Technique for forcing high Reynolds number isotropic turbulence in physical space

John A. Palmore Jr.* and Olivier Desjardins†
Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, New York 14853, USA

(Received 29 July 2017; published 16 March 2018)

Many common engineering problems involve the study of turbulence interaction with other physical processes. For many such physical processes, solutions are expressed most naturally in physical space, necessitating the use of physical space solutions. For simulating isotropic turbulence in physical space, linear forcing is a commonly used strategy because it produces realistic turbulence in an easy-to-implement formulation. However, the method resolves a smaller range of scales on the same mesh than spectral forcing. We propose an alternative approach for turbulence forcing in physical space that uses the low-pass filtered velocity field as the basis of the forcing term. This method is shown to double the range of scales captured by linear forcing while maintaining the flexibility and low computational cost of the original method. This translates to a 60% increase of the Taylor microscale Reynolds number on the same mesh. An extension is made to scalar mixing wherein a scalar field is forced to have an arbitrarily chosen, constant variance. Filtered linear forcing of the scalar field allows for control over the length scale of scalar injection, which could be important when simulating scalar mixing.

DOI: 10.1103/PhysRevFluids.3.034605

I. INTRODUCTION

Turbulent flows are ubiquitous to scientific and engineering applications. Often the applications are concerned with the dynamics of turbulence interaction with other processes such as combustion [1–3], particle-laden flow [4,5], and bubbly flow [6]. These types of flows naturally involve density and velocity fields that are rapidly varying or even discontinuous. Accordingly, simulations of these flows are often performed using physical space solvers, where it is easier to represent rapidly varying fields without the introduction of spurious numerical oscillations to the flow (e.g., Gibbs phenomena). As such, it is desirable to have a viable technique for forcing turbulence in physical space, which is the focus of this work.

The simplest and most often studied turbulent flow is homogeneous isotropic turbulence (HIT), which can be studied using direct numerical simulation (DNS). Statistics from these simulations can be used to create and validate mathematical models of turbulence; however, direct solution of the Navier-Stokes equations will not yield statistically stationary turbulence, as there is no mean shear. In order to achieve stationary statistics, an external force must be applied to sustain the flow.

For an incompressible fluid with constant density, ρ, and constant kinematic viscosity, ν, the dynamics of HIT are governed by the Navier-Stokes equations,

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \boldsymbol{u} + \boldsymbol{F}. \tag{1}$$

*jap436@cornell.edu
†olivier.desjardins@cornell.edu

In Eq. (1), $\boldsymbol{u}$ is the velocity, p is the pressure, t is time, and $\boldsymbol{F}$ is the forcing term. Equation (1) is coupled with the condition that the velocity field must be solenoidal, $\nabla \cdot \boldsymbol{u} = 0$. The time evolution of the mean turbulent kinetic energy, $k = \frac{1}{2}\langle \boldsymbol{u} \cdot \boldsymbol{u} \rangle$, is described by

$$\frac{\partial k}{\partial t} = -\varepsilon + \langle \boldsymbol{u} \cdot \boldsymbol{F} \rangle, \tag{2}$$

where angled brackets represent the Reynolds averaging operator, and $\varepsilon = \nu \langle \nabla \boldsymbol{u} : \nabla \boldsymbol{u} \rangle$ represents the viscous dissipation.

Since the seminal work of Orszag and Patterson [7], HIT simulations have typically been performed using spectral solvers on periodic domains under the assumption of zero-mean velocity. Under this regime, a number of studies were done on forcing isotropic turbulence, including those by Siggia [8], Kerr [9], Eswaran and Pope [10], Chen *et al.* [11], Sullivan *et al.* [12], Overholt and Pope [13], Alvelius [14], and many others. The specific nature of the forcing used in these studies varied, but all of these studies used forcing limited to a band of low-wave-number modes. The current work proposes to study a similar forcing strategy applied using a physical space solver.

II. LINEAR FORCING

Lundgren [15] proposed using the local velocity multiplied by a constant as the forcing term in physical space:

$$\boldsymbol{F} = A\boldsymbol{u}. \tag{3}$$

The physical meaning of A is apparent upon substituting the assumed statistical stationarity into the energy equation, Eq. (2), revealing $A = (2\tau)^{-1}$, where $\tau = k/\varepsilon$ is the eddy turnover time.

Lundgren demonstrated that the method was capable of producing realistic turbulence. Rosales and Meneveau [16] performed a more detailed study using Lundgren's method, finding that the characteristic large eddy size of the flow, $l = U^3/\varepsilon$, was consistently $0.2L$, where U is the root mean square (RMS) velocity and L is the box size used in the simulation. For spectral simulations they found $l \approx 0.4L$. Given that the smallest resolvable scale is fixed by the grid size, linear forcing will require twice the grid resolution to recover the same range of scales as a spectrally forced simulation. This corresponds to an eightfold increase in necessary computational resources in three dimensions.

The technique was further studied by Carroll and Blanquart [17], who noted that the steady state turbulent kinetic energy and dissipation, k_0 and ε_0, could be predicted *a priori*. They used a forcing of the form

$$\boldsymbol{F} = \frac{k_0}{k} A\boldsymbol{u}, \tag{4}$$

which causes a faster convergence to a statistically stationary state. The reason for this advantage is clear upon writing the equation for the energy production,

$$\mathcal{P} = \langle \boldsymbol{u} \cdot \boldsymbol{F} \rangle = \frac{k_0}{k} A \langle \boldsymbol{u} \cdot \boldsymbol{u} \rangle = \frac{k_0}{k} A 2k = \varepsilon_0. \tag{5}$$

The prefactor k_0/k dynamically adjusts the forcing such that a constant dissipation rate is achieved. A similar technique was used by Rosales and Meneveau [16]; however, they did not study the convergence properties of the scheme. From Eq. (2), it follows that the only stationary solution possible is $\varepsilon = \varepsilon_0$. Bassenne *et al.* [18] further improved the statistical convergence by modifying A to depend explicitly on both the current k and ε values.

Despite its limitations, linear forcing is the most used method to force turbulence in physical space, and has inspired a number of extensions. Various modifications have been developed to work for flows including homogeneous shear flows [19], anisotropic wall bounded flows [20], and compressible flows [21]. The success of linear forcing is owed largely to the simplicity of its implementation. The goal of this work is to demonstrate an extension to linear forcing that maintains its flexibility of use while recovering the turbulence scale resolution of low-wave-number, spectral forcing.

The most direct manner to approach this goal would be to apply a sharp spectral filter to the velocity field in order to restrict the forcing to low wave numbers. In practice, however, this form is not practical for several reasons. Physical space solvers are most often used to study problems where the use of the spectral transforms is unsuitable, so the use of a spectral filter may lead to nonphysical results. In this work, we demonstrate a low-pass filter which approximates the sharp spectral filter but only requires the use of a tridiagonal matrix solver. The study focuses on the characteristics of the flow fields generated using this approach and compares directly to linear forcing, which is the de facto standard for physical space HIT simulations. There is also a brief discussion of the effects of the filter sharpness on the resulting flow field. The modified forcing takes the form

$$\boldsymbol{F} = \frac{c}{k} A \widetilde{\boldsymbol{u}}, \tag{6}$$

where $\widetilde{\boldsymbol{u}}$ indicates a filtering operator has been applied to $\boldsymbol{u}$. The choice of value for the constant c is arbitrary. In this work, c is taken to be the predicted kinetic energy of the linearly forced simulation [17], which results in

$$\mathcal{P} = \langle \boldsymbol{u} \cdot \boldsymbol{F} \rangle = \frac{k_0}{k} A \langle \boldsymbol{u} \cdot \widetilde{\boldsymbol{u}} \rangle = \varepsilon_0 \frac{\langle \boldsymbol{u} \cdot \widetilde{\boldsymbol{u}} \rangle}{2k}. \tag{7}$$

The appropriateness of this choice is discussed in Sec. III C.

III. FILTERED LINEAR FORCING STRATEGY

A. Filtering approach

The identification of an appropriate filter is of the utmost importance to this work. It should be a sharp, low-pass filter that is defined on a compact domain. Raymond and Garder's [22] implicitly defined sine filter possesses these characteristics. On a discrete, one-dimensional data set, they define this filter using

$$(\mathcal{I} + \beta_{2p} \mathcal{D}^{2p}) \widetilde{\boldsymbol{u}} = \boldsymbol{u}, \tag{8}$$

where $\mathcal{D}^{2p}$ represents the second-order-accurate, central, finite difference approximation of the $2p$th derivative operator, and β_{2p} is a parameter related to the filter cutoff wave number. This filter is said to be of order $2p$. It is important to distinguish between the filter order, $2p$, and the order of accuracy of the numerical scheme used to compute Eq. (8), which is always second order in this work. Although it could be interesting to use more accurate discretizations of $\mathcal{D}^{2p}$, we have chosen to use a discretization consistent with our second-order-accurate flow solver. The filter order, $2p$, does not relate to the accuracy of the representation on the computational grid, but rather to the sharpness of the filter, as discussed below.

Analysis of the filtering operation, Eq. (8), is most naturally performed in spectral space for easy comparison to spectral filtering. For any field ϕ, we denote its Fourier transform as $\widehat{\phi}$. The function varies with the wave number vector, $\boldsymbol{\kappa}$, which has magnitude κ. The effects of the filtering operation can be understood by looking at the filter's transfer function, $\widehat{G} = \widehat{\widetilde{\phi}}/\widehat{\phi}$. For their work, Raymond and Garder chose β_{2p} as

$$\beta_{2p} = \frac{\Delta^{2p}}{(-4)^p \sin^{2p}\left(\frac{\kappa_c \Delta}{2}\right)}. \tag{9}$$

Equation (9) uses Δ for the grid spacing, and κ_c for the filter cutoff wave number. For the filtering operation to be meaningful, κ_c must be less than the largest resolvable wave number on the grid, $\kappa_\Delta = \pi/\Delta$. Substituting this definition of β_{2p} into Eq. (8) yields the transfer function

$$\widehat{G} = \left(1 + \frac{\sin^{2p}(\kappa \Delta/2)}{\sin^{2p}(\kappa_c \Delta/2)}\right)^{-1}, \tag{10}$$

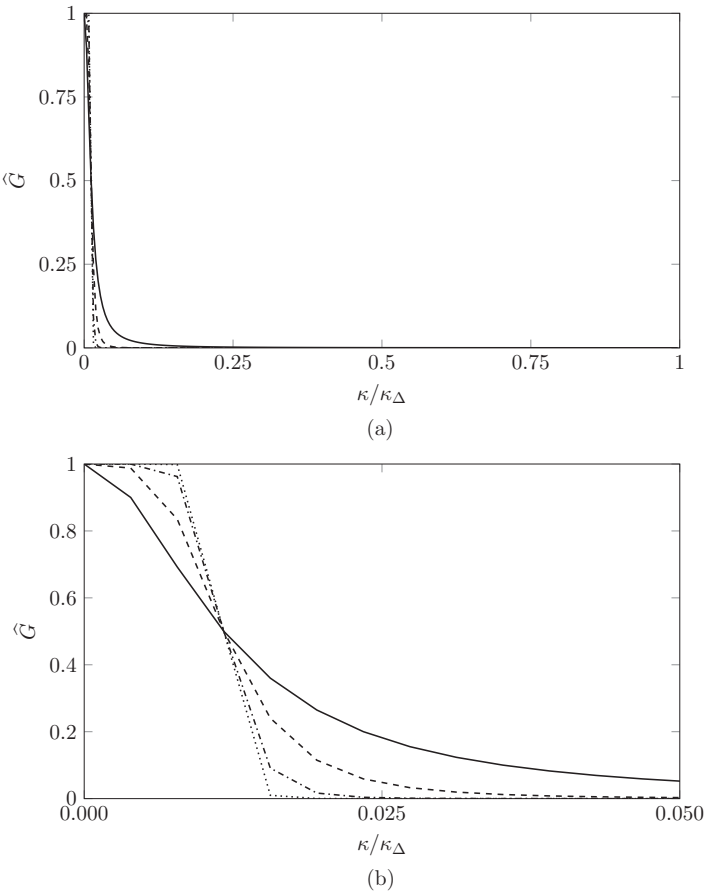

FIG. 1. Magnitude of the filter's transfer function, $\widehat{G}$, at various filter orders, $2p$, with $\kappa_c = 3$: $2p = 2$ (——), $2p = 4$ (– – –), $2p = 8$ (–·–·–), and $2p = 16$ (·····). (a) Full range of κ/κ_Δ. (b) $\kappa/\kappa_\Delta < 0.05$. Note the intersection of the curves at the cutoff frequency, $\kappa_c/\kappa_\Delta = 0.0117$.

i.e., a sine filter. In general, $\widehat{G}$ will be a complex-valued function; however, on a one-dimensional periodic domain with constant grid spacing, the operators generated using Eq. (8) always yield real-valued transfer functions.

This filter converges to the sharp spectral filter as $2p$ approaches infinity. Hence, the filter order, $2p$, represents the sharpness of the filter. Figure 1 demonstrates the transfer function of the filter at various orders for a typical value of κ_c.

The most obvious higher dimensional analog of Eq. (8) is

$$(\boldsymbol{\mathcal{I}} + \beta_{2p}\boldsymbol{\mathcal{L}}^{2p})\,\widetilde{\boldsymbol{u}} = \boldsymbol{u}, \tag{11}$$

where $\boldsymbol{\mathcal{L}}^{2p}$ represents the second-order discretization of the pth iterated Laplacian operator. Unfortunately, this does not result in a simple discretization. When discretized in wave-number space, the discrete Laplacian operator becomes

$$\sum_{i=1}^{3} \sin^2(\kappa_i \Delta/2), \tag{12}$$

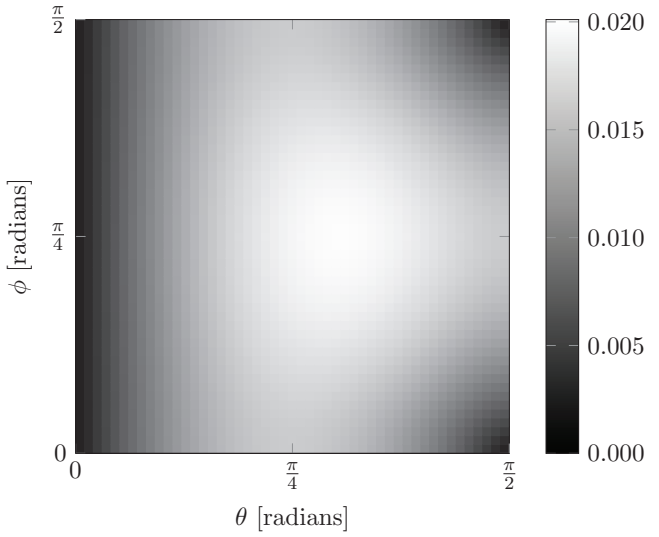

FIG. 2. RMS error of the $2p = 2$ order filter on a 512^3 grid, $\kappa_c = 3$.

where the summation occurs over the x, y, and z directions. Accordingly, the wave-number space representation of the pth iterated Laplacian is

$$\left[\sum_{i=1}^{3} \sin^2(\kappa_i \Delta/2)\right]^p, \tag{13}$$

yielding a transfer function of the form

$$\widehat{G}' = \left(1 + \beta_{2p}\left[\sum_{i=1}^{3} \sin^2(\kappa_i \Delta/2)\right]^p\right)^{-1}. \tag{14}$$

This result poses two problems. First, for the filter to be isotropic, Eq. (14) should be a function of $\kappa = \sqrt{\kappa_i{}^2}$ only; however, Eq. (14) is a function of an effective wave number $\kappa' = \sqrt{\sum_{i=1}^{3} \sin^2(\kappa_i \Delta/2)}$. Second, if κ' is used to define a relation for β_{2p} then the physical meaning of the cutoff wave number is lost, because there is no obvious relation between κ' and any relevant length scale.

In this work, the three-dimensional (3D) filter is taken as successive one-dimensional (1D) filters in the x, y, and z directions. The resulting filter is not isotropic for the same reason as the filter resulting from Eq. (11); however, the problem of defining β_{2p} is eliminated because a relation for β_{2p} is known in one dimension, viz., Eq. (9). If the κ_c are chosen to be the same for all three directions, this construction guarantees the elimination of waves with $\kappa > \sqrt{3}\kappa_c$. It is seen in Sec. III B that this formulation lends itself to a computationally inexpensive implementation.

It is worth noting that work by Bickley [23] indicates the possibility of composing an isotropic Laplacian stencil. The isotropicity comes with the trade-off of using a wider computational stencil, increasing the computational cost of the operation. Given its increased cost, the isotropic Laplacian was deemed inappropriate for this study, but it is mentioned here as it could be useful for other applications. That being said, we note that the filter given by successive 1D applications of Eq. (9) demonstrates small levels of anisotropy in practice. For this case, the anisotropy of the filter can be calculated analytically since the three-dimensional transfer function, $\widehat{G}_{3D}$, is known. Figure 2 shows contours of the RMS error between $\widehat{G}_{3\mathrm{D}}(\boldsymbol{\kappa})$ and the one-dimensional filter $\widehat{G}(\kappa)$ for a typical value

of κ_c. To compare the two functions, the spherical coordinate system,

$$\kappa_x = \kappa \sin\theta \cos\phi, \tag{15}$$

$$\kappa_y = \kappa \sin\theta \sin\phi, \tag{16}$$

$$\kappa_z = \kappa \cos\theta, \tag{17}$$

is used, and the error is computed as

$$\mathcal{E}(\theta,\phi) = \sqrt{\int_0^{\kappa_\Delta} [\widehat{G}(\kappa_x,\kappa_y,\kappa_z) - \widehat{G}(\kappa,0,0)]^2 d\kappa \Big/ \int_0^{\kappa_\Delta} d\kappa}. \tag{18}$$

The RMS error is less than 2.02% everywhere and, unsurprisingly, it takes its maximum value along the diagonal $\kappa_x = \kappa_y = \kappa_z$.

B. Implementation

Implementation of Eq. (8) in a general setting can be tedious because the expansion of $\mathcal{D}^{2p}$ must be known beforehand. Guédot *et al.* [24] used the Raymond and Garder filter to perform postprocessing of large eddy simulation and direct numerical simulation data. In their work, they noted that the algebraically equivalent problem of p second-order operators could be used instead of Eq. (8) by writing

$$(\mathcal{I} + \beta_{2p}\mathcal{D}^{2p})\widetilde{\boldsymbol{u}} = \prod_{i=1}^{p} (\mathcal{I} - \alpha_i \mathcal{D}^2)\widetilde{\boldsymbol{u}}. \tag{19}$$

Here the α_i are the pth roots of $-\beta_{2p}$. The p second-order operators in Eq. (19) are not explicitly multiplied, and instead each of the operators in the product is inverted in sequence. To use Eq. (19), only the form of $\mathcal{D}^2$ need be known *a priori*, which makes implementation straightforward for arbitrarily high $2p$. Inversion of this system, in general, requires a linear solver capable of dealing with the inversion of banded matrices with complex coefficients. In the work of Guédot *at al.*, they solved Eq. (19) using a preconditioned conjugate gradient solver wherein the real and imaginary components of the operator are solved separately [24]. They note that the high computational cost of their implementation is acceptable because of their usage of the filter as a postprocessing tool. In our work, we take advantage of the fact that DNS is often performed on Cartesian grids with uniform grid spacing. On such a grid, the second-order operators become tridiagonal matrices, which allow for quick and exact inversion in parallel using the Thomas algorithm. The computational cost associated with this operation is typically much lower than the overall cost of a time step of the flow solver. This fact is what allows our implementation of Eq. (19) to be sufficiently inexpensive for the filtering operation to be performed in line with simulations. For the remainder of this work, any mention of filtering refers to the solution of Eq. (19) using successive complex, tridiagonal matrix solvers.

C. Production

As pointed out in Sec. II, the filtered linear forcing results in a production that looks as

$$\mathcal{P} = \langle \boldsymbol{u} \cdot \boldsymbol{F} \rangle = \frac{k_0}{k} A \langle \boldsymbol{u} \cdot \widetilde{\boldsymbol{u}} \rangle = \varepsilon_0 \frac{\langle \boldsymbol{u} \cdot \widetilde{\boldsymbol{u}} \rangle}{2k}. \tag{20}$$

The impact of the filtering operation on the production is best studied by reexpressing Eq. (20) in terms of the discrete Fourier modes of $\boldsymbol{u}$. Let $\boldsymbol{K}$ denote the set of all mesh resolved wave numbers, i.e., $\boldsymbol{K} = \{\boldsymbol{\kappa} : |\boldsymbol{\kappa}| < \kappa_\Delta\}$. Then the production can be written as

$$\mathcal{P} = \frac{\varepsilon_0}{2k} \sum_{\boldsymbol{K}} \widehat{G}(\boldsymbol{\kappa}) \widehat{\boldsymbol{u}}^*(\boldsymbol{\kappa},t) \cdot \widehat{\boldsymbol{u}}(\boldsymbol{\kappa},t) = \frac{\varepsilon_0}{k} \sum_{\boldsymbol{K}} \widehat{G}(\boldsymbol{\kappa}) \widehat{k}(\boldsymbol{\kappa},t). \tag{21}$$

The notation $\widehat{\phi}^*$ denotes the complex conjugate of the value $\widehat{\phi}$. Here $\widehat{k}(\boldsymbol{\kappa})$ is the kinetic energy contained by the wave-number vector $\boldsymbol{\kappa}$, and $k = \sum_{\boldsymbol{K}} \widehat{k}$. Since the transfer function is independent of time, this can be further simplified to the form

$$\mathcal{P} = \alpha(t)\varepsilon_0. \tag{22}$$

For the case studied here, the transfer function is always a non-negative, real-valued function of the wave-number vector, and is upper bounded by 1. This leads naturally to a few observations:

(1) The coefficient is limited to the range $0 < \alpha < 1$. This implies that the production from filtered linear forcing is always non-negative, and that the energy injected into the system is less than what would be injected by linear forcing.

(2) In the limit of large κ_c (i.e., $\widehat{G} \approx 1$), the production for linear forcing is recovered, as $\alpha \approx 1$.

(3) As the flow approaches statistical stationarity, $\widehat{k}(\boldsymbol{\kappa},t)$ becomes nearly constant in time. It follows that α approaches a constant value that depends on the form of $\widehat{G}$, and the production approaches a constant value, $\mathcal{P} = \alpha\varepsilon_0$.

IV. COMPUTATIONAL RESULTS

A. Numerical framework

The computational framework used in this study is NGA, a structured, conservative code for use in low-Mach flow simulations [25]. The code solves the Navier-Stokes equation on a staggered (marker-and-cell) grid using a mix of finite volume and finite difference discretizations. In this work, the equations of motion are evolved using a Crank-Nicolson time integrator, and second-order-accurate finite volume spatial operators are used as in Ref. [17]. The grid is constructed using equal grid spacing and the same number of cells, N, in each of the three directions. Simulations are performed on a $(2\pi)^3$ periodic domain, so $\Delta = 2\pi/N$, with the restriction $\kappa_\Delta \eta > 1.5$, where η is the Kolmogorov length scale. The filter given by Eq. (19) is used for the forcing. The velocity field is initialized as in Ref. [16], which used the energy spectrum

$$E = \frac{16}{\sqrt{\pi/2}} \frac{U^2\kappa^4}{{\kappa_0}^5} \exp\left(-2\frac{\kappa^2}{\kappa_0^2}\right), \tag{23}$$

where $\kappa_0 = 2\pi/l_0$. The phase of each component of the velocity field is chosen randomly, and a correction is performed to ensure the velocity field is solenoidal. The value $l_0 = 0.2L$ is used for all simulations; however, it is known that the initialization plays little role in the long-term flow behavior [16].

B. Simulation parameters

The numerical study is organized in four parts: Cases 1–3 test the effects of filter cutoff by varying κ_c; Cases 2, 4, and 5 study the effects of filter sharpness by varying $2p$; the scaling of l is verified by running Cases 6 and 7 on half the number of grid cells and Case 8 on twice the number of grid cells compared to Case 1; and finally, a direct comparison is performed of fields obtained using linear forcing and filtered linear forcing at approximately the same Taylor microscale Reynolds number,

$$\mathrm{Re}_\lambda = \sqrt{15}\frac{U^2}{(\varepsilon\nu)^{1/2}}. \tag{24}$$

Cases 1 and 6 use linear forcing; all other cases use filtered linear forcing.

A summary of the data from all simulations is given in Table I. The reported values are averaged in space and time for a period after the flow has reached statistical stationarity. The cases are organized based upon their number of grid cells in each direction, N, and cutoff wave number, κ_c. The table includes an expected Reynolds number, $\mathrm{Re}_\lambda^{(*)}$, which is elaborated upon in Sec. IV E.

TABLE I. Summary of key values. Note that $A = 1.4\ \mathrm{s}^{-1}$ for all simulations.

Case	N	ν (m/s^2)	κ_c (1/m)	$2p$	$\mathrm{Re}_\lambda^{(*)}$	Re_λ	l/L	$\kappa_\Delta \eta$	ε (m^2/s^3)	U (m/s)
1	512	0.005	N/A	N/A	139	145	0.22	1.5	105	5.3
2	512	0.005	3	4	220	203	0.41	1.72	62	5.4
3	512	0.005	2	4	220	215	0.47	1.79	52	5.3
4	512	0.005	3	2	220	197	0.41	1.79	52	5.09
5	512	0.005	3	16	220	193	0.38	1.72	62	5.27
6	256	0.01275	N/A	N/A	87	88	0.20	1.52	105	5.12
7	256	0.01275	3	2	139	126	0.41	1.78	55	5.22
8	1024	0.00205	3	2	350	320	0.43	1.83	53	5.23

C. Effect of filtering

Cases 1–3 test the effects of filter cutoff wave number. A 512^3 linearly forced field is compared to filtered linearly forced fields using the same parameters with cutoff wave numbers $\kappa_c = 3$ and $\kappa_c = 2$. The filtered linearly forced simulations are performed using the fourth-order ($2p = 4$) filter. The parameters $A = 1.4$ and $\nu = 0.005$ are chosen to match those used by Carroll and Blanquart [17], who achieved $\mathrm{Re}_\lambda = 140$.

As expected, the linearly forced field, Case 1, matches the Taylor microscale Reynolds number, Re_λ, of Carroll and Blanquart [17]. The Reynolds number is higher for the filtered linearly forced fields, obtaining a Reynolds number comparable to that of Chen [11], who achieved $\mathrm{Re}_\lambda = 202$ using spectral forcing. Figure 3 shows the time evolution of the Reynolds number for all cases. Figures 4(a) and 5 show the spectra of energy and dissipation, respectively, normalized by the average dissipation rate and viscosity. They show almost identical spectra over a large range of wave numbers. The filtered linearly forced simulations have slightly higher energy at the lowest wave numbers compared to the linearly forced simulation, which is a result of the forcing technique directly amplifying those modes. Because a higher Re_λ is achieved, the energy spectra from the filtered linearly forced simulations is closer to the Kolmogorov $-5/3$ power decay. Noticeably, the time deviations of Re_λ from its mean are larger with the filtered linear forcing, suggesting a more unsteady process. This can be explained by noting that the larger l values of filtered linear forcing imply that there are fewer eddies of size l in the simulation domain. This, in turn, leads to poorer statistical sampling of the energy at these length scales and, accordingly, more fluctuating behavior.

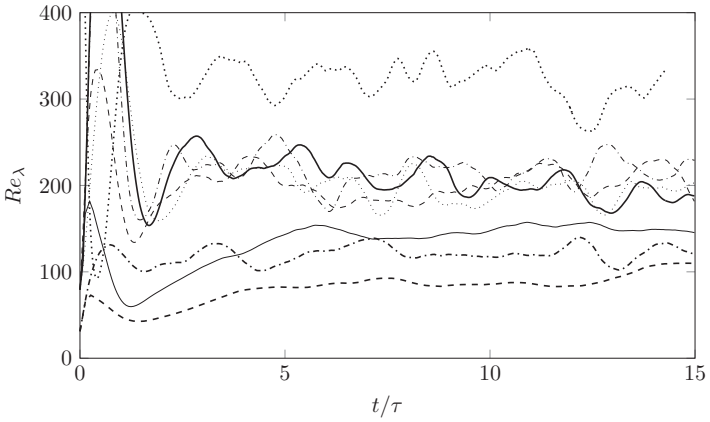

FIG. 3. Taylor microscale Reynolds number, Re_λ, as a function of nondimensional time: Case 1 (——), Case 2 (– – –), Case 3 (–·–·–), Case 4 (·····), Case 5 (——), Case 6 (– – –), Case 7 (–·–·–), and Case 8 (·····).

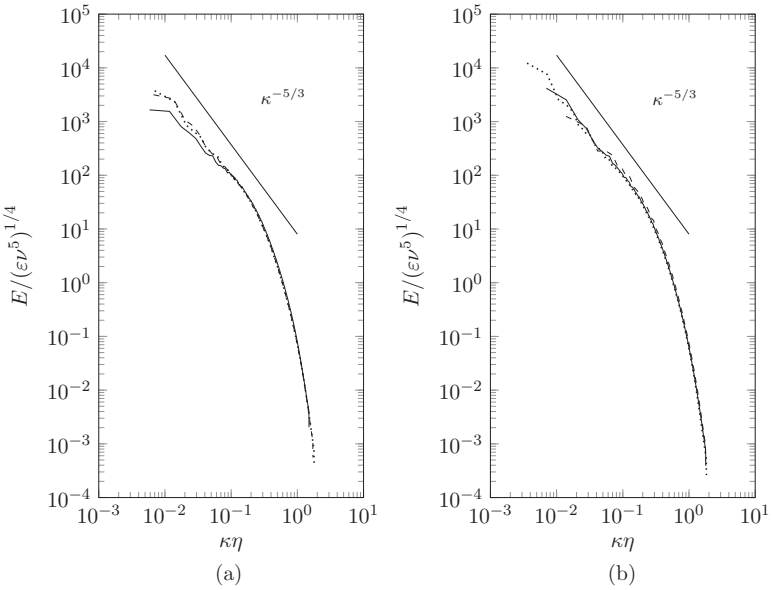

FIG. 4. Energy spectra normalized by dissipation and viscosity: (a) Case 1 (____), Case 2 (_ _ _), and Case 3 (……); (b) Case 4 (____), Case 7 (_ _ _), and Case 8 (……).

D. Effect of filter order

The effects of the filter order are tested in Cases 2, 4, and 5. These simulations use the same fluid parameters as Cases 1–3, and filtered linear forcing with $\kappa_c = 3$ and $2p = 4$, 2, and 16, respectively. There is no significant difference between the achieved Reynolds numbers. In Table I, it can be observed that the second-order ($2p = 2$) filter has a lower RMS velocity than either the $2p = 4$ or the $2p = 16$ case. This effect is compensated by the fact that the dissipation in this case is also lower. The result is a Re_λ that is approximately equal for all three cases. This seems to justify the use of filtered linear forcing with $2p = 2$, which can be inverted using a single, real-valued tridiagonal solver.

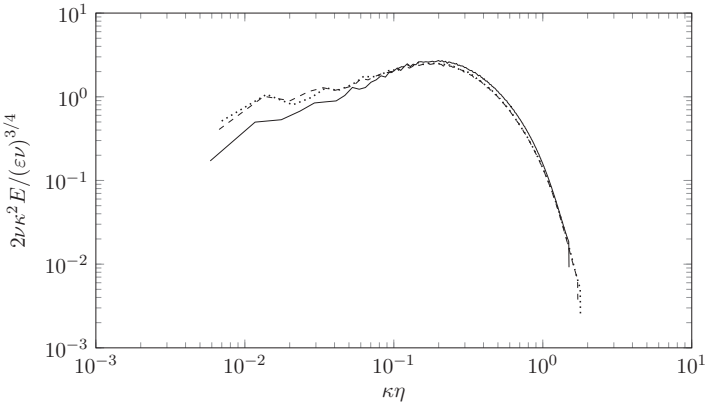

FIG. 5. Dissipation spectra normalized by dissipation and viscosity: Case 1 (____), Case 2 (_ _ _), and Case 3 (……).

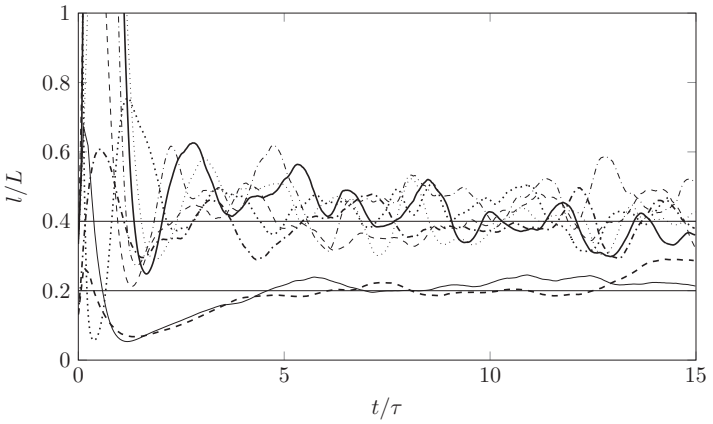

FIG. 6. Normalized characteristic eddy size, l/L, as a function of nondimensional time: Case 1 (——), Case 2 (– – –), Case 3 (–·–·–), Case 4 (·····), Case 5 (——), Case 6 (– – –), Case 7 (–·–·–), and Case 8 (·····). Horizontal lines are shown for $l/L = 0.2$ and $l/L = 0.4$.

E. Scaling

A parameter of particular interest for this simulation is the ratio l/L, which measures the effective resolution of the simulation. Figure 6 shows that the scaling $l/L \approx 0.4$ is achieved for all cases that use the filtered linear forcing approach, which is equivalent to spectral methods as demonstrated by Rosales and Meneveau [16]. Using this approximation, one can predict the simulation Reynolds number *a priori*. Recalling the definition of Re_λ, Eq. (24), the viscosity can be removed with the relation $\nu = (\varepsilon\eta^4)^{1/3}$, and the velocity can be removed with $U = (\varepsilon l)^{1/3}$. Finally we note that $L = N\Delta = N\pi/\kappa_\Delta$. These substitutions yield

$$\mathrm{Re}_\lambda^{(*)} = \sqrt{15}\left(\frac{\pi}{\kappa_\Delta \eta}\frac{l}{L}N\right)^{2/3}. \tag{25}$$

Observing that the parameter l/L is fixed by the simulation technique, and requiring $\kappa_\Delta\eta > 1.5$, the maximum resolvable Re_λ becomes a function of N only. Eq. (25) suggests that, for a fixed mesh, increasing the value of l/L is equivalent to increasing the Reynolds number. Accordingly, filtered linearly forced simulations with $l/L = 0.4$ yield a Reynolds number that is $2^{2/3} \approx 1.6$ times as large as linearly forced simulations, if all other parameters are left unchanged.

The scaling of Eq. (25) for linear forcing and filtered linear forcing was tested by performing simulations on 256^3 and 1024^3 grids (Cases 6–8). In all of these simulations, $A = 1.4$. The ν values for these cases were 0.01275 (Cases 6 and 7) and 0.00205 (Case 8). The agreement between predicted Reynolds number and the simulation is good, as results are within 13% of the expected values. Cases 4, 7, and 8 are filtered linearly forced fields of size 512^3, 256^3, and 1024^3, respectively. Each of these simulations was performed using $2p = 2$ and $\kappa_c = 3$, yielding Re_λ values of 197, 126, and 320, respectively. Figure 4(b) shows the normalized energy spectra for these cases. The spectra of all three cases show a similar shape. As Re_λ increases, the spectra demonstrate both larger large scales and smaller small scales. Case 8, the 1024^3 filtered linearly forced simulation, may be compared directly to the spectral simulation of Gotoh *et al.* [26], who achieved $\mathrm{Re}_\lambda = 381$ or only 16% greater. (It is important to note that Gotoh *et al.*'s simulation reports $\kappa_\Delta\eta = 1.3$, and, accordingly, Eq. (25) predicts a larger Re_λ.)

F. Direct comparison at $\mathrm{Re}_\lambda \approx 140$

Finally, Eq. (25) suggests that a filtered linearly forced simulation will achieve the same maximum Re_λ as a linearly forced simulation with twice the number of grid cells in each direction. This is

TABLE II. Summary of key values. Note that $A = 1.4\ \mathrm{s}^{-1}$ for all simulations.

Case	N	ν (m/s^2)	κ_c (1/m)	$2p$	$\mathrm{Re}_\lambda^{(*)}$	Re_λ	l/L	$\kappa_\Delta\eta$	ε (m^2/s^3)	U (m/s)
1	512	0.005	N/A	N/A	139	145	0.22	1.5	105	5.3
7	256	0.01275	3	2	139	126	0.41	1.78	55	5.22
9	256	0.01275	3	16	139	117	0.37	1.72	64	5.2
10	256	0.01275	3	∞	139	122	0.38	1.74	62	5.29

seen to be approximately true by comparing Cases 1 and 7, which represent a 512^3 linearly forced simulation and a 256^3 filtered linearly forced simulation, respectively. As a further comparison, two new simulations were added using the same grid and fluid parameters of Case 7. Case 9 uses filtered linear forcing with $2p = 16$; Case 10 uses a spectral forcing technique similar to that used by Sullivan *et al.* [12]. The forcing is defined as the inverse Fourier transform of

$$\widehat{\boldsymbol{f}}(\boldsymbol{\kappa}) = \begin{cases} \frac{k_0}{k} A\widehat{\boldsymbol{u}}(\boldsymbol{\kappa}) & \text{if } |\boldsymbol{\kappa}| < \kappa_c \\ 0 & \text{otherwise,} \end{cases} \tag{26}$$

which is the limiting case for filtered linear forcing as $2p \to \infty$. Table II shows that the time-averaged Re_λ for Cases 1, 7, 9, and 10 are 145, 126, 117, and 122, respectively. Over a wide range of wave numbers the spectra of the simulations are indistinguishable in Figs. 7 and 8. The dissipation spectra, shown in Fig. 8, demonstrate strong agreement in their shapes for the inertial and dissipative ranges even though there are noticeable differences between the large scale behaviors of Case 1 and Cases 7, 9, and 10. The differences in the fields can be better seen by looking at the structure function

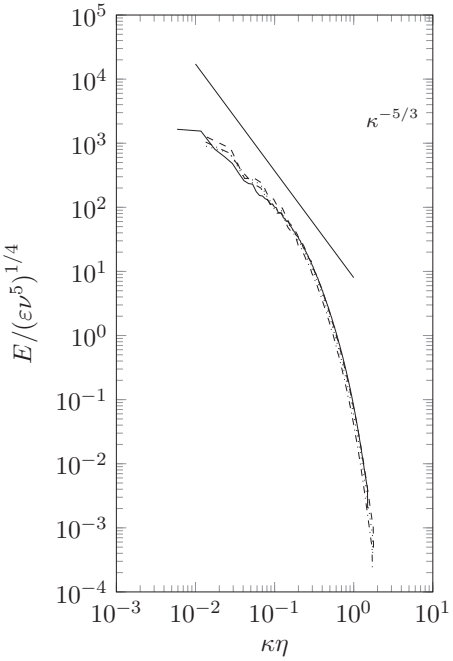

FIG. 7. Energy spectra normalized by dissipation and viscosity: Case 1 (____), Case 7 (_ _ _), Case 9 (_ . _ . _), and Case 10 (. . . . .).

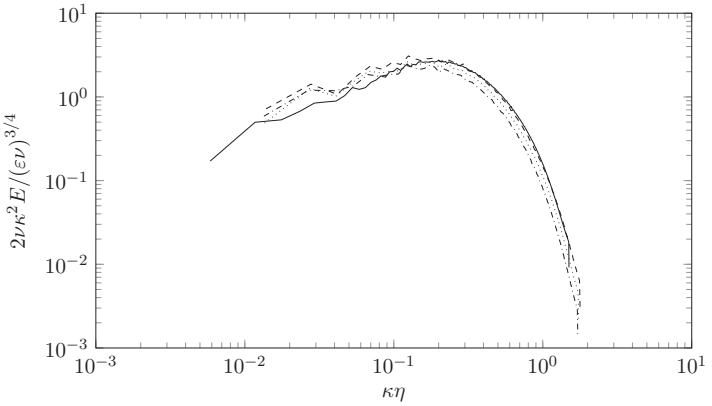

FIG. 8. Dissipation spectra normalized by dissipation and viscosity: Case 1 (——), Case 7 (– – –), Case 9 (–·–·–), and Case 10 (·····).

defined,

$$D_{LL}(r) = \frac{1}{3}\sum_i \langle (u_i(\boldsymbol{x} + r\boldsymbol{e}_i) - u_i(\boldsymbol{x}))^2 \rangle, \tag{27}$$

where summation occurs over each of the x, y, and z directions, and $\boldsymbol{e}_i$ represents the unit vector in the i direction. Figure 9 shows that the structure functions do not demonstrate Kolmogorov scaling [27], i.e., $D_{LL} \propto (\varepsilon r)^{2/3}$; however, Cases 7 and 9, the filtered linearly forced simulations, shows qualitative agreement with Case 10, as well as the $\mathrm{Re}_\lambda = 120$ case from Gotoh *et al.* [26]. This concretely demonstrates the value of this method; Cases 7 and 9 are able to achieve the same turbulence resolution as Case 1 while using only 1/8 of the number of grid cells.

G. Computational expense

Some discussion of the computational expense of the numerical methods is warranted. Compared to linear forcing, filtered linear forcing requires the solution of multiple tridiagonal matrix solvers for each filtering operation. Equation (19) decomposes the $2p$ order operator into p applications of the second-order filter. Since the filter is applied separately in each of the x, y, and z directions three

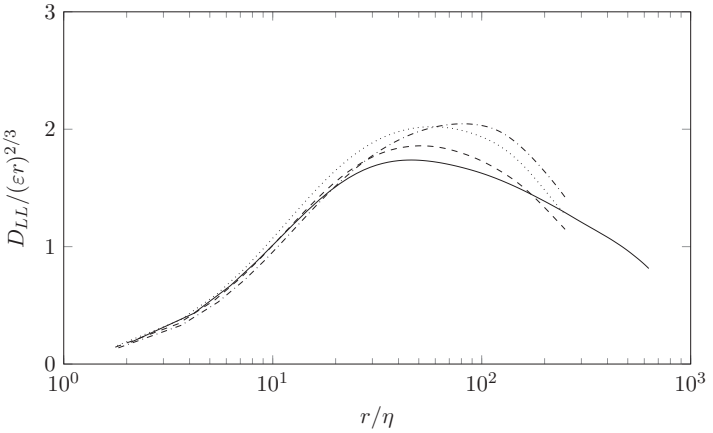

FIG. 9. Normalized structure function: Case 1 (——), Case 7 (– – –), Case 9 (–·–·–), and Case 10 (·····).

TABLE III. Cost per time step of several forcing strategies.

Forcing strategy	$2p$	t (s)
Linear forcing	N/A	2.95
Filtered linear forcing	2	4.36
Filtered linear forcing	4	10.2
Filtered linear forcing	8	12.5
Filtered linear forcing	16	22.2
Spectral forcing	N/A	4.05

applications of this filter are required, totaling $3p$ tridiagonal matrix solvers for each component of the velocity field. This can be compared to the cost of doing a spectral solution, such as the one used in Sec. IV F which requires a Fourier and inverse Fourier transform. The expected cost of a tridiagonal solver is $O(N)$, whereas the cost of a fast Fourier transform is $O(N \log_2 N)$, suggesting that the filtering approach is computationally cheaper than the spectral one. However, this fact must be counterbalanced by the fact that the filtering approach requires the solution of several tridiagonal solvers in series. To compare the cost of filtered linear forcing at various $2p$, simulations were performed on a 512^3 mesh using the parameters of Case 4. The average time per time step was computed for filtered linearly forced simulations using orders of $2p = 2$, 4, 8, and 16 and compared to a linearly forced simulation. A spectrally forced simulation is also shown. The results are given in Table III. Unsurprisingly, the cost of filtered linearly forced simulations increases monotonically with $2p$. The cost per time step of the $2p = 4$ simulation is more than twice that of the $2p = 2$ simulation. This can be attributed to the fact that the filter using $2p = 2$ is computed using a real-valued solver, whereas the $2p > 2$ solvers require the use of complex-valued solvers, which are more expensive. The cost of the $2p = 2$ simulation is roughly similar to that of using the spectral technique, although filtered linear forcing appears to be slightly more expensive. Given that the spectral transform gives the most sharp representation of the forcing, it is concluded that in those situations where spectral transforms are appropriate, they should be used. However, the use of the physical space filtered linear forcing technique provides a framework that works in more general environments, i.e., nonperiodic conditions. In terms of cost per time step, all of the forcing strategies used here are more expensive than linear forcing. However, considering the greater range of turbulent scales that these simulation can compute, they are more computationally efficient to reach a given Re_λ.

V. EXTENSION TO PASSIVE SCALAR MIXING

Carroll *et al.* [28] performed a study of scalar forcing where they were able to force a scalar to constant variance. To achieve this they added a forcing term F_Z to the scalar transport equation,

$$\frac{\partial Z}{\partial t} + \boldsymbol{u} \cdot \nabla Z = D\nabla^2 Z + F_Z, \tag{28}$$

where Z is the scalar and D is its molecular diffusivity. By analogy with Lundgren [15], they advocate for linear forcing, and use

$$F_Z = \frac{\chi}{\sigma^2} Z \tag{29}$$

as the source term, where $\sigma^2 = \langle Z^2 \rangle - \langle Z \rangle^2$ is the scalar variance, and $\chi = \langle 2D|\nabla Z|^2 \rangle$ is the scalar dissipation. In practice, this term will not be sufficient to reach statistical stationarity due to numerical dissipation from the scalar transport scheme. To counteract this process they use a modified form of the forcing,

$$F_Z = \left[\frac{1}{\tau_r} \left(\sqrt{\frac{\alpha}{\sigma^2}} - 1 \right) + \frac{\chi}{\sigma^2} \right] Z, \tag{30}$$

TABLE IV. Summary of key values. For all simulations $A = 0.319\ \mathrm{s}^{-1}$, $\nu = D = 0.0075\ \mathrm{m/s}^2$ on a 256^3 grid, which yields $\mathrm{Re}_\lambda \approx 50$.

Case	κ_c (1/m)	α	σ^2	λ_Z (m)	r
Z1	N/A	1	0.991	0.141	0.588
Z2	N/A	5.616	5.57	0.140	0.583
Z3	N/A	0.17	0.175	0.140	0.581
Z4	3	1	0.936	0.180	0.960
Z5	3	5.616	5.26	0.180	0.967
Z6	3	0.17	0.166	0.183	0.993

where α is the target value of the variance and τ_r is an arbitrary constant which acts as the relaxation time. Equation (30) converges to Eq. (29) as σ^2 approaches α. A simple extension to this would be to filter the scalar before using it as a source, i.e.,

$$F_Z = \left[\frac{1}{\tau_r}\left(\sqrt{\frac{\alpha}{\sigma^2}} - 1\right) + \frac{\chi}{\sigma^2}\right]\widetilde{Z}. \tag{31}$$

Carroll *et al.* [28] have shown that low-wave-number forcing produces statistics consistent with those from forcing a scalar field using an imposed mean gradient. The only significant difference they note is that low-wave-number forcing produces isotropic fields. It is also expected that low-wave-number filtered linear scalar forcing will generate fields with larger ranges of length scales compared to linear scalar forcing.

A series of simulations is performed to test the proposed scalar forcing technique. All simulations are performed on a 256^3 grid with $\nu = D = 0.0075$. The scalar field is initialized using the same initialization of Sec. IV A, replacing the RMS velocity with the RMS scalar value. The velocity field uses linear forcing with $A = 0.319$ which yields $\mathrm{Re}_\lambda \approx 50$. The scalars are transported using a fifth-order-accurate advection scheme [29] and a second-order-accurate, finite volume treatment of diffusion. Simulations use either linear scalar forcing [Eq. (30)] or filtered linear scalar forcing [Eq. (31)] for arbitrarily chosen variances. All simulations use $\tau_r = 0.1$ as the scalar relaxation time, and the second-order filter is used. A summary of the results is shown in Table IV. The reported values are averaged in space and time for a period after the flow has reached statistical stationarity. Table IV also introduces a Taylor scale based on the scalar field defined as

$$\lambda_Z = \sqrt{\frac{3D\sigma^2}{\chi}}. \tag{32}$$

For all cases the variance is near its desired value, α, as can be seen in Fig. 10. The variance for the filtered linearly forced scalar is somewhat lower because the energy injection rate is lower, viz., $\langle Z\tilde{Z}\rangle < \langle Z^2\rangle$. The filtered linearly forced fields demonstrate greater time variation around their mean values. This is consistent with what was seen for the velocity field in filtered linear velocity forcing. Figure 11 shows the normalized scalar variance spectrum. All spectra maintain a similar shape throughout their range.

Because all simulations Z1–Z6 use the same Batchelor scale, $\eta_B = (D/\nu)^{1/2}\eta$, Fig. 11 offers little information about the difference in mixing length scales between the simulations. Table IV shows that the low-wave-number filtered linearly forced simulations achieve higher Taylor microscales, λ_Z, which imply a larger range of scales. This effect could also be elucidated using the scalar integral length scale [30], defined as

$$l_Z = \left(\frac{\varepsilon^{1/3}\sigma^2}{\chi}\right)^{3/2}. \tag{33}$$

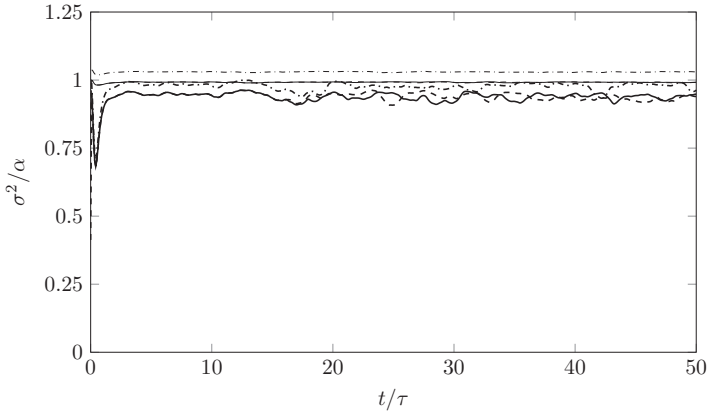

FIG. 10. Scalar variance, σ^2, normalized by the desired variance, α: Case Z1 (——), Case Z2 (– – –), Case Z3 (–·–·–), Case Z4 (——), Case Z5 (– – –), and Case Z6 (–·–·–).

Noting the definition of λ_Z and η_B, this can also be expressed in the form

$$l_Z = \frac{\mathrm{Sc}^{1/2}}{3^{3/2}} \frac{\lambda_Z^3}{\eta_B^2}, \tag{34}$$

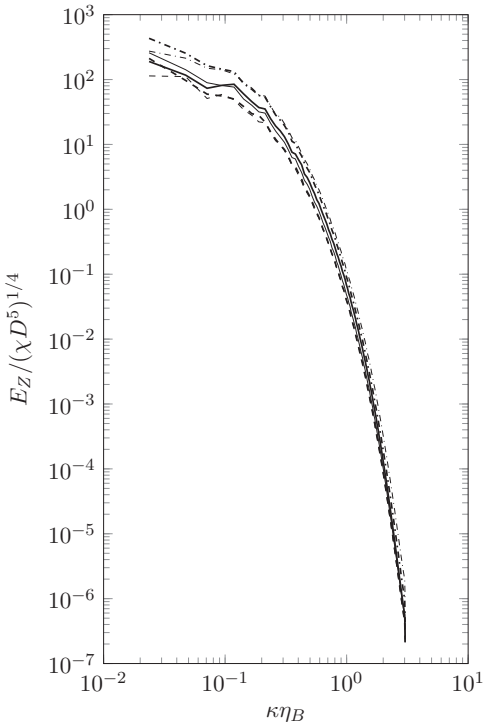

FIG. 11. Normalized scalar variance spectrum. The wave number has been nondimensionalized by the Batchelor length scale, $\eta_B = (D/\nu)^{1/2}\eta$. Case Z1 (——), Case Z2 (– – –), Case Z3 (–·–·–), Case Z4 (——), Case Z5 (– – –), and Case Z6 (–·–·–).

where the Schmidt number, $\mathrm{Sc} = \nu/D$, has been introduced. Because all simulations in this study use the same Sc and η_B, it follows that λ_Z and l_Z contain the same information.

For passive scalar mixing problems, the scalar time scale $\tau_Z = \sigma^2/\chi$ is important to characterize the physics of the mixing process. Since τ is known *a priori* for a linearly forced velocity field, this becomes equivalent to characterizing the time scale ratio $r = \tau_Z/\tau$. Experiments by Warhaft and Lumley [31] show that τ_Z is a function of the length scale at which the scalar variance is injected, implying that r is a function of the flow type. Indeed, it can be shown [30] that

$$r = \frac{2}{3}\left(\frac{l_Z}{l}\right)^{2/3}, \tag{35}$$

where it can be seen that r depends directly on the length scale of scalar variance injection. (Note that the coefficient $\frac{2}{3}$ is modified from the form given in Ref. [30] due to the use of $l = U^3/\varepsilon$ instead of $l = k^{3/2}/\varepsilon$.) It therefore seems that the filtering technique could be exploited to perform simulations that better match experimental conditions. In particular, the application of a bandpass filter to the scalar forcing field would allow arbitrary control over the injection length scale. The development and study of such a filtering operation is, however, beyond the scope of this project.

VI. CONCLUSIONS

Many computational studies of turbulence look at the interplay between turbulence and multiphysics problems. In these problems, physical space implementations of flows solvers are often more natural to use than spectral implementations. This work introduces filtered linear forcing, a technique for simulating homogeneous isotropic turbulence in physical space. The technique uses a filtered velocity field as a source term in the Navier-Stokes equations in order to force the velocity field towards statistical stationarity. It is shown that neither the cutoff frequency nor the sharpness of the filter chosen are of great importance. Using the least sharp version of the implicitly defined family of filters [22], the filtering operation takes the form of a computationally inexpensive tridiagonal matrix solver. The technique is capable of increasing the value of Re_λ by 60% compared to linear forcing [15]. Furthermore, results are compared directly to spectral simulations [11,26] and are shown to produce similar results.

For scalar mixing, a filtered scalar field is added as a source to the scalar transport equation. It is shown to produce a statistically stationary scalar field. Interestingly, the time scale ratio τ_Z/τ changes depending on whether linear scalar forcing or filtered linear scalar forcing is used. This hints at a potential advantage of the filtering operation: its ability to control the form of scalar variance injection. This latter ability could be of interest to create simulations that have the same mixing dynamics as experiments [31].

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grants No. DGE-1144153 and No. DGE-1650441. This material is based upon work supported by the Alfred P. Sloan Foundation. This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the US Department of Energy under Contract No. DE-AC05-00OR22725.

[1] E. Knudsen, E. S. Richardson, E. M. Doran, H. Pitsch, and J. H. Chen, Modeling scalar dissipation and scalar variance in large eddy simulation: Algebraic and transport equation closures, Phys. Fluids **24**, 055103 (2012).

[2] S. Lapointe, B. Savard, and G. Blanquart, Differential diffusion effects, distributed burning, and local extinctions in high Karlovitz premixed flames, Combust. Flame **162**, 3341 (2015).
[3] B. Savard, B. Bobbitt, and G. Blanquart, Structure of a high Karlovitz n-C_7H_{16} premixed turbulent flame, Proc. Combust. Inst. **35**, 1377 (2015).
[4] K. Yeo, S. Dong, E. Climent, and M. Maxey, Modulation of homogeneous turbulence seeded with finite size bubbles or particles, Int. J. Multiphase Flow **36**, 221 (2010).
[5] M. Esmaily-Moghadam and A. Mani, An analytical description of clustering of inertial particles in turbulent flows, Phys. Rev. Fluids **1**, 1 (2016).
[6] A. Toutant, E. Labourasse, O. Lebaigue, and O. Simonin, DNS of the interaction between a deformable buoyant bubble and a spatially decaying turbulence: A priori tests for LES two-phase flow modelling, Comput. Fluids **37**, 877 (2008).
[7] S. A. Orszag and G. S. Patterson, Numerical Simulation of Three-Dimensional Homogeneous Isotropic Turbulence, Phys. Rev. Lett. **28**, 76 (1972).
[8] E. D. Siggia, Numerical study of small-scale intermittency in three-dimensional turbulence, J. Fluid Mech. **107**, 375 (1981).
[9] R. Kerr, Higher-order derivative correlations and the alignment of small-scale structures in isotropic numerical turbulence, J. Fluid Mech. **153**, 31 (1985).
[10] V. Eswaran and S. Pope, An examination of forcing in direct numerical simulations of turbulence, Comput. Fluids **16**, 257 (1988).
[11] S. Chen and G. Doolen, On statistical correlations between velocity increments and locally averaged dissipation in homogeneous turbulence, Phys. Fluids **5**, 458 (1993).
[12] N. P. Sullivan, S. Mahalingam, and R. M. Kerr, Deterministic forcing of homogeneous, isotropic turbulence, Phys. Fluids **6**, 1612 (1994).
[13] M. Overholt and S. Pope, A deterministic forcing scheme for direct numerical simulations of turbulence, Comput. Fluids **27**, 11 (1998).
[14] K. Alvelius, Random forcing of three-dimensional homogeneous turbulence, Phys. Fluids **11**, 1880 (1999).
[15] T. Lundgren, Linearly forced isotropic turbulence, in *Center for Turbulence Research Annual Research Briefs* (Center for Turbulence Research, Stanford University, Stanford, CA, 2003), pp. 461–473.
[16] C. Rosales and C. Meneveau, Linear forcing in numerical simulations of isotropic turbulence: Physical space implementations and convergence properties, Phys. Fluids **17**, 095106 (2005).
[17] P. L. Carroll and G. Blanquart, A proposed modification to Lundgren's physical space velocity forcing method for isotropic turbulence, Phys. Fluids **25**, 105114 (2013).
[18] M. Bassenne, J. Urzay, G. I. Park, and P. Moin, Constant-energetics physical-space forcing methods for improved convergence to homogeneous-isotropic turbulence with application to particle-laden flows, Phys. Fluids **28**, 035114 (2016).
[19] S. Hirabayashi and T. Sato, A novel numerical forcing for homogeneous stratified turbulence in full energy equilibrium, Comput. Fluids **39**, 1789 (2010).
[20] B. de Laage de Meux, B. Audebert, R. Manceau, and R. Perrin, Anisotropic linear forcing for synthetic turbulence generation in large eddy simulation and hybrid RANS/LES modeling, Phys. Fluids **27**, 035115 (2015).
[21] M. R. Petersen and D. Livescu, Forcing for statistically stationary compressible isotropic turbulence, Phys. Fluids **22**, 116101 (2010).
[22] W. Raymond and A. Garder, A review of recursive and implicit filters, Mon. Weather Rev. **119**, 477 (1991).
[23] W. G. Bickley, Finite difference formulae for the square lattice, Q. J. Mech. Appl. Math. **1**, 35 (1948).
[24] L. Guedot, G. Lartigue, and V. Moureau, Design of implicit high-order filters on unstructured grids for the identification of large scale features in large-eddy simulation and application to a swirl burner, Phys. Fluids **27**, 045107 (2015).
[25] O. Desjardins, G. Blanquart, G. Balarac, and H. Pitsch, High order conservative finite difference scheme for variable density low Mach number turbulent flows, J. Comput. Phys. **227**, 7125 (2008).
[26] T. Gotoh, D. Fukayama, and T. Nakano, Velocity field statistics in homogeneous steady turbulence obtained using a high-resolution direct numerical simulation, Phys. Fluids **14**, 1065 (2002).

[27] A. Kolmogorov, The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers, Proc. R. Soc. London Ser. A **434**, 9 (1991).
[28] P. L. Carroll, S. Verma, and G. Blanquart, A novel forcing technique to simulate turbulent mixing in a decaying scalar field, Phys. Fluids **25**, 095102 (2013).
[29] R. R. Nourgaliev and T. G. Theofanous, High-fidelity interface tracking in compressible flows: Unlimited anchored adaptive level set, J. Comput. Phys. **224**, 836 (2007).
[30] J. R. Ristorcelli, Passive scalar mixing: Analytic study of time scale ratio, variance, and mix rate, Phys. Fluids **18**, 075101 (2006).
[31] Z. Warhaft and J. Lumley, An experimental study of the decay of temperature fluctuations in grid-generated turbulence, J. Fluid Mech. **88**, 659 (1978).