Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 225 (2007) 20–32
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Efficient algorithm for simulating homogeneous turbulent shear flow without remeshing

Kyle A. Brucker [1], Juan C. Isaza, T. Vaithianathan [2], Lance R. Collins *

Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14853-7501, USA

Received 11 April 2006; received in revised form 21 September 2006; accepted 9 October 2006
Available online 8 May 2007

Abstract

The Rogallo (1981) algorithm for simulating homogeneous turbulent shear flow solves the equations on a mesh that in physical space deforms with the mean flow. Eventually, when the mesh reaches a particular degree of deformation, the coordinate system must be "remeshed". Remeshing introduces unavoidable numerical errors such as loss of turbulent kinetic energy and dissipation rate and is therefore not desirable. In this paper, we present a new algorithm for simulating homogeneous turbulent shear flow, based on a Fourier decomposition of the velocity field, that avoids the troublesome remeshing step. Equations for the Fourier amplitudes of three components of velocity are advanced in time; however, nonlinear terms are calculated on a stationary, orthogonal mesh in physical space, allowing traditional de-aliasing procedures to be used. The resulting spectral transforms involve a phase shift to account for the frame of reference change; consequently, the standard three-dimensional fast Fourier transforms (3D FFT) cannot be used. We have developed an algorithm that accomplishes the spectral transforms at a computational cost that still scales like $O(N^3 \ln N)$ operations, where N is the number of grid points in each direction. A fully parallel version of the algorithm has been implemented to run on 2^p processors, where p is a positive integer. Results over a broad range of the shear parameter, S^*, demonstrate the advantages of avoiding the remeshing step.

Keywords: Turbulence; Homogeneous turbulent shear; Direct numerical simulations; Pseudospectral algorithm; Remeshing

1. Introduction

Homogeneous turbulent shear flow has long been considered one of the "building block" flows of turbulence. It is the next step up in complexity from isotropic turbulence, but still retains the simplicity of

* Corresponding author. Tel.: +1 814 863 7113; fax: +1 814 865 7846.
E-mail address: LC246@cornell.edu (L.R. Collins).
[1] Current address: Mechanical Engineering, UCSD, San Diego, CA, USA.
[2] Current address: Clear Science Co., 663 Owego Hill Road, Harford, NY 13784-0233, USA.

0021-9991/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2006.10.018

homogeneity, which allows it to be analyzed at a multitude of levels (e.g., see Refs. [1–3] for examples). Moreover, homogeneous turbulent shear flow has many of the key features found in wall-bounded flows (e.g., Reynolds stress, turbulence production, hairpin vortices), without introducing the complexities of fully inhomogeneous turbulence [4]. Consequently, homogeneous turbulent shear flow continues to receive attention in the numerical literature [5–7] and in experimental studies [8].

The workhorse algorithm for simulating homogeneous turbulent shear flow using a pseudospectral method [9,10] is the one developed by Rogallo [11]. In it, the velocity field is decomposed into Fourier modes based on a grid (in physical space) that deforms with the uniform mean shear. This coordinate system is chosen so that all quantities (e.g., velocity and pressure) are periodic in all three directions. This enables a standard three-dimensional fast Fourier transform (3D FFT) algorithm to be used to transform the variables back and forth between the physical and Fourier spaces. A drawback of this method is that the physical-space mesh deforms continuously with time. To relieve this problem, the standard Rogallo algorithm maps the variables on the positively deformed mesh onto a mesh that is deformed in the opposite direction. This step is known as ''remeshing''. As a consequence of the stretching of the coordinate that accompanies the mesh deformation, 50% of the wavenumbers in the direction of shear must be zeroed out. This leads to a sudden loss in both the turbulent kinetic energy and turbulent energy dissipation rate. For low shear rates, the loss is on the order of 5% and therefore acceptable [4]. At higher shear rates, this loss can be a larger fraction of the total (20–40%), motivating some researchers to eliminate the remeshing step entirely, choosing the numerical error associated with the deformed mesh over the errors that arise from remeshing [12].

Baron [13] recognized the shortcomings of remeshing, and instead developed a finite difference discretization scheme that satisfies the ''shear-periodic'' boundary conditions. The algorithm was improved upon by Schumann [14] and used in a series of numerical investigations of stably stratified sheared turbulence [15–17].

Noting the incompatibility between homogeneity and stationarity detailed in earlier studies [18–20], Schumacher and Eckhardt [6] developed a numerical algorithm that produces stationary turbulence that is *nearly* homogeneous. Their method uses periodic boundary conditions in the stream-wise and span-wise directions and free-slip boundary conditions in the shear direction. They use Fourier modes in the periodic directions and Chebyshev functions in the shear direction. Mean shear is introduced through a body force that varies linearly in the direction of the shear. Because of the free-slip boundary conditions at the wall, the small-scale statistics reach a stationary state in contrast to purely homogeneous turbulent shear flows that evolve indefinitely in time. The authors claim that the small-scale structures of the flow should be nearly identical to those found in statistically homogeneous turbulence. Using this algorithm, Schumacher and coworkers have completed a series of studies of the anisotropy of the velocity derivatives in shear flow [7,21,22]. They showed evidence of sensitivity to the shear parameter that contradicts the earlier understanding from experiments [8,23].

Yu and Girimaji [24] studied an alternative approximation to homogeneous turbulent shear flow. They considered turbulence confined between two infinite parallel plates using the lattice Boltzmann method to describe the flow. The flow field was periodic in the stream-wise and span-wise directions and no-slip boundary conditions were applied at the two bounding walls. The initial condition for the velocity was a superposition of a laminar Couette flow at the intended shear rate and a randomly generated isotropic turbulent flow field. They claim that the flow in the core of the channel is nearly homogeneous turbulent shear flow for an intermediate time before the onset of fully-developed turbulent Couette flow. They studied the effects of the shear parameter and Reynolds number on the turbulence statistics during this intermediate time period.

This paper discusses a pseudospectral algorithm for *purely homogeneous turbulent shear flow* that circumvents the need to periodically remesh the coordinate system in physical space and avoids the use of no-slip or stress-free boundary conditions. The algorithm avoids remeshing by performing a phase shift that maintains the physical-space representation on the orthogonal mesh throughout the calculation. The key ingredient is that the algorithm is able to accomplish the spectral transform with the same order of steps as the traditional 3D FFT uses. The basic algorithm will be discussed in Section 2 and its efficient implementation in a distributed parallel code is presented in Section 3. Results of sample calculations along with timings of the code are discussed in Section 4, followed by conclusions in Section 5.

2. Numerical algorithm

2.1. Governing equations

We are interested in the flow of an incompressible fluid in a periodic box of length 2π in each direction. The governing equations for the fluid are

$$\frac{\partial u_i}{\partial x_i} = 0, \tag{1}$$

$$\frac{\partial u_i}{\partial t} + \epsilon_{ijk}\,\omega_j u_k = -\frac{\partial\left(p/\rho + \frac{1}{2}u^2\right)}{\partial x_i} + \nu\,\frac{\partial^2 u_i}{\partial x_j\,\partial x_j}, \tag{2}$$

where u_i is the velocity vector, $u \equiv \sqrt{u_i u_i}$ is the magnitude of the velocity vector, ρ is the fluid density, ν is the kinematic viscosity, ϵ_{ijk} is the alternating unit symbol, $\omega_i = \epsilon_{ijk}\frac{\partial u_k}{\partial x_j}$ is the vorticity, and p is the pressure.

We introduce the Reynolds decomposition as follows:

$$u_i = U_i + u_i', \tag{3}$$

$$\omega_i = \Omega_i + \omega_i', \tag{4}$$

$$p = P + p', \tag{5}$$

where we have adopted the nomenclature of capital letters for mean quantities and prime letters for fluctuating quantities. For simple shear flow we take (without loss of generality)

$$U_i = (\mathcal{S}x_2, 0, 0), \quad \Omega_i = (0, 0, -\mathcal{S}), \tag{6}$$

where $\mathcal{S}$ is the spatially uniform mean shear rate. Taking the ensemble average of Eqs. (1) and (2) yields

$$\frac{\partial U_i}{\partial x_i} = 0, \tag{7}$$

$$\frac{\partial U_i}{\partial t} + \epsilon_{ijk}\left(\Omega_j U_k + \overline{\omega_j' u_k'}\right) = -\frac{\partial\left[P/\rho + \frac{1}{2}\left(U^2 + \overline{u'^2}\right)\right]}{\partial x_i} + \nu\frac{\partial^2 U_i}{\partial x_j\,\partial x_j}. \tag{8}$$

Subtracting the equations for the mean velocity from the total yields the equations for the fluctuating velocity

$$\frac{\partial u_i'}{\partial x_i} = 0, \tag{9}$$

$$\frac{\partial u_i'}{\partial t} + \epsilon_{ijk}\left(\Omega_j u_k' + \omega_j' U_k + \omega_j' u_k' - \overline{\omega_j' u_k'}\right) = -\frac{\partial\left[p'/\rho + U_j u_j' + \frac{1}{2}\left(u'^2 - \overline{u'^2}\right)\right]}{\partial x_i} + \nu\frac{\partial^2 u_i'}{\partial x_j\,\partial x_j}. \tag{10}$$

Invoking homogeneity and combining and simplifying the terms that involve the mean flow, the final form of the equation in physical space is

$$\frac{\partial u_i'}{\partial t} + \mathcal{S}x_2\frac{\partial u_i'}{\partial x_1} + \mathcal{S}\delta_{i1}u_2' + \epsilon_{ijk}\omega_j' u_k' = -\frac{\partial p^*}{\partial x_i} + \nu\frac{\partial^2 u_i'}{\partial x_j\,\partial x_j}, \tag{11}$$

where $p^* \equiv p'/\rho + \frac{1}{2}u'^2$ is the modified pressure.

2.2. Spectral transform

The system of equations is not periodic in the laboratory frame due to the presence of the uniform mean shear. A "shear periodic boundary condition" in the direction of the mean shear involves a shift of the domain in the laboratory frame as shown below for an arbitrary variable ϕ

$$\phi(x_1 + 2\pi\mathcal{S}t, x_2 + 2\pi, x_3) = \phi(x_1, x_2, x_3), \tag{12}$$

$$\frac{\partial\phi}{\partial x_2}(x_1 + 2\pi\mathcal{S}t, x_2 + 2\pi, x_3) = \frac{\partial\phi}{\partial x_2}(x_1, x_2, x_3). \tag{13}$$

Fig. 1 shows a schematic of the shear periodic boundary condition in two dimensions, with the direction of shear in the vertical direction. The dashed lines show the deforming frame of reference in which the flow is periodic. The solid lines indicate the orthogonal frame of reference. Forward and reverse transforms for the variable ϕ in the orthogonal frame of reference are defined as follows:

$$\phi(\mathbf{x},t) = \frac{1}{N^3}\sum_{\mathbf{k}} \hat{\phi}(\mathbf{k},t)\exp[I(k_i x_i - \mathcal{S}tk_1x_2)], \tag{14}$$

$$\hat{\phi}(\mathbf{k},t) = \sum_{\mathbf{x}} \phi(\mathbf{x},t)\exp[-I(k_i x_i - \mathcal{S}tk_1x_2)], \tag{15}$$

where $I \equiv \sqrt{-1}$, and N is the number of grid points in each direction. The cross term in the exponential, $\mathcal{S}tk_1x_2$, arises due to the shear-periodic boundary condition. As a consequence of this term, it is not possible to calculate the forward and reverse transforms using a standard 3D FFT algorithm.

This issue was resolved by Rogallo [11] by changing the frame of reference into one that deforms with the mean flow, i.e.,

$$\begin{aligned} x_1' &\equiv x_1 + \mathcal{S}tx_2, \\ x_2' &\equiv x_2, \\ x_3' &\equiv x_3. \end{aligned}$$

The transform in the x_i' coordinate system reduces to the standard 3D FFT. The only drawback is that this frame implies a constant distortion of the mesh in physical space, leading to a growth in aliasing errors that arises from evaluating the nonlinear terms on the deformed mesh. To relieve this problem, Rogallo introduced a remeshing step. For a cube domain of size 2π, the remeshing is done when $\mathcal{S}t = 1/2, 3/2, 5/2, \ldots$. As already noted, this introduces some undesirable features such as energy and dissipation loss.

To avoid this issue, we introduce an algorithm that works directly with Eqs. (14) and (15) in the orthogonal frame of reference. The difficulty is to accomplish the transform in $O(N^3 \ln N)$ operations, where N is the number of grid points in each direction. This is accomplished by decomposing the three-dimensional transform into a sum of products of one- and two-dimensional transforms (i.e., pencils and planes). To illustrate this point, we show the procedure for evaluating a forward transform.

The first step is to transform x_1, x_3 to k_1, k_3 by calling N two-dimensional real $\rightarrow$ complex FFTs yielding for an arbitrary physical-space variable $\phi(x_1, x_2, x_3)$

$$\check{\phi}(k_1, x_2, k_3) = \sum_{x_1}\sum_{x_3} \phi(x_1, x_2, x_3)\exp[-I(k_1x_1 + k_3x_3)]. \tag{16}$$

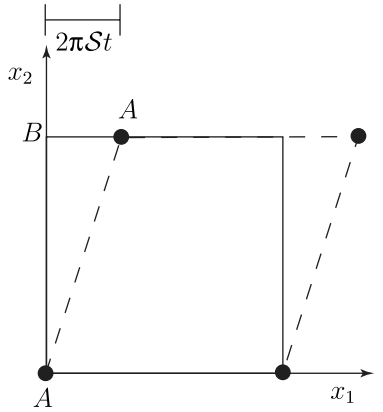


Fig. 1. Schematic of the shear periodic boundary conditions in two dimensions. Mean shear of magnitude $\mathcal{S}$ lies in the vertical direction. Solid lines indicate orthogonal frame; dashed lines indicate deforming frame in which boundary conditions are periodic. Black dots are periodic points.

We then phase shift the result to accommodate the uniform mean shear

$$\tilde{\phi}(k_1, x_2, k_3) \equiv \breve{\phi}(k_1, x_2, k_3) \exp[I\mathcal{S}tk_1x_2]. \tag{17}$$

The transform is completed by calling N^2 one-dimensional complex → complex FFTs to obtain

$$\hat{\phi}(k_1, k_2, k_3) = \sum_{x_2} \tilde{\phi}(k_1, x_2, k_3) \exp[-Ik_2x_2]. \tag{18}$$

The first calculation over planes will scale like $O(N \times N^2 \ln N)$ operations, the multiplication step $O(N^3)$, and the final transform over pencils like $O(N^2 \times N \ln N)$, yielding an overall scaling of $O(N^3 \ln N)$ operations.

We can similarly outline the reverse transform. The first step is to perform N^2 one-dimensional complex → complex pencil transformations to obtain

$$\tilde{\phi}(k_1, x_2, k_3) = \frac{1}{N} \sum_{k_2} \hat{\phi}(k_1, k_2, k_3) \exp[Ik_2x_2]. \tag{19}$$

Next is the multiplication step

$$\breve{\phi}(k_1, x_2, k_3) \equiv \tilde{\phi}(k_1, x_2, k_3) \exp[-I\mathcal{S}tk_1x_2], \tag{20}$$

followed by N two-dimensional complex → real, plane FFTs, yielding the desired transform

$$\phi(x_1, x_2, x_3) = \frac{1}{N^2} \sum_{k_1} \sum_{k_3} \breve{\phi}(k_1, x_2, k_3) \exp[I(k_1x_1 + k_3x_3)]. \tag{21}$$

Again, the overall calculation scales as $O(N^3 \ln N)$ operations.

2.3. Spectral transform of governing equations

We can apply the spectral transform defined in Eq. (15) to the governing equations. We define the velocity and pressure transforms as follows:

$$\hat{u}_i(\mathbf{k}, t) = \sum_{\mathbf{x}} u_i'(\mathbf{x}, t) \exp[-Ik_i'x_i], \tag{22}$$

$$\hat{p}^*(\mathbf{k}, t) = \sum_{\mathbf{x}} p^*(\mathbf{x}, t) \exp[-Ik_i'x_i], \tag{23}$$

where we introduce the modified wavevector k_i' is defined as

$$k_i' \equiv k_i - \mathcal{S}tk_1\delta_{i2}. \tag{24}$$

Applying the spectral transformation to the continuity relationship yields

$$\hat{u}_ik_i' = 0. \tag{25}$$

Transformation of the time derivative in the Navier–Stokes equation yields

$$\mathscr{F}\left\{\frac{\partial u_i'}{\partial t}\right\} = \frac{\partial \hat{u}_i}{\partial t} - I\mathcal{S}k_1x_2\hat{u}_i. \tag{26}$$

The second term arises from the fact that time is explicit in the shear-periodic spectral transform. This term precisely cancels the transform of the convective term in Eq. (11). Transformation of spatial gradients follows the usual rule, with k_i' replacing k_i

$$\mathscr{F}\left\{\frac{\partial u_j'}{\partial x_i}\right\} = Ik_i'\hat{u}_j.$$

Applying these relationships to the transform of Eq. (11) yields

$$\frac{\partial \hat{u}_i}{\partial t} + \epsilon_{ijk}\mathscr{F}\{\omega_j'u_k'\} = -Ik_i'\hat{p}^* - \nu k^2\hat{u}_i - \mathcal{S}\delta_{i1}\hat{u}_2, \tag{27}$$

where

$$k^2 \equiv k_i' k_i' = k_1^2 + (k_2 - \mathcal{S}tk_1)^2 + k_3^2, \tag{28}$$

and $\mathscr{F}\{\}$ denotes the spectral transform defined by Eq. (15).

It is customary, in simulations of homogeneous turbulence, to eliminate the pressure from the transformed Navier–Stokes equation using the transformed continuity relationship. To proceed, we differentiate the continuity equation with respect to time, yielding

$$\frac{\partial}{\partial t} k_i' \hat{u}_i = (k_i - \mathcal{S}tk_1\delta_{i2}) \frac{\partial \hat{u}_i}{\partial t} - \mathcal{S}k_1 \hat{u}_2 = 0. \tag{29}$$

Substituting for $\partial \hat{u}_i/\partial t$ from the momentum equation, simplifying and solving for $\hat{p}^*$ yields

$$\hat{p}^* = \frac{I}{k^2}\left[k_i' \epsilon_{ijk} \mathscr{F}\{\omega_j' u_k'\} + 2\mathcal{S}k_1 \hat{u}_2\right], \tag{30}$$

where the viscous term is zero due to continuity.

Substituting this back into Eq. (27) and simplifying yields the final evolution equation for the Fourier coefficients

$$\left[\frac{\partial}{\partial t} + \nu k^2\right]\hat{u}_i = \left(-\delta_{im} + \frac{k_i' k_m'}{k^2}\right)\epsilon_{mjk}\mathscr{F}\left\{\omega_j' u_k'\right\} + 2\frac{k_i'}{k^2}k_1 \mathcal{S}\hat{u}_2 - \delta_{i1}\mathcal{S}\hat{u}_2. \tag{31}$$

3. Efficient parallel implementation

The algorithm discussed in Section 2 has been implemented in a distributed computational environment using message passing interface (MPI). This section gives a short summary of the CPU, communication and memory considerations in setting up the parallel code in an optimal way in FORTRAN 90. The main challenge involves efficiently computing the spectral transforms.

3.1. Memory allocation

In transform space, each variable can be stored in a complex array of size $[N/2 + 1, N, N]$, where $N = 2^p$ is the size of the calculation (typically $7 \leqslant p \leqslant 10$ corresponding to $128 \leqslant N \leqslant 1024$), and the first index is reduced nearly in half by the conjugate relationship

$$\hat{\phi}(-\mathbf{k}) = \hat{\phi}^*(\mathbf{k}), \tag{32}$$

where the superscript $*$ refers to the complex conjugate. This ensures that the spectral transform in physical space is a real quantity. To subdivide the arrays across the computing cluster, each array is divided along the third dimension, producing array components of size $[N/2 + 1, N, M]$, where $M \equiv N/N_{\text{proc}}$ and N_{proc} is the number of processors. This balances the load across all of the processors, although it complicates the calculation of spectral transforms, as the data on all processors is required to complete a spectral transform. The numerical approach to circumvent this problem is discussed in the next section. The code carries seven variables (two components of velocity and five working variables) and so the memory requirement per node for a double-precision calculation with $N^3 = 2^{3p}$ grid points is (in gigabytes)

$$\text{Mem} = \frac{7 \times 2^{[3(p+1)-30]}}{N_{\text{proc}}}\ \text{GB}. \tag{33}$$

Thus, for example, a 1024^3 calculation on 32 nodes requires less than 2 GB per node.

3.2. Parallel FFT

We first consider the inverse transform of a spectral variable $\hat{\phi}(k_1, k_2, k_3)$. As noted above, the data along the third dimension is divided among the N_{proc} processors. Each node has complete information in the k_1–k_2

plane, enabling the computation of one-dimensional complex-to-complex transforms to obtain $\tilde{\phi}(k_1, x_2, k_3)$, as described by Eq. (19). The phase shifting step shown in Eq. (20) is also computed locally on each node to produce $\check{\phi}(k_1, x_2, k_3)$. The final step of the inverse transform is the evaluation of the two-dimensional transforms shown in Eq. (21). In order to complete this step, the data must be reorganized so that the full k_1–k_3 planes are available on each node. The data are efficiently redistributed through a call to the MPI function ALLTOALL [25], resulting in data now divided along the second dimension. The two-dimensional complex-to-real transforms are now performed on each node to obtain $\phi(x_1, x_2, x_3)$ (with the data still divided among the processors along the second index).

3.3. Temporal update

Discretization in time is accomplished using a second-order, explicit Runge–Kutta algorithm. The viscous term on the left-hand side of Eq. (31) is handled exactly by introducing the integrating factor $\exp[\nu k^2 t]$ to both sides of the equation. The resulting expression can be written as

$$\frac{\partial \hat{v}_i}{\partial t} = \mathrm{RHS}_i \exp[\nu k^2 t], \tag{34}$$

where

$$\hat{v}_i = \hat{u}_i \exp[\nu k^2 t],$$
$$\mathrm{RHS}_i = \left(-\delta_{im} + \frac{k_i' k_m'}{k^2}\right) \epsilon_{mjk} \mathscr{F}\{\omega_j' u_k'\} + 2\frac{k_i'}{k^2} k_1 \mathcal{S}\hat{u}_2 - \delta_{i1} \mathcal{S}\hat{u}_2.$$

The discrete form of Eq. (34) is then

$$\hat{v}_i^{n+1} = \hat{v}_i^n + \frac{h}{2}\{\mathrm{RHS}_i^n \exp[\nu k^2 t] + \mathrm{RHS}_i^{n+1} \exp[\nu k^2 (t+h)]\},$$

where n is the iteration, h is the time increment, and

$$\mathrm{RHS}_i^n = \mathrm{RHS}_i(\hat{v}_i^n),$$
$$\mathrm{RHS}_i^{n+1} = \mathrm{RHS}_i\left(\hat{v}_i^n + h\mathrm{RHS}_i^n\right).$$

Expressing the result in terms of $\hat{u}_i$ yields the second-order update we used in the code

$$\hat{u}_i^{n+1} = \hat{u}_i^n \exp[-\nu k^2 h] + \frac{h}{2}\left\{\mathrm{RHS}_i^n \exp[-\nu k^2 h] + \mathrm{RHS}_i^{n+1}\right\}. \tag{35}$$

Aliasing errors that result from the pseudo-spectral evaluation of the quadratic nonlinear terms were removed by a method commonly used in isotropic simulations. The velocity components in spectral space were set to zero following each update for $k \geqslant k_{\max}$, where $k_{\max} = \sqrt{2}N/3$ is the maximum wavenumber. In addition, the nonlinear terms computed on the grid were averaged with the value obtained on a coordinate system shifted in all three directions by an amount $\Delta x/2$. The combined effect of zeroing part of the spectrum and averaging the nonlinear terms on the grid with the phase-shifted values eliminates the entire aliasing error. A full description of the method can be found in Appendix A of Ref. [26].

3.4. Parallel performance

The parallel performance of the code was evaluated by running 10 iterations for different size calculations (64^3–512^3 grid points) and number of processors (1–32 processors). The computational facility has 16 nodes, each having two AMD MP2200$^+$ processors with 3 GB of memory. The nodes are interconnected via standard dual-channel Gb Ethernet. A summary of the timings for various simulations is shown in Table 1. Notice that for the smaller calculations (64^3 and 128^3) the scaling is less than linear, while the scaling at the largest size of 512^3 is much closer to linear. As the size of the calculation increases, the relative proportion of CPU to communication increases, improving the scaling of the code. This leads to an optimal number of nodes for a given calculation size (i.e., 1 node for 64^3, 4 nodes for 128^3, 8 nodes for 256^3 and 16 nodes for 512^3).

Table 1
Summary of the parallel performance of the code for double precision calculations at the indicated size and number of nodes

Nodes	64^3	128^3	256^3	512^3
1	3.84	32.29	311.07	–
2	2.65	22.36	197.68	–
4	1.5	12.15	108.4	914.77
8	0.77	6.75	57.25	478.03
16	0.46	4.72	39.53	244.16
32	0.31	6.92	26.06	178.92

Times are reported in seconds (wall clock).

4. Results and discussion

To test the new code, we perform direct numerical simulations of homogeneous turbulent shear flow at different values of the shear parameter and compare the results with simulations done with the Rogallo algorithm. For simplicity we shall use the velocity vector $u_i = (u, v, w)$ in the standard coordinate system (x, y, z), where the mean shear is $\mathcal{S} = \mathrm{d}U/\mathrm{d}y$. The initial velocity is generated using a random phase algorithm, with the initial energy spectrum given by

$$E(k) = C_\kappa \epsilon^{2/3} k_0^{-5/3} \begin{cases} (k/k_0)^2 & \text{if } k < k_0, \\ (k/k_0)^{-5/3} & \text{if } k_0 \leqslant k \leqslant k_\eta, \\ 0 & \text{if } k > k_\eta, \end{cases}$$

where $C_\kappa \approx 1.5$ is the Kolmogorov constant, ϵ is the prescribed energy dissipation rate, k_0 is the location in the peak of the energy spectrum, and k_η is the maximum energy-containing wavenumber, defined to be consistent with the dissipation spectrum as

$$\frac{k_\eta}{k_0} \equiv \left[\frac{2\epsilon^{1/3}}{3\nu C_\kappa k_0^{4/3}} + \frac{11}{15} \right]^{3/4}.$$

The condition $k_{\max}\eta \geqslant 1$ is the criterion used to ensure adequate resolution of the small scales, where $k_{\max} \equiv \sqrt{2}N/3$ is the maximum resolved wavenumber and $\eta \equiv (\nu^3/\epsilon)^{1/4}$ is the Kolmogorov length scale. Fig. 2 shows the initial energy spectrum for the 128^3 runs. The value of k_0 is chosen to force the initial integral

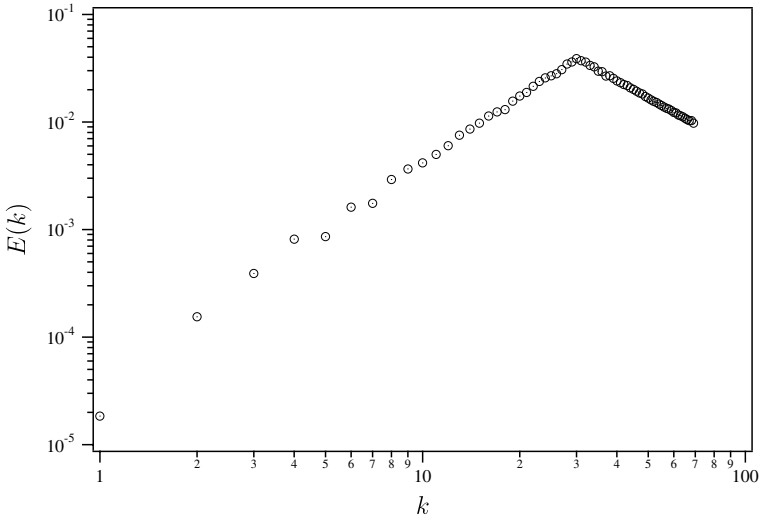


Fig. 2. Initial energy spectrum used for all of the homogeneous turbulent shear flow simulations.

length scale to be moderately small, since the large scales grow in homogeneous turbulent shear flows and the calculation must be stopped when their size is comparable to the computational box size of 2π. We used the criterion that the integral length scale cannot exceed 10% of the size of the computational box. Several 128^3 and 256^3 runs are compared to the Rogallo code for low and high values of the shear parameter $S^* \equiv \mathcal{S}q/\epsilon$, where q is the turbulence kinetic energy and ϵ is the turbulence dissipation.

4.1. Comparison with rogallo code

As seen in Fig. 3, there is nearly perfect agreement between the two codes at the lowest shear rate corresponding to $S^* \sim 5$. However at the highest shear rate, shown in Fig. 4, the Rogallo algorithm yields a saw-tooth pattern in some of the statistics due to the remeshing steps. It is important to note that the reduction in the dissipation rate is more pronounced than the losses to the kinetic energy or Reynolds stress because information is lost at high wavenumbers following the remeshing procedure.

Due to the problem of remeshing at high shear parameter values, Lee et al. [12] chose not to remesh the grid, arguing that the adverse effect of frequent remeshing (due to the high shear parameter) on the small scales

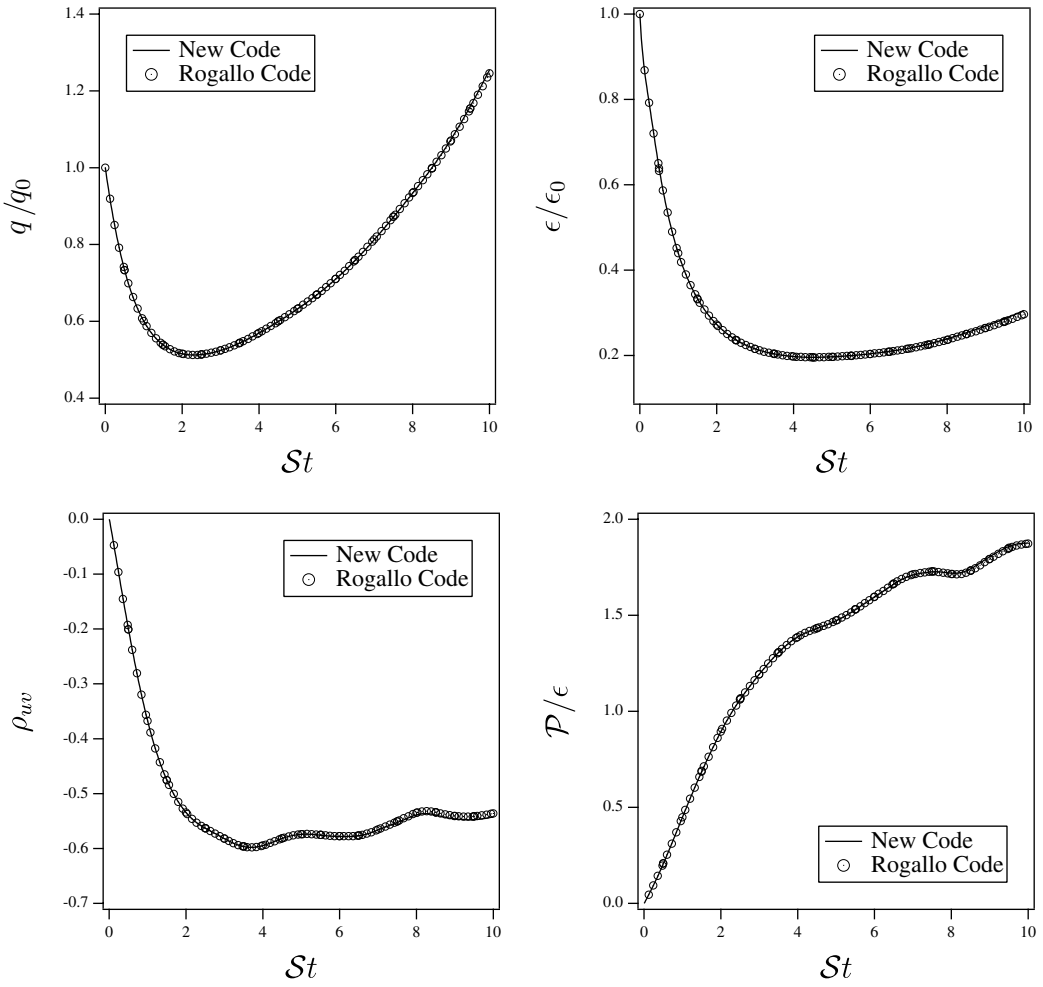
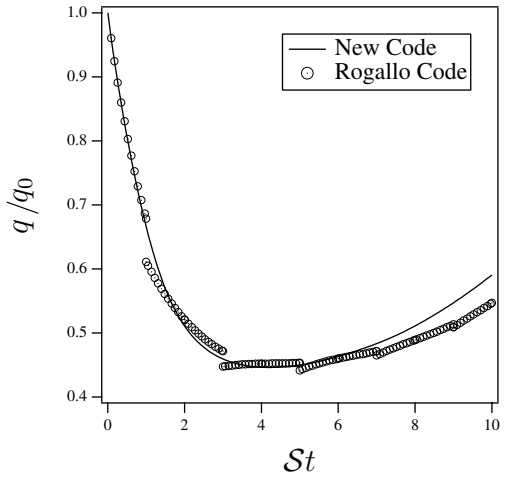
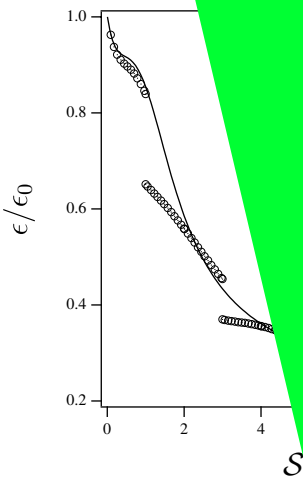
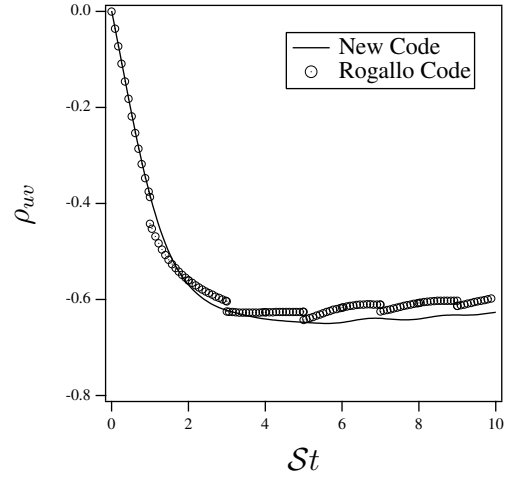


Fig. 3. Time evolution of the turbulent kinetic energy (upper left) and energy dissipation rate (upper right), Reynolds stress normalized by the square root of the product of the variances for u' and v' (lower left), and production over dissipation (lower right) for a low value of the shear parameter $S^* \sim 5$. The solid line is the result from the new algorithm and the markers are the result of the Rogallo code with remeshing.







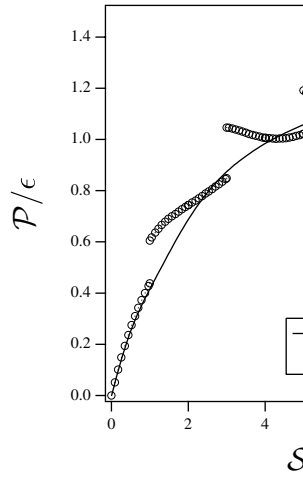
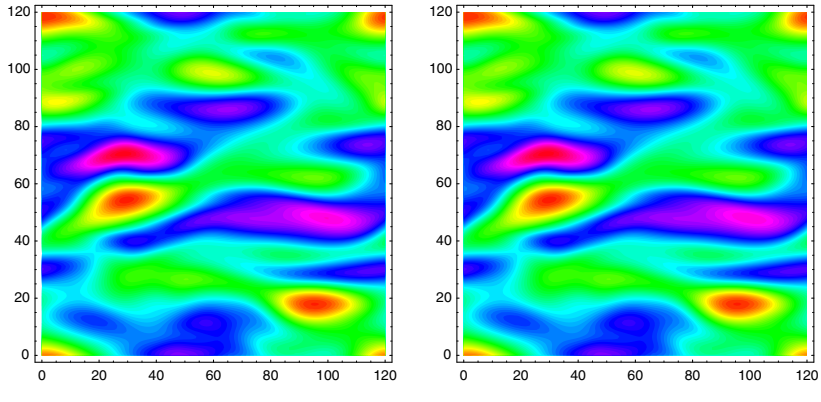
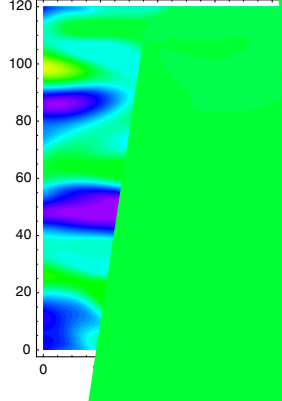


Fig. 4. Time evolution of the turbulent kinetic energy (upper left) and energy dissipation rate (upper right), Reynolds st
the square root of the product of the variances for u' and v' (lower left), and production over dissipation (lower right) f
the shear parameter $S^* \sim 30$. The solid line is the result from the new algorithm and the markers are the result of the R
remeshing.



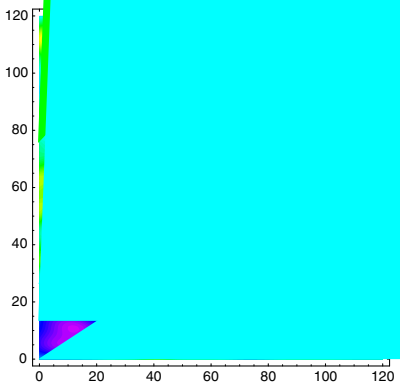


was m
Thus,
velocit
phase
deriva

Iso
case o
shing
three
remes
in Fig
reme
more
the n
ance
while
rithn
ities
disto

4.2.

T
now
resp
latio



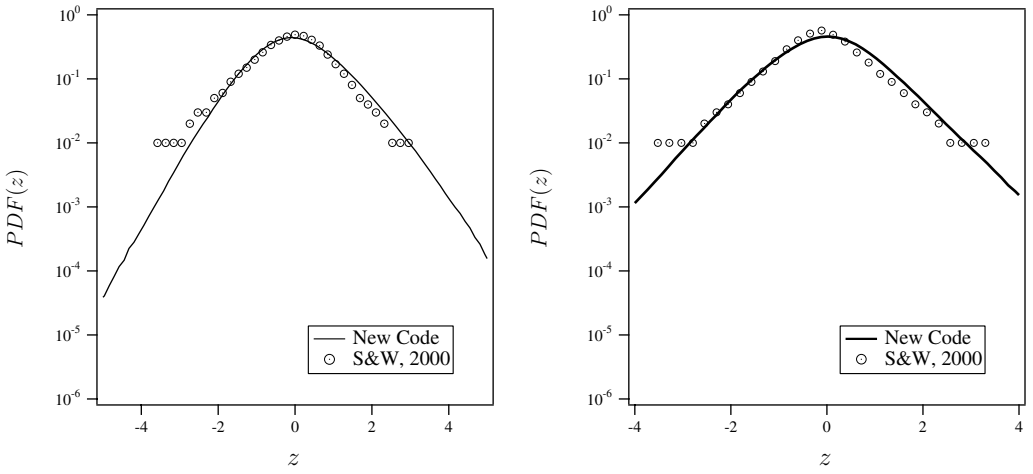


Fig. 7. Probability density functions of $z = (\frac{\partial u}{\partial x})/\langle(\frac{\partial u}{\partial x})^2\rangle^{1/2}$ (left) and $z = (\frac{\partial v}{\partial x})/\langle(\frac{\partial v}{\partial x})^2\rangle^{1/2}$ (right).

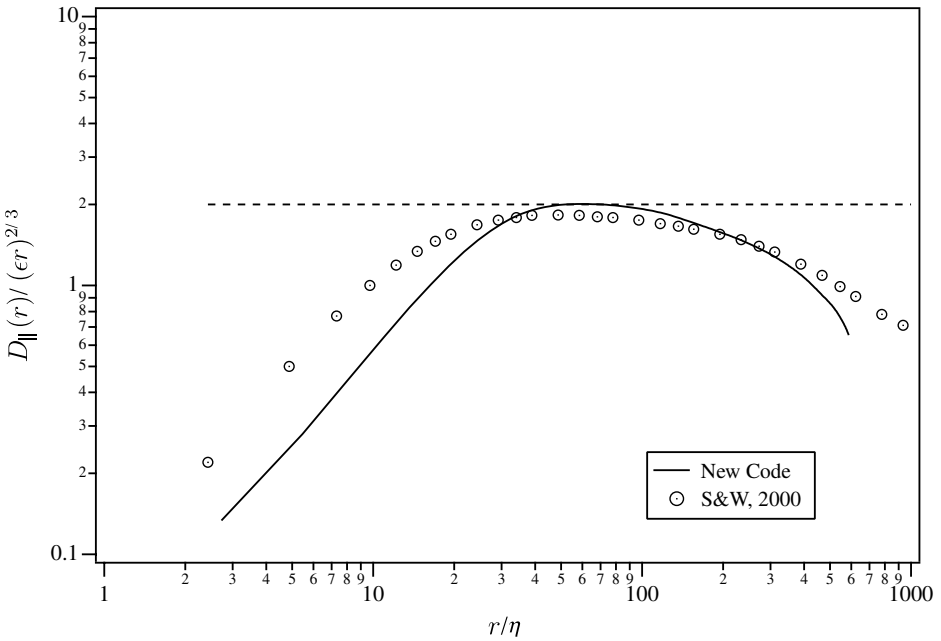


Fig. 8. Second-order structure function of the stream-wise velocity component. The horizontal line represents the Kolmogorov prediction.

Fig. 7 shows the probability density functions, PDF, of $\partial u/\partial x$ and $\partial v/\partial x$ and Fig. 8 shows the second-order structure functions of the stream-wise velocity component (the horizontal line shows the Kolmogorov prediction). The agreement for all three statistics is very good, especially given that the Reynolds numbers are not perfectly matched.

5. Conclusions

In summary, we have presented an alternative pseudospectral algorithm for computing homogeneous turbulent shear flow that eliminates many of the problems that result from the remeshing step. The new algorithm takes mean shear into consideration by defining a new spectral transform with a phase shift. We outlined the methodology for evaluating the three-dimensional spectral transform in $O(N^3 \ln N)$ operations. The parallel-

ization of the algorithm was discussed, and the results of the final code were compared with the results of the Rogallo algorithm with and without remeshing. The new algorithm was shown to be in perfect agreement with the Rogallo code for low values of the shear parameter. At higher shear parameters, the advantages of the present algorithm become evident. The saw-toothed pattern is not evident in the new code; moreover, the new code showed clear advantages over the Rogallo code with remeshing turned off. The algorithm was also shown to be in excellent quantitative agreement with published experimental data.

Acknowledgments

This work was supported in part by the National Science Foundation (NSF) through Grants PHY-0216406 and PHY-0554675. JCI was supported by a fellowship from the Fulbright Commission and EAFIT University in Colombia.

References

[1] J.L. Lumley, Similarity and the turbulent energy spectrum, Phys. Fluids 10 (1967) 855.
[2] N. Nakauchi, An application of the modified zero-fourth-cumulant approximation to homogeneous axisymmetric turbulence, J. Phy. Soc. Japan 53 (1984) 1682.
[3] S.B. Pope, Stochastic Lagrangian models of velocity in homogenous turbulent shear flow, Phys. Fluids 14 (2002) 1696–1702.
[4] M.M. Rogers, P. Moin, The structure of the vorticity field in homogeneous turbulent flows, J. Fluid Mech. 176 (1987) 33.
[5] Alain Pumir, Turbulence in homogeneous shear flow, Phys. Fluids 8 (1996) 3112.
[6] J. Schumacher, B. Eckhardt, On statistically stationary homogeneous shear turbulence, Europhys. Lett. 52 (2000) 627.
[7] J. Schumacher, K.R. Sreenivasan, P.-K. Yeung, Derivative moments in turbulent shear flow, Phys. Fluids 15 (2003) 84–90.
[8] X. Shen, Z. Warhaft, The anisotropy of small scale structure in high Reynolds number (R_λ) turbulent shear flow, Phys. Fluids 12 (2000) 2942.
[9] S.A. Orszag, G.S. Patterson, Numerical Simulation of Turbulence, Springer, New York, 1972.
[10] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods in Fluid Dynamics, Springer, New York, 1988.
[11] R.S. Rogallo, Numerical experiments in homogeneous turbulence, Technical Report 81315, NASA, 1981.
[12] M.J. Lee, J. Kim, P. Moin, Structure of turbulence at high shear rate, J. Fluid Mech. 216 (1990) 561–583.
[13] F. Baron, Macro-simulation tridimensionelle d'ecoulements turbulents cisailles, Ph.D. Thesis, Univ. Pierre et Marie Curie, 1982.
[14] U. Schumann, Algorithms for direct numerical simulation of shear-periodic turbulence, in: Soubbaramayer, J.P. Boujot (Eds.), Proceedings of the 9th International Conference on Numerical Methods in Fluid Dynamics, Lecture Notes in Physics, vol. 218, Springer, Berlin, 1985.
[15] U. Schumann, S.E. Elghobashi, T. Gertz, Direct simulation of stably stratified turbulent homogeneous shear flows, Notes on Numerical Fluid Mechanics, vol. 15, Vieweg, 1986, pp. 245–264.
[16] U. Schumann, The counter gradient heat flux in turbulent stratified flows, Nucl. Eng. Des. 100 (1987) 255–262.
[17] T. Gertz, U. Schumann, S.E. Elghobashi, Direct numerical simulation of stratified homogeneous turbulent shear flow, J. Fluid Mech. 200 (1989) 563–594.
[18] H.F. Champagne, G.V. Harris, S. Corrsin, Experiments on nearly homogeneous turbulent shear flow, J. Fluid Mech. 41 (1970) 81.
[19] G.V. Harris, A.J. Grahm, S. Corrsin, Further experiments in nearly homogeneous shear flow, J. Fluid Mech. 81 (1977) 657.
[20] S. Tavoularis, S. Corsin, Experiments in homogeneous sheared turbulence. Part 1, J. Fluid Mech. 104 (1981) 311–348.
[21] J. Schumacher, Derivative moments in stationary turbulent shear turbulence, J. Fluid Mech. 441 (2001) 109.
[22] J. Schumacher, Relation between shear parameter and Reynolds number in statistically stationary turbulent shear flows, Phys. Fluids 16 (2004) 3094–3102.
[23] Z. Warhaft, X. Shen, Some comments on the small scale structure of turbulence at high Reynolds number, Phys. Fluids 13 (2001) 1352.
[24] D. Yu, S.S. Girimaji, DNS of homogeneous shear turbulence revisited with the lattice Boltzmann method, J. Turbul. 6 (2005) 1–17.
[25] Anon., MPI: a message-passing interface standard, Int. J. High Performance Comput. Appl. 8 (1994) 159–416.
[26] Richard W. Johnson (Ed.), The Handbook of Fluid Dynamics, CRC Press, Boca Raton, FL, 1998 (Chapter: Mathematics of Fluid Mechanics).