

$$C_{T,K} = C_T^l \sum_{k=1}^K (N - k) = C_T^l \frac{K(2N - K - 1)}{2}. \tag{7}$$

Furthermore, the speedup of our implementation of pipelined PARAREAL can be obtained as

$$S_T(N) = \frac{1}{\left(1 + \frac{K}{N}\right) \frac{C_G}{C_F} + \frac{K}{N} + \frac{C_{T,K}}{N C_F}}. \tag{8}$$

Let denote by  $C_{\widehat{G}}$  the cost of integrating over a given time slice using  $\widehat{G}^{\Delta t}$ . A straightforward adaptation of (8) to the case of pipelined PARAREAL with spatial coarsening using  $\widehat{K}$  iterations yields the following projected speedup

$$\widehat{S}_T(N) = \frac{1}{\left(1 + \frac{\widehat{K}}{N}\right) \frac{C_{\widehat{G}} + C_{\widehat{R}} + C_{\widehat{I}}}{C_F} + \frac{\widehat{K}}{N} + \frac{C_{T,\widehat{K}}}{N C_F}}, \tag{9}$$

where  $C_{\widehat{R}}$  ( $C_{\widehat{I}}$ ) represents the cost of application of the restriction (interpolation) operator to a vector of appropriate dimension, respectively. Neglecting the communications in time is thus only reasonable if

$$\frac{C_{T,\widehat{K}}}{N C_F} \ll \left(1 + \frac{\widehat{K}}{N}\right) \frac{C_{\widehat{G}} + C_{\widehat{R}} + C_{\widehat{I}}}{C_F} + \frac{\widehat{K}}{N}, \tag{10}$$

a condition later discussed in Sect. 4.7.

Finally, we deduce the space–time parallel speedup  $\mathcal{S}_{S,T}(N_{proc})$  as

$$\mathcal{S}_{S,T}(N_{proc}) = \mathcal{S}_S(N_s) \widehat{S}_T(N), \tag{11}$$

where  $\mathcal{S}_S(N_s)$  is the speedup brought by the parallelization in space of the fine solver  $\mathcal{F}$  on  $N_s$  processes, i.e.,

$$\mathcal{S}_S(N_s) = \frac{T_{\mathcal{F},serial}^s}{T_{\mathcal{F},N_s}^s}. \tag{12}$$

Hence, we deduce that the space–time parallelization is only viable if

$$\mathcal{S}_{S,T}(N_{proc}) > \mathcal{S}_S(N_{proc}), \tag{13}$$

a condition later analysed in Sec. 4.2.

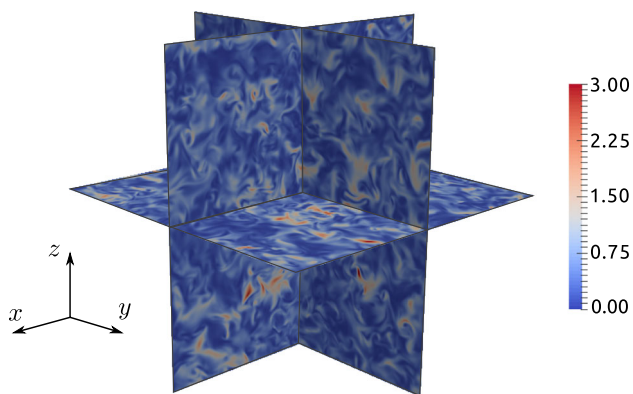
We will rely on the expected speedup of the space–time parallel method  $\mathcal{S}_{S,T}(N_{proc})$  and on the parallel efficiency<sup>1</sup> to predict the parallel performance of our model in Sec. 4.2. We next describe our case study and give details on both the fine and coarse solvers.

### 3 Description of the turbulent flow problem and the CFD solver

We are interested in the simulation of complex turbulent flows and focus on a canonical but relevant test case, well known from the CFD community. The decay of homogeneous isotropic turbulence (HIT) has been studied by many authors from moderate [33,44] to very large scale problems [22]. This case may be seen as the simplest configuration which incorporates fundamental and relevant turbulence mechanisms [31]. In the context of Direct Numerical Simulation (DNS), the problem size is essentially driven by the range of length scales to resolve, which sets the required grid resolution. Specifically, increasing the Reynolds number leads to a wider spectrum of length scales to resolve, but also improves the relevance of the simulation regarding high Reynolds, industrial configurations. In this setup, synthetically generated turbulence decays under the action of dissipation mechanisms. A description of the flow field during the simulation is given in Fig. 2 in terms of turbulent kinetic energy, which highlights the evolution of energy-carrying turbulent eddies and the strong nonlinearity of turbulent flows.

Regarding parallel in time integration algorithms, interesting features of turbulent flows are (i) their chaotic nature (ii) their strongly unsteady behavior, which typically requires explicit integration [7]. Interestingly, the decay of HIT incorporates these issues at a moderate cost, making this model problem a challenging candidate for time-parallel algorithms.

<sup>1</sup> We define the parallel efficiency as the ratio of the parallel speedup over the number of processes involved.



**Fig. 2** Plane cuts of local turbulent kinetic energy  $k_{e,loc}/k_{e,0}$  for  $Re_{\lambda,0} = 322$  ( $k_{e,loc} = \frac{u \cdot u}{2}$ )

### 3.1 Governing equations

The compressible Navier–Stokes equations read:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{14}$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u} + p \boldsymbol{\delta}) = \nabla \cdot \boldsymbol{\tau}, \tag{15}$$

$$\frac{\partial \rho e_T}{\partial t} + \nabla \cdot (\mathbf{u}(\rho e_T + p)) = \nabla \cdot (\mathbf{u} \cdot \boldsymbol{\tau} - \mathbf{q}), \tag{16}$$

where  $\rho$  is the density,  $\mathbf{u} = (u, v, w)$  the velocity vector,  $p$  the pressure,  $\boldsymbol{\delta}$  the unity tensor,  $e_T = e + \frac{u \cdot u}{2}$  the total energy,  $e$  the specific internal energy,  $\boldsymbol{\tau}$  the viscous stress tensor and  $\mathbf{q}$  the heat flux.  $e$ ,  $p$  and  $\rho$  are linked by the equation of state, obtained for a calorically perfect gas:

$$p = (\gamma - 1)\rho e, \tag{17}$$

with  $\gamma$  the specific heat coefficient ratio, whereas  $\boldsymbol{\tau}$  and  $\mathbf{q}$  are modeled via classical Stokes’ hypothesis:

$$\boldsymbol{\tau} = 2\mu \mathbf{S} - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\boldsymbol{\delta}, \tag{18}$$

$$\mathbf{q} = -k_c \nabla T, \tag{19}$$

with  $\mathbf{S} = (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)/2$  the strain-rate tensor,  $\mu$  the dynamic viscosity,  $k_c$  the thermal conductivity, the temperature  $T = (\gamma - 1)e/R$  and  $R$  the gas constant. Temperature dependency for  $\mu$  is accounted for through a power-law assumption:

$$\mu/\mu_{ref} = (T/T_{ref})^{3/4}, \tag{20}$$

and the ratio between  $\mu$  and  $k_c$  is set constant through a constant Prandtl number, which closes the set of equations.

### 3.2 Decay of homogeneous isotropic turbulence

The decay of homogeneous isotropic turbulence is studied using a uniform grid on a 3D periodical box of size  $L$ . Note that all the flow quantities discussed in this section are time dependent unless otherwise specified. In particular the 0 subscript refers to the initial state at  $t = 0$ . Because the problem is homogeneous, the statistical evolution of HIT decay can be reduced to a temporal evolution using the following spatial averaging:

$$\langle f \rangle = \frac{1}{L^3} \iiint_V f(x, y, z) \, dx dy dz. \tag{21}$$

Initial conditions may be obtained by setting the flow fields  $(\rho_0, \mathbf{u}_0, T_0)$ , and parameters  $R, \gamma, Pr, \mu_{ref}$  and  $T_{ref}$ . These fields are constructed through a random process that builds a synthetic turbulent flow field following a prescribed energy spectrum, mostly concentrated at large scales. The detailed methodology associated with the construction of the initial turbulent field is described in Sect. 3.5 and Ap. A of [21], respectively. This procedure sets the flow conditions, essentially characterized by two non-dimensional, time evolving quantities which drive the turbulence evolution: the Taylor scale Reynolds number,<sup>2</sup>

$$Re_\lambda = \frac{\langle \rho \rangle u' \lambda}{\langle \mu \rangle}, \tag{22}$$

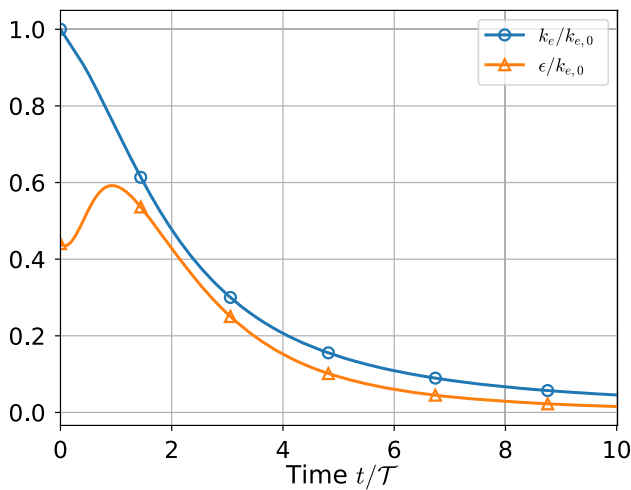
and the turbulent Mach number:

$$M_t = \sqrt{3} \frac{u'}{c}, \tag{23}$$

where  $u' = \sqrt{(\langle uu \rangle + \langle vv \rangle + \langle ww \rangle)/3}$ , and  $c = \sqrt{\gamma RT}$  is the speed of sound.  $\lambda$  is the (transverse) Taylor micro-scale, and provides a length scale associated with intermediary-sized eddies of the flow field.<sup>3</sup>  $Re_\lambda$  sets the width of the turbulent kinetic energy spectrum, and  $M_t$  indicates the influence of compressibility effects, that may trigger discontinuities (e.g., shocklets) in the solution for high  $M_t$ . The turbulent kinetic energy  $k_e$  and the dissipation  $\epsilon$ , are defined as follows:

<sup>2</sup> Since there is no mean flow in this configuration, we cannot use the classical definition of the Reynolds number. We rather use  $Re_\lambda$  (see e.g., [11,44]), which characterizes the ratio of turbulent eddy viscosity, over the fluid viscosity. This Reynolds number, at least one order of magnitude lower than the classical one, has therefore no relevance regarding laminar to turbulent transition, but instead informs about the range of length scales present in an already turbulent flow (see e.g., [31] for a complete description).

<sup>3</sup> For isotropic turbulence  $\lambda$  is conveniently expressed as  $\lambda = u' \sqrt{15 \langle \mu \rangle / (\langle \rho \rangle \epsilon)}$  (see, e.g., [31, Sec. 6.3, p.199]).



**Fig. 3** Time-evolution of  $k_e/k_{e,0}$  and  $\epsilon/k_{e,0}$ , with  $T = (\lambda/u')_{t=0}$ ,  $N_L = 80$ ,  $Re_{\lambda_0} = 46$

$$k_e = \frac{3}{2}u'^2, \quad \epsilon = \left\langle \frac{\mu}{\rho} S.S \right\rangle. \quad (24)$$

Because the footprint of  $k_e$  and  $\epsilon$  is mostly located at large and small scales, respectively, these two quantities are interesting indicators to track the turbulence behavior during the HIT simulation. We can derive a transport equation for  $k_e$  which reduces, using the 3D homogeneous hypothesis and thus isotropy, to:

$$\frac{dk_e}{dt} = -\epsilon. \quad (25)$$

It has been observed [31] that, after a short transient phase, both values exhibit a power-law decay, as shown in Fig. 3.

Physically, it illustrates the energy cascade of the turbulence: energy from the large scale eddies is continuously transferred to the small scale eddies until the latter dissipate through molecular viscosity. The Kolmogorov length scale characterizes the smallest, *dissipative* scales:

$$\eta = \left( \frac{\langle \mu \rangle^3}{\langle \rho \rangle^3 \epsilon} \right)^{1/4}, \quad (26)$$

which sets the smallest grid cell size to achieve *direct* numerical simulation. Smallest length scales must be accurately resolved on the spatial grid. Considering the largest wavenumber that can be represented on a given spatial uniform Cartesian mesh of size  $N_L^3$  (see [31, Chap. 9]):

$$\kappa_{\max}(N_L) = \frac{\pi N_L}{L}, \quad (27)$$

we can derive a condition consistent with the correct resolution of eddies of size  $\eta$ :

$$\forall t, \quad \eta \kappa_{\max}(N_L) \geq \alpha, \quad (28)$$

where  $\alpha$  is a coefficient dependent of the solver and specifically the space discretization scheme.

### 3.3 Massively parallel Navier Stokes solver HYBRID

We use the compressible structured solver HYBRID, developed in C++ by [24], which aims at studying fundamental turbulence problems, such as shock-turbulence interaction [25]. In the absence of discontinuities, the code uses a centered 6th-order finite difference scheme, while time-integration is performed thanks to a 4th-order explicit Runge–Kutta method (RK4). Full details on space and time discretizations are given in [21]. The HYBRID solver uses MPI-based parallelism which relies on space decomposition. It shows very good weak scaling results when using up to 2 million cores [4]. The adopted numerical methodology developed in HYBRID (structured mesh, explicit space–time discretization) also exhibits excellent strong scaling properties, if the number of cells  $N_{cpc}^{min}$  of the spatial subdomain associated with one MPI process (and thus a single CPU core) exceeds a few hundreds. The exact value of  $N_{cpc}^{min}$  to maintain parallel efficiency is necessarily architecture dependent, and sets the *lower bound* of the operating limit for a sole and efficient space parallelization. Given its moderate efficiency, time parallelization should be used to extend the number of usable core for efficient computations, thus keeping  $N_{cpc} > N_{cpc}^{min}$ .

Once the simulation is properly initialized, an appropriate choice of the time step needs to be considered to minimize the computational cost of a *direct* numerical simulation. Although implicit time stepping is usually not rewarding because of the limited accuracy observed at large time steps [7], smallest physical time scales are still significantly larger than the maximum time step  $\delta t^{CFL}$  authorized by stability considerations. Hence, cost efficient simulations must be carried out with  $\delta t \approx \delta t^{CFL}$ . For this specific solver, we have estimated the *CFL* number as 1.79. This value indeed corresponds to the limit of linear stability of the 6th-order centered finite difference scheme associated with the RK4 time-integration method.

### 3.4 Transfer operators for Parareal with spatial coarsening

We detail next the transfer operators used in the variant of PARAREAL based on spatial coarsening as briefly introduced in Sect. 2.2. Since the space discretization in HYBRID is based on high-order finite difference schemes on a Cartesian structured mesh, geometric coarsening in space (known

as vertex-centered coarsening in multigrid [43]) is adopted to construct the coarse level of the two-grid hierarchy. In this setting, the fine mesh is coarsened by simply considering every other node in the three spatial directions, thus leading to a cell size ratio of 2 in each direction between the coarse and the fine mesh.

As restriction operator  $\widehat{\mathcal{R}}$ , we have considered both the injection and the three-dimensional full-weighting operator [43, Sect. 2.9]. Due to this coarsening in space, the time step on the coarse grid is simply set as

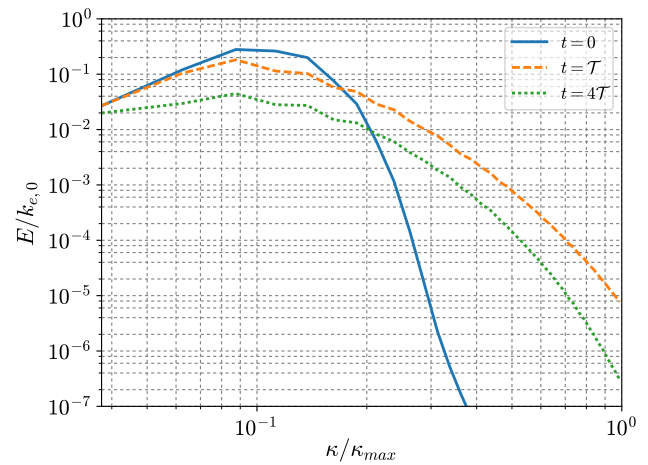
$$\Delta_t = 2\delta_t,$$

to keep the same CFL condition on both fine and coarse grids. Concerning  $\widehat{\mathcal{I}}$ , we have selected various interpolations (trilinear [43, Sec. 2.9], tricubic [26], 7<sup>th</sup>-order or Fourier transformation based [38]). The 7th-order interpolation is based on tensor products of one-dimensional operator, which makes it simultaneously computationally cheap and implementation friendly. The influence of both restriction and interpolation operators on the convergence of PARAREAL with spatial coarsening as well as their costs will be numerically investigated in Sects. 4.4 and 4.7, respectively.

### 3.5 Direct Numerical Simulation methodology

In the following subsection, we develop the methodology to perform the HIT Direct Numerical Simulation, with the objective to propose a *systematic* and *reproducible* framework for these simulations. The Navier–Stokes equations (14), (15) and (16) are solved following a *direct* numerical simulation framework. Thus, the grid requirements need to be adjusted with the minimum length scale  $\eta$ , which sets the ratio between the largest scales (i.e., the box size  $L$ ) and the smallest scales. Recent DNS of HIT with spectral space discretization used  $\alpha = 1$  [22]. Taking into account that HYBRID uses high-order centered space discretization methods, we choose a more restrictive  $\alpha = 1.5$  criterion in (28) for the simulation with  $\mathcal{F}$ , consistent with modified wave numbers expression for 6th-order schemes in space (e.g., [27]). This resolution criterion needs to be satisfied at all times during the decay of turbulence. As the flow field undergoes a transient during which it evolves from synthetic to physical turbulence, with a characteristic “eddy turnover” time scale  $k_e/\epsilon$ , we specifically enforce this criterion after a transient  $t_\phi$ , i.e., using  $\eta = \eta(t_\phi)$  in (28).

In order to propose an estimation of  $t_\phi$ , we first extract the spectral density of energy  $E(\kappa)$  of the turbulent flow field. It represents the energy content associated with the norm  $\kappa$  of the related wave vectors, and is interpreted as so called *turbulent eddies* of corresponding length scale in the study of turbulent flows. Hence,



**Fig. 4** Energy spectra during the decay of HIT, with  $\mathcal{T} = (\lambda/u')_{t=0}$ ,  $N_L = 80$ ,  $Re_{\lambda_0} = 46$

$$k_e = \sum_{\kappa} E(\kappa). \tag{29}$$

Details on the computation of  $E(\kappa)$  for HIT are provided in “Appendix A”.

A representation of energy spectra is given in Fig. 4 at various snapshots of the HIT decay. The spectrum of the synthetic initial solution has an energy peak located at the large scales, consistent with the initialization process. After some time, the turbulent energy cascade takes place, which results in  $k_e$  being spread over every length scales of the flow field.

We estimate the end of this transient  $t_\phi$  as the time for which the spectral density of energy starts decaying at *all* scales:

$$t_\phi = \min \{ t > 0 \mid \forall \kappa, \delta E(\kappa) < 0 \}, \tag{30}$$

where  $\delta E(\kappa)$  is the growth rate of  $E(\kappa)$ . This criterion is useful in finding a systematic condition to determine the maximum  $Re_{\lambda_0}$  that can be simulated using a mesh of given size  $N_L^3$ . We choose to define it as the maximum value of  $Re_{\lambda_0}$  that satisfies condition (28) at  $t_\phi$ , that is:

$$Re_{\lambda_0, \max} = \max \{ Re_{\lambda_0} \mid \forall t > t_\phi, \eta \kappa_{\max}(N_L) \geq \alpha \}. \tag{31}$$

A trial and error procedure is therefore required to obtain the desired resolved state at time  $t_\phi$  and results in a well identified initial state, fully characterized by  $Re_{\lambda_0}$ .

To complete the methodology, an error metric is defined to estimate the accuracy of the parallel time integration. As a basis in the application of PARAREAL to turbulent flows we analyse the physical relevance of the solution rather than the strict convergence of the algorithm. Because the complex flow dynamics of a turbulent flow is characterized in a statistical sense by a transfer from the large to the small

scales, the quality of the solution is evaluated through the spectral density of energy. It may reveal undesired effects such as nonphysical dissipation or dispersion promoted by the numerical scheme. The accuracy of the solution is measured using an error metric developed in Sect. 4.3, which relies on the energy content of the flow field and characterizes the ability of PARAREAL to capture the flow dynamics.

## 4 Numerical results

This section provides a detailed numerical study of the different factors influencing the parallel performance of PARAREAL with spatial coarsening, when considering our test case in Computational Fluid Dynamics. In Sect. 4.2, we compare the pure spatial and space–time parallelizations in terms of speedup and efficiency. This allows us to determine situations for which space–time parallelization is meaningful. Then, in this setting, we examine the numerical quality of the solutions provided by PARAREAL in Sect. 4.3. In Sects. 4.4, 4.5 and 4.6 we investigate the influence of several parameters on the quality of the solution. Finally, in Sect. 4.7, we continue investigating the parallel efficiency of PARAREAL with spatial coarsening, by including in the analysis the costs of the transfer operators and of the time communications.

### 4.1 Methodology

The numerical simulations were performed on the EOS supercomputer at CALMIP, Toulouse, France. This platform is equipped with 612 compute nodes, each node hosting two 10 core Intel Ivy Bridge chips (that run at 2.8 GHz), and 64 GB of system memory. Each socket was equipped with 25 MB of cache memory. EOS nodes are connected by a nonblocking fat tree network, with a network bandwidth of 6.89 Gbytes/s for sending and receiving data. The code was compiled using the Intel 14.0.2.144 compiler, with Intel MPI 4.1.3.049 library.

As pointed out in Sect. 2.3, the parallel performance of PARAREAL is simulated. More precisely, the computations related to all time slices in PARAREAL are performed sequentially in time, whereas the applications of both  $\mathcal{F}$  and  $\mathcal{G}$  through the HYBRID code are parallel in space. To estimate the cost to communicate a single global solution  $C_{\mathcal{T}}^t$  in PARAREAL, we have adopted the following simple strategy. First, we recreate exactly the same spatial decomposition on  $N_{proc}/N$  processes as in HYBRID. Secondly, we perform standard MPI communications of all the relevant physical fields from one time slice to the next over the  $N$  processes and measure the corresponding wall-clock times on the EOS supercomputer. This procedure is then repeated 20 times to provide a meaningful estimation,  $C_{\mathcal{T}}^t$  being then obtained as the maximum wall-clock time. We refer the reader to the

C++ source code<sup>4</sup> for further details. A similar methodology is applied to obtain estimates of  $C_{\mathcal{F}}$  and  $C_{\hat{\mathcal{G}}}$ .

In the following sections, unless stated otherwise, we consider the direct numerical simulation of the decay of homogenous isotropic turbulence at  $Re_\lambda = 46$  on a Cartesian mesh of size  $80^3$  (i.e.,  $N_L = 80$ ). We have estimated the initial CFL number as 1.79 for each simulation. Due to the energy decay, the CFL number is found to slightly decrease as time increases.

### 4.2 Preliminary analysis of the Parareal parallel performance

We address the question of the parallel performance of PARAREAL based on spatial coarsening. An important point is to determine whether space–time parallelization can be more appropriate than plain spatial parallelization in our application. To this end, we operate the solver with  $N_{cpc} < N_{cpc}^{min}$  to deliberately reach the limit of the spatial decomposition. Given  $\mathcal{F}$  and  $\hat{\mathcal{G}}$  introduced in Sect. 3.3, we consider the influence of both  $N$  and  $\hat{K}$  (number of time slices and number of iterations, respectively) on the speedup of PARAREAL with spatial coarsening. For ease of exposition, we first neglect the costs related to both the transfer operators ( $C_{\hat{\mathcal{R}}}$  and  $C_{\hat{\mathcal{T}}}$ ) and the communications in time ( $C_{\mathcal{T},\hat{K}}$ ) in (9). A refined analysis will be proposed later in Sec. 4.7.

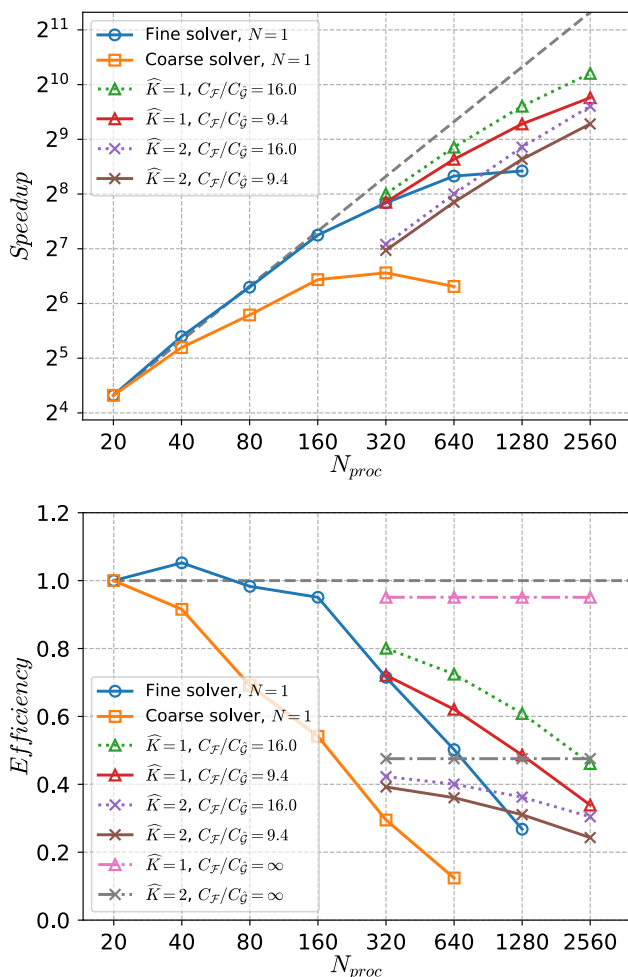
The strong scalability of  $\mathcal{F}$  and  $\hat{\mathcal{G}}$  is shown in Fig. 5(top). The fine solver  $\mathcal{F}$  exhibits good scalability properties for a number of processes up to 160. Nevertheless, for a larger number of processes, the performance starts to saturate. Not surprisingly, we also observe a rather quick deterioration of the parallel performance of the coarse solver  $\hat{\mathcal{G}}$ . It is important to remark at this stage that this earlier decrease in performance of the coarse solver will bound the cost ratio  $C_{\mathcal{F}}/C_{\hat{\mathcal{G}}}$  at a lower value than the expected one<sup>5</sup> (which is equal to 16, assuming a perfect strong scaling and excluding communication costs).

Given a number of processes in space fixed to  $N_{proc}/N = 160$ , we consider the additional parallelization in time with 2, 4, 8 and 16 time slices, respectively. The simulated speedups and parallel efficiencies obtained after one or two iterations of PARAREAL with spatial coarsening are then provided in the Fig. 5. The inherent limits of PARAREAL are indicated in Fig. 5 (bottom). They are obtained by considering an infinite cost ratio  $C_{\mathcal{F}}/C_{\hat{\mathcal{G}}}$ . Performing a single iteration of PARAREAL with spatial coarsening does lead to an increased speedup and parallel efficiency with respect to the pure spatial parallelization, whatever the number of

<sup>4</sup> available at <https://gitlab.com/tlunet/parallel-in-time>.

<sup>5</sup> This value of the theoretical ratio of coarse-to-fine execution is explained by the coarsening in space by a factor of 2 in each direction and by the choice of the coarse time step detailed in Sect. 3.4.





**Fig. 5** Simulated speedup (top) and parallel efficiency (bottom) of HYBRID (fine and coarse solvers), and of PARAREAL with spatial coarsening after  $\widehat{K} = 1, 2$  iterations

time slices. This is a rather satisfactory result. However, for two or more iterations of PARAREAL with spatial coarsening, the success is limited. Indeed, for  $N_{proc} = 320$  or  $N_{proc} = 640$ , the speedup of the space–time parallel algorithm is found to be lower than the speedup obtained with exclusive space parallelization. Not surprisingly, the parallel efficiency is maximized if the number of PARAREAL iterations is limited. Interestingly, the gap of parallel efficiency between  $\widehat{K} = 1$  and  $\widehat{K} = 2$  reduces as the number of time slices  $N$  is increased. This trend is explained because the reduced efficiency of the coarse solver is balanced by a larger number of fine solver computations as  $\widehat{K}$  increases. However, because the considered efficiency levels are rather low, we mainly favor the application of a *single* iteration of PARAREAL with spatial coarsening in the rest of this study.<sup>6</sup>

<sup>6</sup> We refer the reader to only Sec. 4.3 for numerical experiments using two iterations ( $\widehat{K} = 2$ ) of Parareal with spatial coarsening.

**Table 1** Relative gain of the space–time parallelization  $\sigma_{S,T}^*$  after one iteration of PARAREAL with spatial coarsening (neglecting the costs of communication in time and transfer operators)

$N_{proc}$	160	320	640	1280
$N_{proc}/N = 80$	– 18%	– 3.9%	+ 10%	+ 49%
$N_{proc}/N = 160$		+ 0.82%	+ <b>24%</b>	+ 82%
$N_{proc}/N = 320$			– 1.1%	+ <b>54%</b>

$N_{proc}$  denotes the total number of processes with  $N_{proc}/N$  processes being assigned to the parallelization in space only. The numbers of time slices considered here are 2, 4, 8 and 16, respectively

We now investigate the influence of the  $N_{proc}/N$  parameter on the performance of the combined space–time parallelization in our application. Hence, we introduce the relative gain of the space–time parallelization as

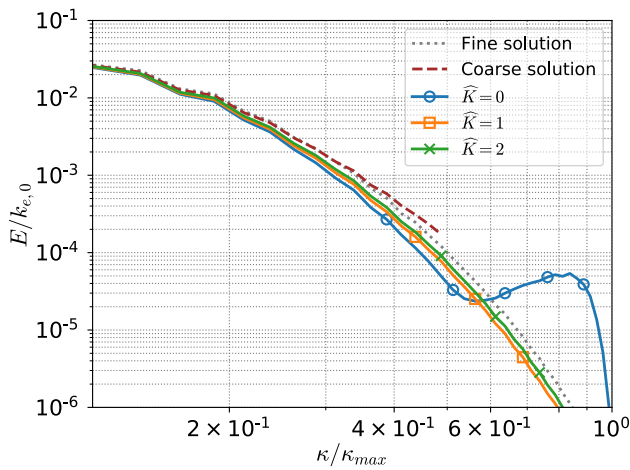
$$\sigma_{S,T} = \frac{\mathcal{S}_{S,T}(N_{proc}) - \mathcal{S}_S(N_{proc})}{\mathcal{S}_S(N_{proc})}. \tag{32}$$

A positive value of  $\sigma_{S,T}$  thus indicates that the space–time parallelization is worth considering. We denote by  $\sigma_{S,T}^*$  the relative gain of the space–time parallelization when neglecting the costs of communication in time and transfer operators. Although  $\sigma_{S,T}^*$  is an ideal value, it is worth noting that  $\sigma_{S,T} \approx \sigma_{S,T}^*$  for a sufficiently long time slice, i.e., if  $C_{\mathcal{T},K}$ ,  $C_{\widehat{\mathcal{T}}}$  and  $C_{\widehat{\mathcal{G}}}$  are negligible compared to  $C_{\mathcal{F}}$  and  $C_{\widehat{\mathcal{G}}}$ .

Table 1 collects the values of  $\sigma_{S,T}^*$  versus  $N_{proc}/N$ , when a single iteration of PARAREAL with spatial coarsening is performed. For a fixed number of processes devoted to the spatial parallelization, increasing the number of time slices ( $N$ ) does improve  $\sigma_{S,T}^*$  as expected. This behavior is indeed in agreement with Fig. 5(top). Table 1 also reveals that the parallelization in space and in time is worth considering when the total number of processes is large. In this situation, we rather favor the case of a low number of time slices (see the bold values in Table 1), since the convergence of PARAREAL is reached in at most  $N$  iterations. Hence, in what follows, we choose  $N = 4$  and study the quality of the solution of the first iterations of PARAREAL on a large number of processors ( $N_{proc} = 640$  or  $N_{proc} = 1280$ ). This is next investigated in Sect. 4.3.

### 4.3 Energy spectra after successive iterations of Parareal with spatial coarsening

We here analyze the flow solution obtained after successive iterations of PARAREAL with spatial coarsening. In particular, we focus on the energy spectrum of the resulting turbulent flow. As described above, we use trilinear interpolation and injection as transfer operators, and  $N_{\Delta_t, \widehat{\mathcal{G}}} = 20$ . The simulation of the HIT decay is solved between physical times  $t_\phi = 1.4\mathcal{T}$  and  $t_{end} = 4.4\mathcal{T}$ , which corresponds to the



**Fig. 6** Energy spectra after  $\widehat{K}$  iterations of PARAREAL with spatial coarsening at  $t_{end}$  ( $N = 4$ ,  $N_{\Delta t, \widehat{G}} = 20$ , trilinear interpolation and injection as transfer operators)

relevant part of the energy decay (see Fig. 3). The energy spectrum is represented on Fig. 6 for  $\widehat{G}$  (coarse integrator without interpolation),  $\mathcal{F}$  (fine integrator, reference solution) and the first iterations of PARAREAL with spatial coarsening, respectively.

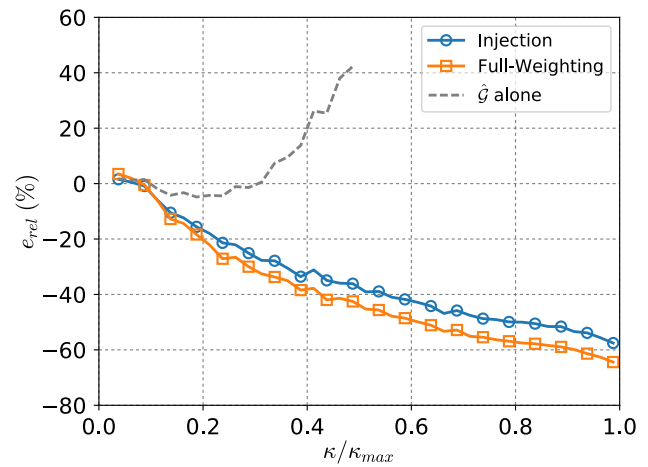
Because of the insufficient spatial resolution for  $\widehat{G}$ , an over-estimation of the energy on the small scales around  $\kappa = 0.5\kappa_{max}$ . This is a classical observation of unresolved direct simulation that results from a partial cut off of the physical dissipation mechanisms operating at the smallest scales. The effects of the interpolation are directly highlighted in the case  $\widehat{K} = 0$  for which the middle range scales are significantly diffused, while a peak of energy is observed for the smallest scales. The first iteration ( $\widehat{K} = 1$ ) results in a dampening of this high frequency peak, along with a significant improvement of the middle range scales. Interestingly the second iteration ( $\widehat{K} = 2$ ) does not bring significant improvements, reinforcing the idea of using a single iteration of PARAREAL with spatial coarsening.

To better quantify the quality of the solution, we extract the relative error of the energy spectrum  $e_{rel}$ , defined as:

$$e_{rel}(\kappa) = \frac{\widetilde{E}(\kappa) - E_{\mathcal{F}}(\kappa)}{E_{\mathcal{F}}(\kappa)} \tag{33}$$

where  $\widetilde{E}(\kappa)$  corresponds to the spectrum of an approximate solution, ( $\widehat{G}$  or PARAREAL iteration) and  $E_{\mathcal{F}}(\kappa)$  the spectrum of the reference solution computed with  $\mathcal{F}$  solely. While a negative value for  $e_{rel}(\kappa)$  indicates energy damping, a positive value indicates energy amplification. We add quantification on the spectrum error by looking at the relative error on quantities integrated over all length scales, the energy:

$$e_{k_e} = \frac{\widetilde{k}_e - k_{e,\mathcal{F}}}{k_{e,\mathcal{F}}}, \tag{34}$$



**Fig. 7** Influence of the restriction operator on the relative energy error after one iteration of PARAREAL with spatial coarsening ( $N = 4$ ,  $N_{\Delta t, \widehat{G}} = 20$ , trilinear interpolation)

and the dissipation:

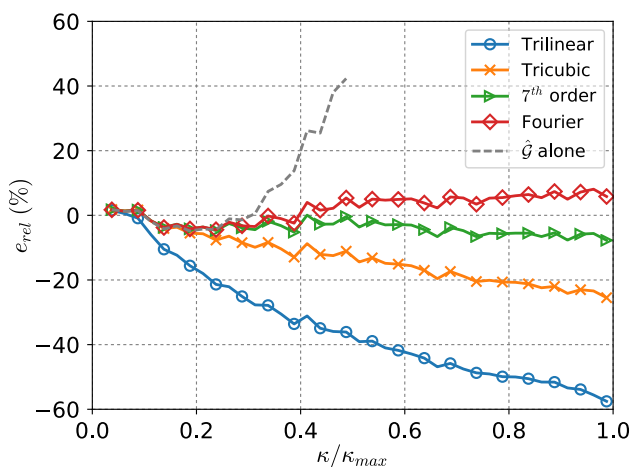
$$e_{\epsilon} = \frac{\widetilde{\epsilon} - \epsilon_{\mathcal{F}}}{\epsilon_{\mathcal{F}}}. \tag{35}$$

These two scalars are representative of statistical errors at two levels:  $e_{k_e}$  characterizes the large length scales, while  $e_{\epsilon}$  is more a measure of the behaviour at small length scales. The three error indicators  $e_{rel}$ ,  $e_{k_e}$  and  $e_{\epsilon}$  will be combined to investigate the influence of several parameters on the quality of the solution: the transfer operators in Sect. 4.4, the Reynolds number in Sect. 4.5 and the time slice length in Sect. 4.6.

### 4.4 Influence of the transfer operators

The influence of the restriction operator on the relative energy error is shown in Fig. 7. In agreement with the developments above, we consider the solution after one iteration of PARAREAL to provide a possible comparison. Whatever the restriction operators, the absolute value of the relative energy error is found to grow in the small scale regime. Moreover, the curves related to the injection and the full-weighting operators look very similar. Hence, we rather consider the injection as restriction operator in the rest of the manuscript, to minimize  $C_{\widehat{\mathcal{R}}}$ .

The influence of the interpolation operator on the relative energy error is shown in Fig. 8. The corresponding relative error indicators on the turbulent kinetic energy and on the dissipation respectively are given in Table 2. Figure 8 reveals that high-order interpolation operators must be rather favored to minimize the relative error in energy. This is also confirmed in Table 2. Hence, in the following, we consider the 7th-order interpolation operator as a standard choice.



**Fig. 8** Influence of the interpolation operator on the relative energy error after one iteration of PARAREAL with spatial coarsening ( $N = 4$ ,  $N_{\Delta_t, \hat{G}} = 20$ , injection as restriction operator)

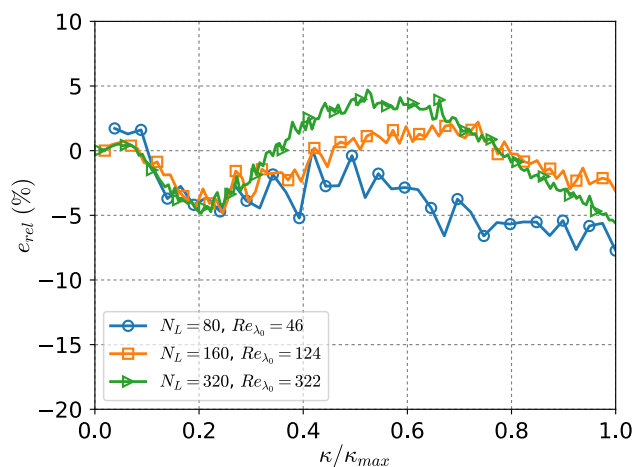
**Table 2** Influence of the interpolation operator on the relative errors on  $k_e$  and  $\epsilon$  after one iteration of PARAREAL with spatial coarsening ( $N = 4$ ,  $N_{\Delta_t, \hat{G}} = 20$ , injection as restriction operator)

Interpolation	$e_{k_e}$ (%)	$e_\epsilon$ (%)
Trilinear	-6.6	-17
Tricubic	-1.5	-5.5
7th-order	-0.85	-2.8
Fourier	-0.79	-2.1
$\hat{G}$ alone	-0.83	-2.7

### 4.5 Influence of the Reynolds number

We consider the simulation of the decay of homogeneous isotropic turbulence on two finer grids ( $160^3$  and  $320^3$ , respectively) associated with larger Reynolds numbers to analyse the influence of the Reynolds number on the quality of the solution. From a physical standpoint, the major effect on the simulation is a widening of the range of length scales carried by the turbulent flow.

The influence of the Reynolds number  $Re_\lambda$  on the relative error on the energy is shown in Fig. 9. The three curves exhibit a very similar behavior at all scales for each considered Reynolds number. The corresponding relative errors on the turbulent kinetic energy and on the dissipation are given in Table 3. The low values of both relative errors indicate that performing only one iteration of PARAREAL is also meaningful when the Reynolds number is increased. This relative independence to the Reynolds number is attributed to the nature of the coarse integrator. The spatial coarsening challenges the ability of PARAREAL to handle correctly the *dissipative* scales of the turbulent flow. The grid size being designed to specifically handle the smallest length scales, this



**Fig. 9** Influence of the Reynolds number  $Re_\lambda$  on the relative energy error  $e_{rel}$  after one iteration of PARAREAL with spatial coarsening ( $N = 4$ ,  $N_{\Delta_t, \hat{G}} = 20$ , injection and 7th-order interpolation as transfer operators)

**Table 3** Influence of the Reynolds number  $Re_\lambda$  on the relative error on  $k_e$  and  $\epsilon$  after one iteration of PARAREAL with spatial coarsening ( $N = 4$ ,  $N_{\Delta_t, \hat{G}} = 20$ , injection and 7th-order interpolation as transfer operators)

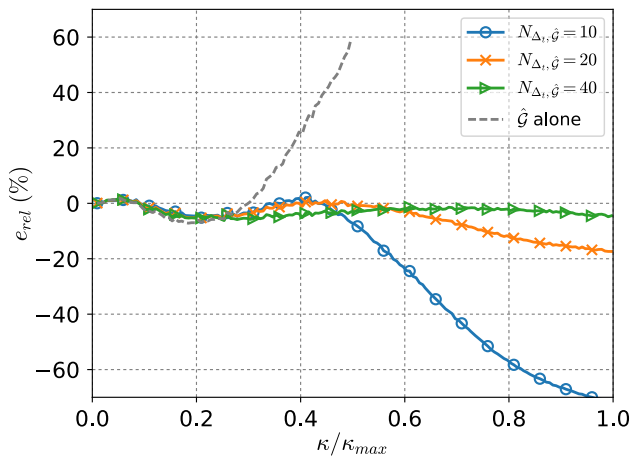
$N_L$	$Re_{\lambda_0}$	$t_\phi/T$	$t_{end}/T$	$Re_{\lambda_\phi}$	$Re_{\lambda_E}$	$e_{k_e}$ (%)	$e_\epsilon$ (%)
80	46	1.4	4.4	26	16	-0.85	-2.8
160	124	2.7	4.1	41	30	-0.68	-2.0
320	322	3.4	4.1	55	46	-0.60	-1.5

$Re_{\lambda_\phi}$  ( $Re_{\lambda_E}$ ) is the Reynolds number at the beginning (end) of the parallel in time simulation

constraint does not evolve significantly relatively to the cell size, hence with an increase of the Reynolds number. These encouraging results will be the subject of a forthcoming study. It suggests similar behavior of PARAREAL in large scale simulations characterized with significantly higher Reynolds numbers (e.g., [22]).

### 4.6 Influence of the time slice length

The increase of the Reynolds number previously developed is also useful to test the influence of the time slice size in the simulation, as investigated in [12, Sec.4], [14, Sec 5.2] and [32, Sec 4.5]. This is indeed a critical parameter regarding the parallel efficiency of the algorithm as mentioned in Sect. 2.3. Indeed, for small Reynolds numbers, the required number of time steps is reduced. Using large time slices is therefore not relevant as the final stage of the simulation falls into a low global energy level, which may be correctly handled by the coarse integrator itself. For large Reynolds numbers instead, a much larger number of time steps is required and PARAREAL can be applied with large time slices, while keeping a



**Fig. 10** Influence of  $N_{\Delta_r, \hat{g}}$  on the relative error on the energy spectrum after one iteration of PARAREAL with spatial coarsening ( $N = 4$ , injection and 7<sup>th</sup>-order interpolation as transfer operators)

global energy level out of reach (regarding *direct* simulation constraints) for the coarse propagator.

We thus investigate the influence of  $N_{\Delta_r, \hat{g}}$  on the energy relative error after one iteration of PARAREAL with spatial coarsening. Since changing this parameter will also modify the physical time  $t_{end}$ , we have decided to restart the time parallel algorithm after its first iteration to complete the full simulation with  $t_{end}$  matching the other choices of  $N_{\Delta_r, \hat{g}}$ . We consider a simulation at Reynolds number  $Re_{\lambda_0} = 322$  with several values of  $N_{\Delta_r, \hat{g}} \in \{10, 20, 40\}$ , which implies restarting PARAREAL with spatial coarsening 3, 1 and 0 times, respectively. HYBRID was ran using the same CFL constraint and a number of time slices of  $N = 4$ , leading to  $t_{end} = 4.8T$ .

The main effects of the variation of  $N_{\Delta_r, \hat{g}}$  on  $e_{rel}$  are represented in Fig. 10. We observe a reduction of the dampening error at the small scales. We attribute this effect to the dissipation of the small scales initially affected by injection/interpolation steps. Increasing  $N_{\Delta_r, \hat{g}}$  does not significantly deteriorate  $e_\epsilon$ , which remains almost independent of  $N_{\Delta_r, \hat{g}}$ , although we observe a slight dampening appearing for the middle range, more energetic length scales ( $\kappa \approx 0.4\kappa_{max}$ ). On the other hand, the latter amplification has an expected detrimental effect on the level of  $e_{k_e}$  which increases with  $N_{\Delta_r, \hat{g}}$ . Results are summarized in Table 4, together with the results obtained with the coarse integrator operated solely as a reference. Note that the apparently small error observed with the coarse integrator benefits from an error compensation with the adopted metric.

#### 4.7 Refined analysis of the speedup of Parareal

Finally, based on (9), we include the costs of both the communications in time ( $C_\tau^t$ ) and the transfer operators ( $C_{\mathcal{I}}$  and  $C_{\mathcal{R}}$ )

**Table 4** Influence of  $N_{\Delta_r, \hat{g}}$  on the relative errors on  $k_e$  and  $\epsilon$  after one iteration of PARAREAL with spatial coarsening ( $Re_{\lambda_0} = 322$ , injection and 7<sup>th</sup>-order interpolation as transfer operators)

$N_{\Delta_r, \hat{g}}$	$e_k$ (%)	$e_\epsilon$ (%)
10	-0.24	-3.3
20	-0.36	-2.7
40	-0.49	-3.4
$\hat{g}$ alone	-0.63	-4.2

**Table 5** Influence of  $C_{\mathcal{T},1}$  and  $C_{\mathcal{I}}$  on the relative gain of the space-time parallelization  $\sigma_{S,T}$  after one iteration of PARAREAL with spatial coarsening ( $\hat{K} = 1$ )

$N_{proc}$	Case	$\sigma_{S,T}$ (%)	$\sigma_{S,T}^*$ (%)
640	(a)	19	24
	(b)	18	24
1280	(a)	47	54
	(b)	46	54

Case (a) includes only  $C_{\mathcal{T},1}$ , whereas case (b) considers both  $C_{\mathcal{T},1}$  and  $C_{\mathcal{I}}$  in relation (32) ( $N = 4$ ,  $N_{\Delta_r, \hat{g}} = 20$ , injection and 7<sup>th</sup>-order interpolation as transfer operators)

in the analysis of the speedup. We consider the configuration at  $Re_{\lambda_0} = 46$  ( $N_L = 80$ ) with the 7th-order interpolation and the injection as transfer operators.

The cost related to the restriction operator can be neglected since no floating operations are involved, i.e.,  $C_{\mathcal{R}} = 0$ . Furthermore, we can relate the cost of the interpolation to the cost of performing one single time-step of the coarse solver. Indeed the one-dimensional formulas for either the 7th-order interpolation or the 6th-order finite difference discretization in HYBRID involve a similar complexity (15 and 11 floating point operations are required, respectively). A more precise estimation of the complete computational complexity<sup>7</sup> then leads to

$$C_{\mathcal{I}} \leq \frac{15}{39} \times \frac{15}{11} \frac{C_{\hat{g}}}{N_{\Delta_r, \hat{g}}}. \quad (36)$$

In the following, we consider the upper bound in (36) as an accurate estimation of  $C_{\mathcal{I}}$ . The relative gain of the space-time parallelization  $\sigma_{S,T}$  (see (32)) is then given in Table 5.

Table 5 reveals that the effective relative gains  $\sigma_{S,T}$  are relatively close to the ideal values  $\sigma_{S,T}^*$  discussed in Sect. 4.2. The effective gain can be indeed significant (especially when considering a large number of processors), which is

<sup>7</sup> In HYBRID the computation of the right-hand side of the Navier-Stokes equations involves 39 gradient evaluations (3 for (14), 18 for (15), 6 for (16), 9 for (18) and 3 for (19), respectively). Since the RK4 time integration method requires 4 evaluations of right-hand sides, the total amount of gradient evaluations for one time step of the coarse solver is then 156. Finally, 60 stencil evaluations per coarse point are required for the interpolation of the 5 different fields.

a satisfactory behavior. Finally, comparing cases (a) and (b) in Table 5 reveals that the influence of  $C_{\mathcal{T}}$  is found to be marginal.

## 5 Conclusions

This manuscript investigates the applicability of PARAREAL with spatial coarsening to a relevant test case in Computational Fluid Dynamics related to the simulation of the decay of homogeneous isotropic turbulence. The time parallel simulation of such flows does actually lead to an instructional test case, since their main interesting features are both their chaotic nature and their strongly unsteady behavior. In a first phase, we have decided to simulate the parallelization in time, whereas the parallelization in space is truly implemented on a distributed memory passing system. Explicit Runge–Kutta time integration methods and high-order finite difference schemes are used for the temporal and spatial discretizations of the Navier–Stokes equations.

A methodology related to the computation of the energy spectrum has been proposed to assess the numerical quality of the iterative solution provided by the time parallel algorithm. Based on this analysis, we have found that the solution after a single iteration of PARAREAL with spatial coarsening was physically relevant, provided that a high-order interpolation operator in space is employed. In this setting, the extensive numerical experiments clearly illustrate the possible benefits of using parallelization in time. This rather encouraging result from a physical point of view needs of course to be confirmed by a detailed convergence study. This is indeed an important research direction that we are currently considering.

We are deeply convinced that this test case can serve as a relevant benchmark for time parallel methods. Hence, to propose a reproducible framework, we have carefully described the complete methodology and plan to make the simulation code freely available to the community, if a significant interest appears. Indeed, an instructive next step would be then to assess the performance of current or emerging time parallel algorithms on this configuration.

**Acknowledgements** The authors thank the two reviewers for their comments. The authors would like to acknowledge CALMIP for the donation of computing hours on the EOS computer. This work was granted access to the HPC resources of CALMIP under allocation P1425 and P17005. Thibaut Lunet would like to thank Daniel Ruprecht, Michael Minion, Debasmitta Samaddar, Jacob B. Schroder, Martin J. Gander, Robert Speck, Andreas Schmidt, Felix Kwok, Beth Wingate and Raymond J. Spiteri for the interesting discussions at the 5th Workshop on Parallel-in-time integration at the Banff International Research Station (BIRS), and the Jülich Supercomputing Center for granting a Student Travel Award for attending this conference.

## A. Computation of the energy spectrum

We briefly describe the computation of the energy spectrum. More details can be found in the literature (see, e.g., [20, Ap. B]). We denote by  $\hat{\mathbf{u}}(\kappa_x, \kappa_y, \kappa_z)$  the Discrete Fourier Transform (DFT) of  $\mathbf{u}(x, y, z)$ , using  $1/(N_L^3)$  to normalize the direct (inverse) DFT, respectively. We define  $\kappa_x, \kappa_y$  and  $\kappa_z$  the discrete one-dimensional wavenumbers as:

$$\kappa_{[\cdot]} = \left\{ n \frac{2\pi}{L} \text{ with } n \in \mathbb{Z} \mid -\frac{N_L}{2} + 1 \leq n \leq \frac{N_L}{2} \right\}, \quad (37)$$

and the associated vector norm:

$$\kappa = \|\boldsymbol{\kappa}\| = \sqrt{\kappa_x^2 + \kappa_y^2 + \kappa_z^2}. \quad (38)$$

The 3D Fourier space is discretized into three-dimensional shells of thickness  $d\kappa$  (usually  $\frac{2\pi d\kappa}{L} = 1$ ). To deduce the energy corresponding to the wavenumber  $\kappa_i$ , a bin count is then performed using an integer number of shells:

$$E(\kappa_i) = \frac{1}{2N_L^6} \sum_{\kappa \in [\kappa_i - \frac{d\kappa}{2}, \kappa_i + \frac{d\kappa}{2}]} |\hat{\mathbf{u}}(\kappa_x, \kappa_y, \kappa_z)|^2. \quad (39)$$

## References

1. Arteaga, A., Ruprecht, D., Krause, R.: A stencil-based implementation of Parareal in the C++ domain specific embedded language STELLA. *Appl. Math. Comp.* **267**, 727–741 (2015)
2. Aubanel, E.: Scheduling of tasks in the Parareal algorithm. *Parallel Comput.* **37**(3), 172–182 (2011)
3. Bal, G.: Analysis of the convergence and of the stability of the Parareal algorithm for partial differential equations. In: Kornhuber, R., et al. (eds.) *Domain Decomposition Methods in Science and Engineering*, Lecture Notes in Computational Science and Engineering, vol. 40, pp. 426–432. Springer, New York (2005)
4. Bermejo-Moreno, I., Bodart, J., Larsson, J., Barney, B., Nichols, J., Jones, S.: Solving the compressible Navier–Stokes equations on up to 1.97 million cores and 4.1 trillion grid points. In: *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*. ACM, p. 62 (2013)
5. Berry, L.A., Elwasif, W., Reynolds-Barredo, J., Samaddar, D., Sanchez, R., Newman, D.E.: Event-based Parareal: a data-flow based implementation of Parareal. *J. Comput. Phys.* **231**(17), 5945–5954 (2012)
6. Celledoni, E., Kvamsdal, T.: Parallelization in time for thermo-viscoplastic problems in extrusion of aluminium. *Int. J. Numer. Methods Eng.* **79**(5), 576–598 (2009)
7. Choi, H., Moin, P.: Effects of the computational time step on numerical solutions of turbulent flow. *J. Comput. Phys.* **113**(1), 1–4 (1994)
8. Croce, R., Ruprecht, D., Krause, R.: Parallel-in-space-and-time simulation of the three-dimensional, unsteady Navier–Stokes equations for incompressible flow. In: Bock, H., Hoang, X., Rannacher, R., Schlöder, J. (eds.) *Modeling, Simulation and Optimization of Complex Processes-HPSC 2012*, pp. 13–23. Springer, New York (2014)

9. Dai, X., Maday, Y.: Stable Parareal in time method for first-and second-order hyperbolic systems. *SIAM J. Sci. Comput.* **35**(1), A52–A78 (2013)
10. Dongarra, J., Hittinger, J., Bell, J., Chacon, L., Falgout, R., Heroux, M., Hovland, P., Ng, E., Webster, C., Wild, S.: Applied mathematics research for exascale computing. Tech. Rep. LLNL-TR-651000, Lawrence Livermore National Laboratory, USA (2014). <https://science.energy.gov/~media/ascr/pdf/research/am/docs/EMWGreport.pdf>
11. Donzis, D.A., Yeung, P.K., Sreenivasan, K.R.: Dissipation and enstrophy in isotropic turbulence: resolution effects and scaling in direct numerical simulations. *Phys. Fluids* **20**, 045108 (2008)
12. Eghbal, A., Gerber, A.G., Aubanel, E.: Acceleration of unsteady hydrodynamic simulations using the Parareal algorithm. *J. Comput. Sci.* **19**, 57–76 (2017)
13. Farhat, C., Cortial, J., Dastillung, C., Bavestrello, H.: Time-parallel implicit integrators for the near real-time prediction of linear structural dynamics responses. *Int. J. Numer. Methods Eng.* **67**(5), 697–724 (2006)
14. Fischer, P.F., Hecht, F., Maday, Y.: A Parareal in time semi-implicit approximation of the Navier–Stokes equations. In: Kornhuber, R., et al. (eds.) *Domain Decomposition Methods in Science and Engineering*. Lecture Notes in Computational Science and Engineering, vol. 40, pp. 433–440. Springer, New York (2005)
15. Gander, M., Petcu, M.: Analysis of a Krylov subspace enhanced Parareal algorithm for linear problem. In: *ESAIM: Proceedings*, vol. 25, pp. 114–129 (2009)
16. Gander, M.J.: 50 years of time parallel time integration. In: Carraro, T., Geiger, M., Körkel, S., Rannacher, R. (eds.) *Multiple Shooting and Time Domain Decomposition Methods*, pp. 69–114. Springer, New York (2015)
17. Gander, M., Hairer, E.: Nonlinear convergence analysis for the Parareal algorithm. In: Widlund, O.B., Keyes, D.E. (eds.) *Domain Decomposition Methods in Science and Engineering*, Lecture Notes in Computational Science and Engineering, vol. 60, pp. 45–56. Springer, New York (2008)
18. Gander, M.J., Vandewalle, S.: Analysis of the Parareal time-parallel time-integration method. *SIAM J. Sci. Comput.* **29**(2), 556–578 (2007)
19. Gropp, W.: *MPI: The Complete Reference*, vol. 2, The MPI-2 Extensions. MIT Press, Cambridge (1998)
20. Jamme, S.: Étude de l’interaction entre une turbulence homogène isotrope et une onde de choc. Ph.D. thesis, Institut National Polytechnique de Toulouse, France (1998)
21. Johnsen, E., Larsson, J., Bhagatwala, A.V., Cabot, W.H., Moin, P., Olson, B.J., Rawat, P.S., Shankar, S.K., Sjögreen, B., Yee, H., et al.: Assessment of high-resolution methods for numerical simulations of compressible turbulence with shock waves. *J. Comput. Phys.* **229**(4), 1213–1237 (2010)
22. Kaneda, Y., Ishihara, T., Yokokawa, M., Itakura, K., Uno, A.: Energy dissipation rate and energy spectrum in high resolution direct numerical simulations of turbulence in a periodic box. *Phys. Fluids* **15**(2), L21–L24 (2003)
23. Kreienbuehl, A., Naegel, A., Ruprecht, D., Speck, R., Wittum, G., Krause, R.: Numerical simulation of skin transport using Parareal. *Comput. Vis. Sci.* **17**(2), 99–108 (2015)
24. Larsson, J., Lele, S., Moin, P.: Effect of numerical dissipation on the predicted spectra for compressible turbulence. *CTR Annual Research Briefs* pp. 47–57 (2007)
25. Larsson, J., Bermejo-Moreno, I., Lele, S.K.: Reynolds-and Mach-number effects in canonical shock-turbulence interaction. *J. Fluid Mech.* **717**, 293 (2013)
26. Lekien, F., Marsden, J.: Tricubic interpolation in three dimensions. *Int. J. Numer. Methods Eng.* **63**(3), 455–471 (2005)
27. Lele, S.: Compact finite difference schemes with spectral-like resolution. *J. Comput. Phys.* **103**(1), 16–42 (1992)
28. Lions, J.L., Maday, Y., Turinici, G.: A “Parareal” in time discretization of PDE’s. *C. R. Math. Acad. Sci. Paris* **332**(7), 661–668 (2001)
29. Minion, M.: A hybrid Parareal spectral deferred corrections method. *Comm. App. Math. Comp. Sci.* **5**(2), 265–301 (2010)
30. Mula, O.: Some contributions towards the parallel simulation of time dependent neutron transport and the integration of observed data in real time. Ph.D. thesis, University of Paris VI, France (2014)
31. Pope, S.B.: *Turbulent Flows*. Cambridge University Press, Cambridge (2000)
32. Reynolds-Barredo, J.M., Newman, D.E., Sanchez, R., Samaddar, D., Berry, L.A., Elwasif, W.R.: Mechanisms for the convergence of time-parallelized, Parareal turbulent plasma simulations. *J. Comput. Phys.* **231**(23), 7851–7867 (2012)
33. Rogallo, R.S.: Numerical experiments in homogeneous turbulence. Tech. rep., NASA Ames Research Center, NASA Technical Memorandum 81315 (1981)
34. Ruprecht, D.: A shared memory implementation of pipelined Parareal. arXiv preprint [arXiv:1509.06935](https://arxiv.org/abs/1509.06935) (2015)
35. Ruprecht, D.: Wave propagation characteristics of Parareal (2017). [arXiv:1701.01359](https://arxiv.org/abs/1701.01359) [math.NA]
36. Ruprecht, D.: Convergence of Parareal with spatial coarsening. *PAMM* **14**(1), 1031–1034 (2014)
37. Ruprecht, D., Krause, R.: Explicit parallel-in-time integration of a linear acoustic-advection system. *Comput. Fluids* **59**, 72–83 (2012)
38. Russell, F.P., Wilkinson, K.A., Kelly, P.H., Skylaris, C.K.: Optimised three-dimensional Fourier interpolation: an analysis of techniques and application to a linear-scaling density functional theory code. *Comput. Phys. Commun.* **187**, 8–19 (2015)
39. Samaddar, D., Newman, D.E., Sánchez, R.: Parallelization in time of numerical simulations of fully-developed plasma turbulence using the Parareal algorithm. *J. Comput. Phys.* **229**(18), 6558–6573 (2010)
40. Schiesser, W.E.: *The Numerical Method of Lines: Integration of Partial Differential Equations*. Elsevier, London (2012)
41. Staff, G.A., Rønquist, E.M.: Stability of the Parareal algorithm. In: Kornhuber, R., et al. (eds.) *Domain Decomposition Methods in Science and Engineering*, Lecture Notes in Computational Science and Engineering, vol. 40, pp. 449–456. Springer, New York (2005)
42. Steiner, J., Ruprecht, D., Speck, R., Krause, R.: Convergence of Parareal for the Navier–Stokes equations depending on the Reynolds number. In: Abdulle, A., Deparis, S., Kressner, D., Nobile, E., Picasso, M. (eds.) *Numerical Mathematics and Advanced Applications—ENUMATH 2013*, vol. 103, pp. 195–202. Springer, New York (2015)
43. Trottenberg, U., Oosterlee, C.W., Schüller, A.: *Multigrid*. Academic, New York (2001)
44. Vincent, A., Meneguzzi, M.: The spatial structure and statistical properties of homogeneous turbulence. *J. Fluid Mech.* **225**, 1–20 (1991)
45. Wang, Q., Gomez, S., Blonigan, P., Gregory, A., Qian, E.: Towards scalable parallel-in-time turbulent flow simulations. *Phys. Fluids* **25**(11), 10818 (2013)