Available online at www.sciencedirect.com
SCIENCE DIRECT®

Journal of Computational Physics 215 (2006) 153–171

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

A parallel implicit method for the direct numerical simulation of wall-bounded compressible turbulence

M. Pino Martín [a,*], Graham V. Candler [b]

[a] Department of Mechanical and Aerospace Engineering, Princeton University, Princeton, NJ 08544, United States
[b] Mechanical and Aerospace Engineering Department, University of Minnesota, Minneapolis, MN 55455, United States

Received 24 November 2004; received in revised form 24 October 2005; accepted 25 October 2005
Available online 15 December 2005

Abstract

A new second-order accurate implicit temporal numerical scheme for the direct numerical simulation of turbulent flows is presented. The formulation of the implicit method and the corresponding tunable parameters are introduced. The numerical simulation results are compared with the results given by explicit Runge–Kutta schemes, theoretical results, and published experimental and numerical data. An assessment of the accuracy and performance of the method to simulate turbulent flows is made for temporally decaying isotropic turbulence and subsonic and supersonic turbulent boundary layers. Whereas no significant advantage over typical explicit time integration methods are found for the incompressible flows; it is shown that the implicit scheme yields significant reduction in computer cost while assuring time-accurate solutions for compressible turbulence simulations.

Keywords: Implicit time integration; Direct numerical simulation; Wall-bounded turbulent flow

1. Introduction

The detailed simulation of compressible turbulence requires solving the conservation of mass, momentum and energy equations. For direct numerical simulations (DNS), all possible length scales and time scales must be resolved by the numerical method. Thus, DNS requires accurate representation of time-dependent wave propagation with high wavenumber (or high frequency) and small amplitude waves. Thus, numerical methods with minimal dissipation and dispersion properties are necessary to obtain accurate results.

In addition, gathering turbulence statistics requires large amounts of computing time because the simulations must be run on very large grids for many thousands of time steps. Generally, explicit Runge–Kutta (RK) methods are used to approximate the time derivative because they provide high-order accuracy, they have large stability limits relative to those of other explicit time integration techniques, and they are easy to program. In contrast, implicit methods permit much larger time steps with a loss in

[*] Corresponding author. Tel.: +1 609 258 7318; fax: +1 609 258 1993.
E-mail address: pmartin@princeton.edu (M.P. Martín).

0021-9991/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2005.10.017

accuracy. The cost of a RK solution increases as the stable time step decreases, while the cost of implicit methods depends on the allowable time step to obtain a given level of accuracy. Therefore, it is conceivable that for flows with very strict RK stability limit, we can use an implicit method with sufficient accuracy at less cost.

True implicit methods require the inversion of a large sparse matrix, demanding a great deal of computation and interprocessor communication. The Lower–Upper Symmetric Gausss–Seidel (LU-SGS) method of Yoon and Jameson [19] approximates the implicit problem, which eliminates the need for large block matrix inversions. Candler et al. [1] modify the LU-SGS method to make it easily parallelizable for inviscid flows. In the resulting Data-Parallel Lower–Upper Relaxation (DP-LUR) method, the Gauss–Seidel sweeps of the LU-SGS method are replaced with a series of pointwise relaxation steps. The DP-LUR formulation removes all data dependencies, making the relaxation steps perfectly parallel. Wright et al. [18] extend the DP-LUR formulation to viscous flows.

In this paper, we assess and develop an implicit method for the DNS of wall-bounded compressible turbulent flows. In these flows, the stable time step is determined by the wall-normal grid spacing and the speed of sound near the wall. This results in an extremely small stable time step for RK methods, making simulations very costly. On the other hand, the implicit method can take very large stable time steps, but with less accuracy and potentially more cost per time step. Thus, the key question is whether sufficiently accurate simulations can be performed with the implicit method at less cost than the Runge–Kutta method.

In the following sections, two formulations of the new implicit method are introduced, one form being more suitable for DNS of supersonic wall-bounded flows. The corresponding tunable parameters are also presented. The performance for both methods is assessed by comparing the results given by the implicit methods and those given by the well-established Runge–Kutta method. The dispersion and dissipation properties of the schemes, as well as the time accuracy of the solutions are monitored. In turn, the parameter space to achieve accurate solutions is determined. The time integration methods are described in Section 2 and their efficient implementation in parallel computers is addressed in Section 3. In Section 4, we introduce the accuracy indices for comparing the explicit and implicit method results. In Sections 5–8, we study the performance of the implicit methods on test problems. These include a one-dimensional unsteady test problem, temporally decaying isotropic turbulence, and turbulent boundary layers. Conclusions are given in Section 9.

2. Time integration

The equations governing compressible fluid motion can be written as

$$\frac{\partial V}{\partial t} + \frac{\partial \mathscr{F}}{\partial x} + \frac{\partial \mathscr{G}}{\partial y} + \frac{\partial \mathscr{H}}{\partial z} = 0, \tag{1}$$

where V is the vector of conserved quantities and $\mathscr{F}$, $\mathscr{G}$, and $\mathscr{H}$ are the flux vectors in the three spatial directions. With the spatial and temporal discretizations decoupled, Eq. (1) can be viewed as an ordinary differential equation in time:

$$\frac{\mathrm{d}U}{\mathrm{d}t} = \mathscr{R}[U], \tag{2}$$

where U is the solution of Eq. (2) and

$$\mathscr{R}[U] = -\left(\frac{\partial \mathscr{F}}{\partial x} + \frac{\partial \mathscr{G}}{\partial y} + \frac{\partial \mathscr{H}}{\partial z}\right). \tag{3}$$

The numerical stability of a method imposes a maximum stable advancing time step. This time step is based on the length of time that it takes for information to travel one grid spacing. Thus, the time step depends on both the fluid dynamics and the grid spacing. For the one-dimensional inviscid equations, the maximum stable time step is

$$\Delta t_{\max} = \frac{\Delta x}{|u| + a}, \tag{4}$$

where Δx is the grid spacing, u is the fluid velocity and a is the speed of sound. For viscous flows, the maximum stable time step is the minimum of the convective time step, Eq. (4), and the viscous time step, Δt_{v}, which is given by

$$\Delta t_{\mathrm{v}} = \frac{\rho}{\mu \Delta x^2}, \tag{5}$$

In practice, some fraction of $\Delta t_{\max}$ is used, and the actual time step is given by

$$\Delta t = \mathrm{CFL}\, \Delta t_{\max}, \tag{6}$$

where CFL is the Courant–Friedrichs–Lewy number.

The accurate computation of turbulent flows requires time advancement methods that resolve all relevant time scales. Popular explicit time-stepping methods include the family of Runge–Kutta methods. These are briefly summarized below for comparison against the implicit time integration schemes, which are also presented in this section.

2.1. Classical Runge–Kutta schemes

Runge–Kutta methods propagate the solution over a time interval by combining the information from several pseudo time steps and then using the information to match a Taylor series expansion up to a given order of accuracy. For example, given the solution at time n, the standard fourth-order accurate Runge–Kutta (RK4 hereafter) method is given by

$$\begin{aligned} U^{(1)} &= U^n + \frac{\Delta t}{2}\mathscr{R}[U^n], \\ U^{(2)} &= U^n + \frac{\Delta t}{2}\mathscr{R}[U^{(1)}], \\ U^{(3)} &= U^n + \Delta t \mathscr{R}[U^{(2)}], \\ U^{n+1} &= U^n + \frac{\Delta t}{6}\left(\mathscr{R}[U^n] + 2\mathscr{R}[U^{(1)}] + 2\mathscr{R}[U^{(2)}] + \mathscr{R}[U^{(3)}]\right). \end{aligned} \tag{7}$$

2.2. Second-order accurate DP-LUR (DP2) method

The DP-LUR method [1,18] is first-order accurate. Here, we extend it to second-order accuracy. The implicit form of the equations, using a second-order accurate discretization of the temporal derivative, is

$$\frac{3U^{n+1} - 4U^n + U^{n-1}}{2\Delta t} + \frac{\partial \mathscr{F}^{n+1}}{\partial x} + \frac{\partial \mathscr{G}^{n+1}}{\partial y} + \frac{\partial \mathscr{H}^{n+1}}{\partial z} = 0. \tag{8}$$

The flux vectors can be written in terms of the convective and viscous parts

$$\mathscr{F} = F + F_v, \quad \mathscr{G} = G + G_v, \quad \mathscr{H} = H + H_v. \tag{9}$$

Following the DP-LUR derivation, and focusing on the inviscid problem for now, we linearize the flux vector using

$$F^{n+1} \simeq F^n + \left(\frac{\partial F}{\partial U}\right)^n (U^{n+1} - U^n) = F^n + A^n \delta U^n, \tag{10}$$

$$G^{n+1} \simeq G^n + B^n \delta U^n, \tag{11}$$

$$H^{n+1} \simeq H^n + C^n \delta U^n. \tag{12}$$

We then split the Jacobian matrices using the diagonal matrix. For example,

$$A = S\Lambda S^{-1} = A_+ + A_-, \tag{13}$$

where

$$A_+ = S\Lambda_+ S^{-1}, \quad A_- = S\Lambda_- S^{-1}, \tag{14}$$

and the diagonal matrix Λ contains the positive (Λ^+) and negative (Λ^-) traveling characteristic waves of the flow. Thus, the spatial derivatives can be written as

$$\frac{\partial F^{n+1}}{\partial x} \simeq \frac{\partial F^n}{\partial x} + A^n \frac{\partial(\delta U^n)}{\partial x} = \frac{\partial F^n}{\partial x} + \frac{1}{\Delta x}(A_+ \nabla x + A_- \Delta x)^n \delta U^n, \tag{15}$$

where ∇ and Δ are spatial difference operators chosen to correctly represent the direction of information propagation determined by the sign of the split Jacobians. Namely,

$$\nabla_x \delta U = \delta U_{i,j,k} - \delta U_{i-1,j,k}, \quad \Delta_x \delta U = \delta U_{i+1,j,k} - \delta U_{i,j,k}. \tag{16}$$

We estimate the future vector U iteratively such that

$$U^{n+1} = U^p + U^{p+1} - U^p = U^p + \delta U^p. \tag{17}$$

In Eq. (17), we identify inner and outer iteration processes. The n superscript represents the temporal discretization, in which we solve for U^{n+1} with an “outer iteration” process. The p superscript represents the implicit iteration index or “inner iteration” process, in which $\delta U^p \to 0$ so that $U^p \to U^{n+1}$. The implicit iteration loop is initiated at $p = 0$ with $U^p = U^n$. The iteration process to converge δU^p to zero is described as follows. Eq. (8) is approximated as

$$\begin{aligned}\frac{3}{2}(U^p + \delta U^p) - 2U^n + \frac{1}{2}U^{n-1} = {} & -\Delta t\left(\frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} + \frac{\partial H}{\partial z}\right)^p \\ & - \Delta t\left[\frac{1}{\Delta x}(A_+\nabla x + A_-\Delta x) + \frac{1}{\Delta y}(B_+\nabla y + B_-\Delta y) + \frac{1}{\Delta z}(C_+\nabla z + C_-\Delta z)\right]^p \delta U^p\end{aligned} \tag{18}$$

Additionally, the Jacobians can be approximated as [19]

$$A_+ = \frac{1}{2}(A + \Gamma_A I), \quad A_- = \frac{1}{2}(A - \Gamma_A I), \tag{19}$$

where Γ_A is the spectral radius of the Jacobian A, given by the magnitude of the largest eigenvalue $|u| + a$. Collecting the diagonal terms on the left-hand side, we find

$$\begin{aligned}\left[\frac{3}{2}I + \Delta t\left(\frac{\Gamma_A}{\Delta x} + \frac{\Gamma_B}{\Delta y} + \frac{\Gamma_C}{\Delta z}\right)^p I\right]\delta U^p = {} & -\left(\frac{3}{2}U^p - 2U^n + \frac{1}{2}U^{n-1}\right) - \Delta t\left(\frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} + \frac{\partial H}{\partial z}\right)^p \\ & - \frac{\Delta t}{\Delta x}\left(A_-\delta U_{i+1,j,k} - A_+\delta U_{i-1,j,k}\right)^p \\ & - \frac{\Delta t}{\Delta y}\left(B_-\delta U_{i,j+1,k} - B_+\delta U_{i,j-1,k}\right)^p \\ & - \frac{\Delta t}{\Delta z}\left(C_-\delta U_{i,j,k+1} - C_+\delta U_{i,j,k-1}\right)^p.\end{aligned} \tag{20}$$

The expression for δU^p in Eq. (20) is solved in two steps. First, the off-diagonal terms are neglected and the right-hand side is divided by the diagonal operator to obtain

$$\delta U^{(0)} = -\left[\frac{3}{2}I + \Delta t\left(\frac{\Gamma_A}{\Delta x} + \frac{\Gamma_B}{\Delta y} + \frac{\Gamma_C}{\Delta z}\right)^p I\right]^{-1} \times \left[\left(\frac{3}{2}U^p - 2U^n + \frac{1}{2}U^{n-1}\right) + \Delta t\left(\frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} + \frac{\partial H}{\partial z}\right)^p\right]. \tag{21}$$

The second step consists of a series of $m_{\max}$ relaxation steps for $m = 1, m_{\max}$,

$$\left[\frac{3}{2}I + \Delta t\left(\frac{\Gamma_A}{\Delta x} + \frac{\Gamma_B}{\Delta y} + \frac{\Gamma_C}{\Delta z}\right)^p I\right]\delta U^m = -\left(\frac{3}{2}U^p - 2U^n + \frac{1}{2}U^{n-1}\right) - \Delta t\left(\frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} + \frac{\partial H}{\partial z}\right)^p - \frac{\Delta t}{\Delta x}\left(A_-\delta U_{i+1,j,k} - A_+\delta U_{i-1,j,k}\right)^{m-1} - \frac{\Delta t}{\Delta y}\left(B_-\delta U_{i,j+1,k} - B_+\delta U_{i,j-1,k}\right)^{m-1} - \frac{\Delta t}{\Delta z}\left(C_-\delta U_{i,j,k+1} - C_+\delta U_{i,j,k-1}\right)^{m-1}. \tag{22}$$

Then, $\delta U^p = \delta U^{m_{\max}}$. In practice, δU^p is converged to zero within a small error tolerance, ϵ. During each p iteration step, the condition $\delta U^p \leqslant \epsilon$ is checked for convergence. If convergence is not achieved, $U^{p+1} = U^p + \delta U^p$ and we advance the implicit loop. When convergence is achieved, we exit the implicit iteration loop, we let $U^{n+1} = U^p$, and we advance the outer loop.

The generalization of Eq. (22) to viscous flows involves replacing the inviscid fluxes by the full fluxes, Eq. (9), as well as modifying the spectral radii to include the maximum eigenvalue of the flux vector Jacobians [16] such that

$$\Gamma_{Av} = \Gamma_A + 2\frac{\gamma\mu}{Pr\rho\Delta x}, \quad \Gamma_{Bv} = \Gamma_B + 2\frac{\gamma\mu}{Pr\rho\Delta y}, \quad \Gamma_{Cv} = \Gamma_C + 2\frac{\gamma\mu}{Pr\rho\Delta z}, \tag{23}$$

where γ is the ratio of specific heats, μ is the viscosity, and Pr is the Prandtl number.

2.3. Modified DP2 (DP2M) scheme

As it is demonstrated in the later sections of the paper, the original DP2 method leads to dispersive errors under certain conditions when applied to problems including time dependent wave propagation phenomena. To minimize dispersive errors, we consider a modified version of the implicit time integration given in Eq. (8). Namely,

$$\frac{3U^{n+1} - 4U^n + U^{n-1}}{2\Delta t} + \frac{1}{2}\left(\frac{\partial\mathscr{F}^n}{\partial x} + \frac{\partial\mathscr{G}^n}{\partial y} + \frac{\partial\mathscr{H}^n}{\partial z}\right) + \frac{1}{2}\left(\frac{\partial\mathscr{F}^{n+1}}{\partial x} + \frac{\partial\mathscr{G}^{n+1}}{\partial y} + \frac{\partial\mathscr{H}^{n+1}}{\partial z}\right) = 0. \tag{24}$$

As it is shown below, the more diffusive character of the formulation in Eq. (24) relative to that of Eq. (8) results in reduced dispersive errors. Thus, the modified version of Eq. (22) is given by

$$\left[\frac{3}{2}I + \frac{\Delta t}{2}\left(\frac{\rho_A}{\Delta x} + \frac{\rho_B}{\Delta y} + \frac{\rho_C}{\Delta z}\right)^p I\right]\delta U^m = -\left(\frac{3}{2}U^p - 2U^n + \frac{1}{2}U^{n-1}\right) - \Delta t\left(\frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} + \frac{\partial H}{\partial z}\right)^p - \frac{\Delta t}{2\Delta x} \times \left(A_-\delta U_{i+1,j,k} - A_+\delta U_{i-1,j,k}\right)^{m-1} - \frac{\Delta t}{2\Delta y}\left(B_-\delta U_{i,j+1,k} - B_+\delta U_{i,j-1,k}\right)^{m-1} - \frac{\Delta t}{2\Delta z}\left(C_-\delta U_{i,j,k+1} - C_+\delta U_{i,j,k-1}\right)^{m-1}. \tag{25}$$

2.4. Tunable parameters for DP2 and DP2M

There are four tunable parameters for the implicit methods: the maximum number of relaxation steps $m_{\max}$, the maximum number of implicit steps $p_{\max}$, the error tolerance for the convergence of the implicit loop ϵ, and the CFL number, which is a tunable parameter for all time integration methods.

The relaxation loop arises from lagging the off-diagonal terms in the expression for δU^p in Eq. (20). Provided that the solution changes slowly, which must be true to achieve convergence and time accuracy, a relatively small number of m steps are required. We chose $m_{\max} = 4$. In turn, if this is a bad choice, it will be difficult to achieve convergence and the implicit method will be costly.

The maximum number of implicit steps will be determined at convergence, we let $p_{\max} = 20$ and expect that convergence is achieved for $p < p_{\max}$. If $p = p_{\max}$, then either convergence is not achieved and the solution is inaccurate, or the method is costly. In the code implementation, we enforce that if $p > p_{\max}$, an error message is produced.

The error tolerance determines the accuracy of the solution and its value must be chosen separately from one problem to another, in the same way as the CFL number is chosen for any explicit or implicit method. In Sections 5–8, we determine the range of acceptable ϵ and CFL values to achieve accurate solutions. In turn, we determine the computational cost of the implicit method solutions relative to those given by explicit schemes. A combination of these parameters that provides a sufficiently accurate solution with a reduced computational cost per solution integration time (relative to explicit methods) is desirable.

3. Parallel performance of the implicit methods

Many aspects of the DP2 and DP2M formulations make the implementation efficient and inexpensive on parallel distributed calculations. First, the calculation of $\delta U^{(0)}$ in Eq. (21) is trivial since the inversion of the diagonal matrix

$$\left[\frac{3}{2}I + \Delta t\left(\frac{\rho_A}{\Delta x} + \frac{\rho_B}{\Delta y} + \frac{\rho_C}{\Delta z}\right)I\right]$$

is a simple scalar inversion. Second, the evaluation of the spatial derivatives is the most costly part in Eq. (21). This evaluation is also necessary for other methods such as an explicit Runge–Kutta scheme. Thus, Eq. (21) requires very little additional computation compared to a one-stage Runge–Kutta method. Third, the spatial derivatives in Eq. (22) do not need to be re-evaluated every m step, since they are calculated in Eq. (21). Fourth, the addition of the off-diagonal implicit terms in Eq. (22) is relatively inexpensive because these quantities are computed and used immediately, which eliminates the need to store them. Thus, for small values of $m_{\max}$, the cost (both in terms of computer time and memory) of evaluating the implicit method for one value of p is similar to a single stage of a Runge–Kutta method.

We implement the method by partitioning the problem across the processors in two grid dimensions (usually the streamwise and spanwise directions). We communicate the necessary boundary data between processors with asynchronous MPI communication protocols, which allows the cost to be completely hidden for typical problems. In practice, we achieve a factor of 96 speed-up on 128 processors of a cluster with 2.2 GHz Xeons connected with Myrinet.

4. Parameters for comparison

4.1. Convergence index

In Eq. (20), δU^p is converged to zero within a small tolerance in order to minimize the global solution error. To do so, we use the following convergence criterion:

$$\frac{1}{N_x N_y N_z}\sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\sum_{k=1}^{N_z}\left|\frac{\delta\rho^p}{\rho_0}\right|_{i,j,k} \leqslant \epsilon, \tag{26}$$

where N_i is the total number of grid points in the ith direction, ρ is the density, ρ_0 is a normalization factor given by a reference value of ρ, and ϵ is the user-specified convergence tolerance. Thus, Eq. (26) represents the per-point average L_1 norm of the density convergence error. When the absolute average change in the continuity equation is less than or equal to the prescribed tolerance, the current δU^p is accepted as an update to the solution and Eq. (20) holds. If we make the convergence tolerance more stringent, the solution requires a larger number of p iterations; if the tolerance is too large, the resulting solution may contain errors due to the lack of convergence to U^{n+1}. An assessment of the tolerance values and resulting accuracy is performed in Sections 5–8 for the different test problems.

4.2. Computational-cost index

The number of evaluations of Eq. (3) can be used as an index of the computational cost for the different methods. Note that the number of evaluations increases with (1) the number of time-steps to reach a given physical time, (2) increasing number of Runge–Kutta stages per time step, and (3) increasing stringency of the convergence criteria for the implicit method. An assessment of the cost index for the implicit method relative to that of the RK method, IC, is performed in Sections 5–8.

4.3. Accuracy index

The validation of accuracy for the simulation data can be done by direct comparison against the exact solution, well-established simulation or experimental data, and/or a theoretical solution (if available); or to grid converged results (if computationally possible) using a well-established method. As an index of accuracy we use

$$\mathrm{IE} = \left[\frac{\sum (u_{\mathrm{cal}} - u_0)^2}{\sum u_0^2}\right]^{1/2} \times 100, \tag{27}$$

where u_{cal} is the value of the solution that we want to assess, and u_0 is the reference data available for comparison.

5. Propagation of a one-dimensional disturbance

To test the accuracy of the implicit methods, we assess the dispersion and dissipation errors of the scheme when they are applied to the time-dependent long-range propagation of a linear one-dimensional disturbance. We compare the results given by the DP2 and RK4 schemes. For this problem, the explicit solutions do not have a strict stability limit. Therefore, computational savings using the implicit methods are not expected.

The governing equation is the linear wave equation

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0. \tag{28}$$

The spatial derivatives are approximated using a sixth-order accurate Padé scheme [10] with a time step $\Delta t = \alpha\mathrm{CFL}$, where α is the convective velocity of the wave and is unity according to Eq. (28). We initialize the disturbance at $t = 0$ with a sine wave

$$u(x) = \sin\left(\frac{2\pi x}{a\Delta x}\right), \tag{29}$$

where $a\Delta x$ is the wavelength of the disturbance and a represents the number of grid points per wavelength (PPW). We use two tests with 16 and 8 PPW. To emphasize numerical errors, we propagate the disturbance over 100 and 50 times for $a = 16$ and 8, respectively.

Fig. 1 shows the convergence of the DP2 solution with varying CFL number. For both test cases with CFL = 0.05, the error relative to the exact solution is about 8%. The effect of the error tolerance for DP2 is shown in Fig. 2. The solution converges with $\epsilon = 10^{-6}$ for CFL = 0.25 and with $\epsilon = 10^{-4}$ for CFL = 0.01 and 0.05. However, the solution is accurate only for CFL = 0.05 where the error is within 9%. For non-converged solutions, dispersive errors increase with increasing CFL number. Fig. 3 shows the comparison of the converged solutions with RK4 and DP2. The RK4 simulation converges with CFL = 0.25, where the error is less than 1%.

For DP2M, Fig. 4 shows the effect of ϵ at various CFL numbers. The results are similar to those shown in Fig. 2. However, the dispersive errors are significantly reduced. Table 1 lists the error and the cost of the DP2M simulations relative to that of the converged RK4 solution for PPW = 16. The accurate DP2M solution with CFL = 0.05 and $\epsilon = 10^{-6}$ is two and a half times more costly than obtaining the converged RK4 solution. For this problem, the relative cost of DP2 and DP2M is similar.

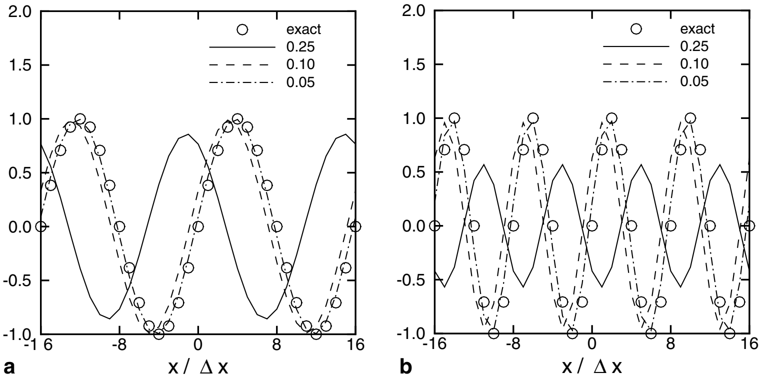

Fig. 1. Effect of CFL on DP2 with $\epsilon = 10^{-6}$. One-dimensional propagating sine wave with (a) 16 grid points per wavelength and (b) 8 grid points per wavelength.

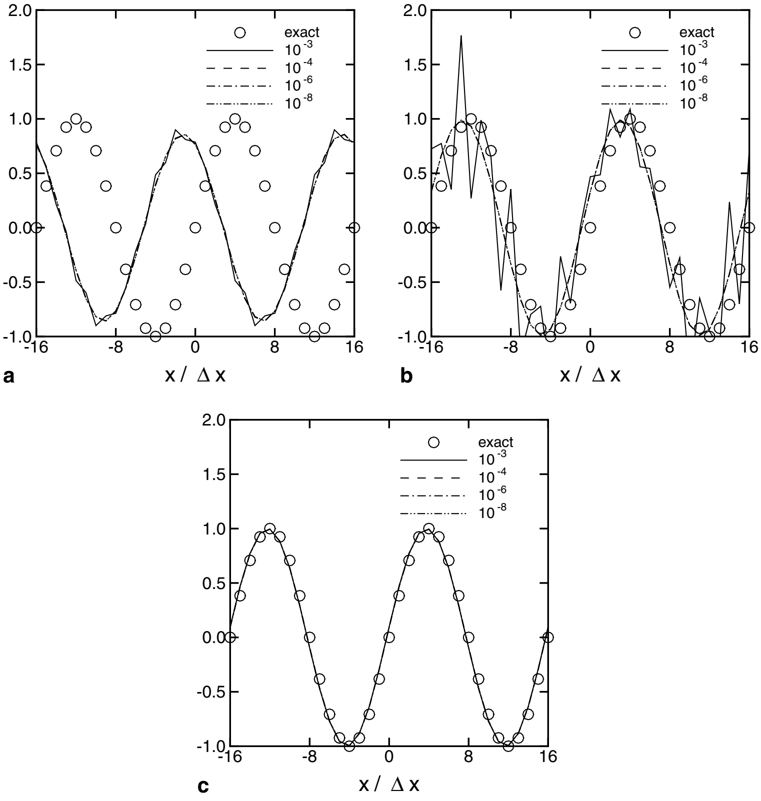

Fig. 2. Effect of error tolerance on DP2. One-dimensional propagating sine wave with 16 grid points per wavelength at CFL (a) 0.25, (b) 0.1, and (c) 0.05.

6. Isotropic turbulence

In this section, we consider decaying isotropic turbulence to test the implicit methods. The governing equations are the compressible Navier–Stokes equations. The spatial derivatives are approximated using a sixth-order accurate Padé scheme. The time step is given by

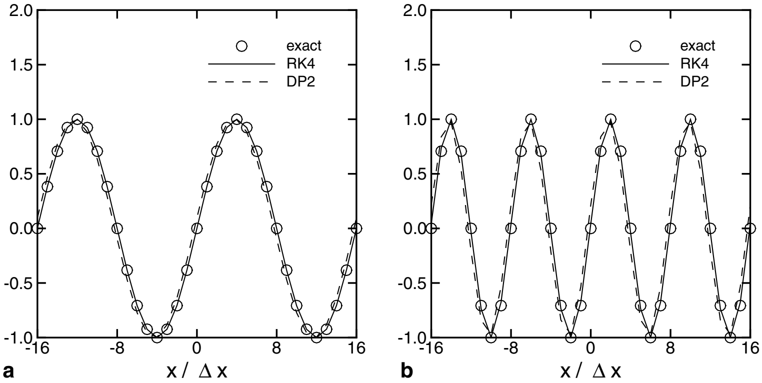

Fig. 3. Comparison of RK4 at CFL = 0.25 and DP2 at CFL = 0.05 and $\epsilon = 10^{-4}$. One-dimensional propagating sine wave with (a) 16 grid points per wavelength and (b) 8 grid points per wavelength.

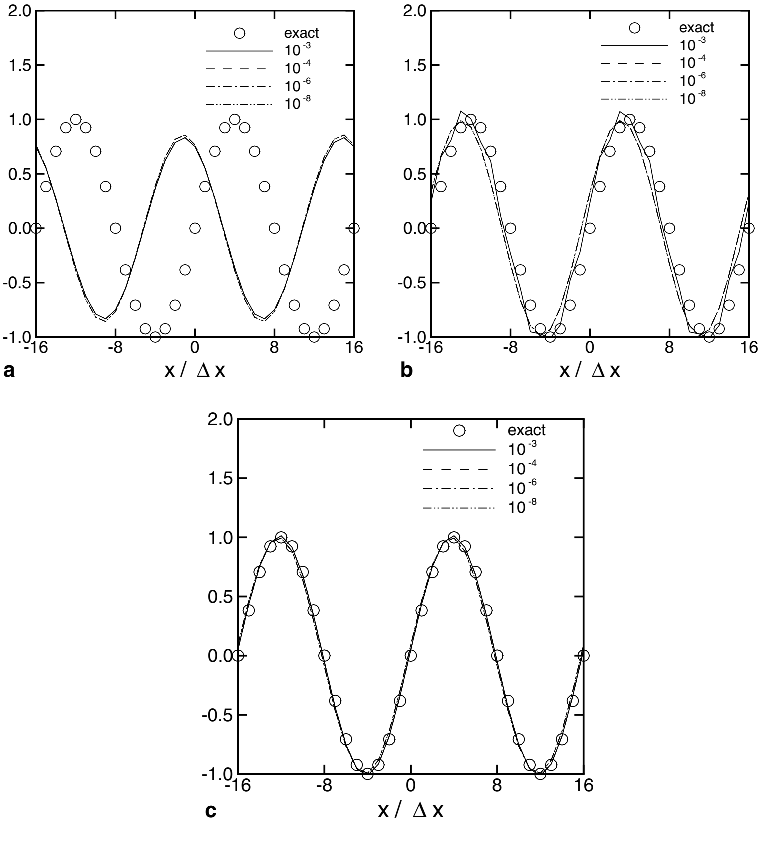

Fig. 4. Effect of error tolerance on DP2M. One-dimensional propagating sine wave with 16 grid points per wavelength at CFL (a) 0.25, (b) 0.1, and (c) 0.05.

$$\Delta t = \mathrm{CFL}\min\left(\frac{\Delta x}{|u|+a}+\frac{\Delta y}{|v|+a}+\frac{\Delta z}{|w|+a}+\frac{\rho}{\mu}\left(\frac{1}{\Delta x^2}+\frac{1}{\Delta y^2}+\frac{1}{\Delta z^2}\right)\right), \tag{30}$$

where Δx, Δy and Δz are the local grid spacings in each coordinate direction, ρ is the density, and μ is the viscosity. The initial flow conditions are $M_t = q/a = 0.3$ and $Re_\lambda = \rho u_{\mathrm{RMS}}\lambda/\mu = 35$ with 64^3 grid points, where $q = \langle u_i' u_i' \rangle$ and λ is the Taylor microscale. We simulate the decay of isotropic turbulence for six

Table 1
Relative accuracy and cost index variation for DP2M depending on CFL number and error tolerance for the one-dimensional test problem with 16 PPW

CFL	ϵ	$\mathrm{IE}_{\mathrm{DP2M}}$ (%)	$\mathrm{IC}_{\mathrm{DP2M}}$
0.25	10^{-3}	$\geqslant 100$	0.75
0.25	10^{-4}	$\geqslant 100$	0.75
0.25	10^{-6}	$\geqslant 100$	1.25
0.25	10^{-8}	$\geqslant 100$	1.50
0.10	10^{-3}	23	1.88
0.10	10^{-4}	32	1.88
0.10	10^{-6}	32	2.50
0.05	10^{-3}	4	3.75
0.05	10^{-4}	4	2.50
0.05	10^{-6}	8	2.50

CFL = 0.25 for the RK4 reference with $\mathrm{IE}_{\mathrm{RK4}} = 0.16\%$.

non-dimensional time periods based on the initial large-eddy turn over time, $\tau_t = \lambda/u_{\mathrm{RMS}}$. The length of the simulation and the simulation parameters are representative of those used to study fundamental physical phenomena or developing turbulence modes in isotropic turbulence [9,11,12]. The numerical simulation code is the same as that used in the isotropic turbulence studies of Martín and Candler [11].

We compare the results given by the implicit and RK4 schemes, monitoring the temporal evolution of the turbulent kinetic energy. The global error in the solution is computed with

$$\mathrm{IE} = \frac{1}{N}\sum_{n=0}^{N}\left[\sqrt{\frac{(q_{\mathrm{implicit}} - q_{\mathrm{RK4}})^2}{(q_{\mathrm{RK4}})^2}}\right]^n \times 100, \tag{31}$$

where the sum over n represents the cumulative sum in time from $t = 0$ to $t = t_N$ in equally spaced time intervals. To further assess the dissipation, we consider the frequency distribution of the error, which is given by

$$\mathrm{IE}(k) = \sqrt{\frac{[E_{\mathrm{implicit}}(k) - E_{\mathrm{RK4}}(k)]^2}{[E_{\mathrm{RK4}}(k)]^2}}, \tag{32}$$

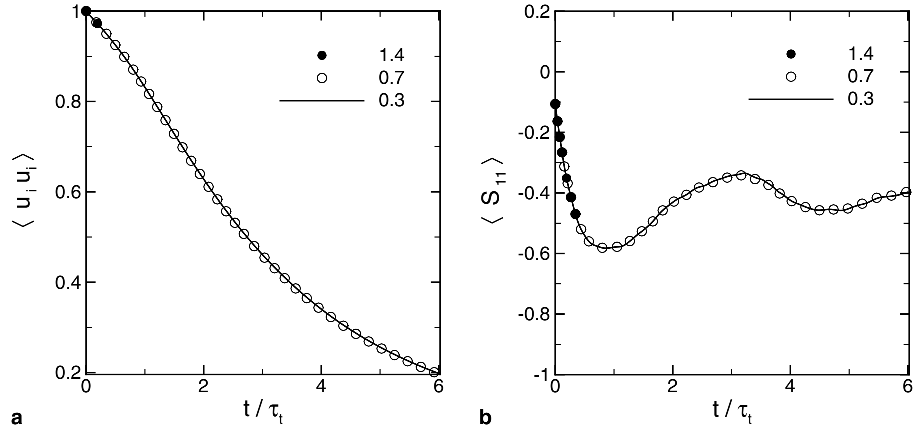

Fig. 5. Effect of CFL on RK4 for decaying isotropic turbulence at $Re_\lambda = 35$ and $M_t = 0.3$. Temporal evolution of the (a) turbulent kinetic energy and (b) skewness of velocity.

where $E(k)$ is the turbulent kinetic energy at wavenumber k.

Fig. 5 shows the effect of CFL on the stability of the RK4 method. We observe that the simulation is not stable (numerical instabilities give physically non-meaningful results and crash the simulation) for CFL = 1.4, while the solution is stable and converges for CFL = 0.7. For this flow and conditions, the stability limit for the RK4 scheme is reached with CFL = 0.7.

Exploring how varying the CFL number affects the DP2 and DP2M solutions, we found that CFL > 2.8 gives unstable results for this problem. Fig. 6 shows the effect of error tolerance on the implicit scheme solutions with CFL = 2.8 in comparison to the RK4 solution with CFL = 0.7. For the DP2 and DP2M, Fig. 6a and b show that the implicit methods reproduce the RK4 solution with ϵ up to 10^{-4}. The global error, IE, is roughly 2% and 1% for the DP2 and DP2M solutions, respectively. Fig. 6c and d show the frequency distribution of the error for the implicit schemes. Comparing these figures, we observe that for $\epsilon \leqslant 10^{-4}$, DP2M results in reduced error at all frequencies. For more stringent values of ϵ, both schemes are accurate, in that the error in is no greater than 2%. It should be noted that the purely dissipative region of the turbulent kinetic energy spectra is near wavenumber $k = 50$. Thus in this wavenumber range, the amount of turbulent kinetic energy is very small (six orders of magnitude below the energetic content in the large-scale/low-frequency range) and IE(k) is not meaningful.

Table 2 lists the error and cost of the implicit simulations relative to that of the RK4 solution with CFL = 0.7. The implicit time-accurate solutions are two to three times faster than those with RK4.

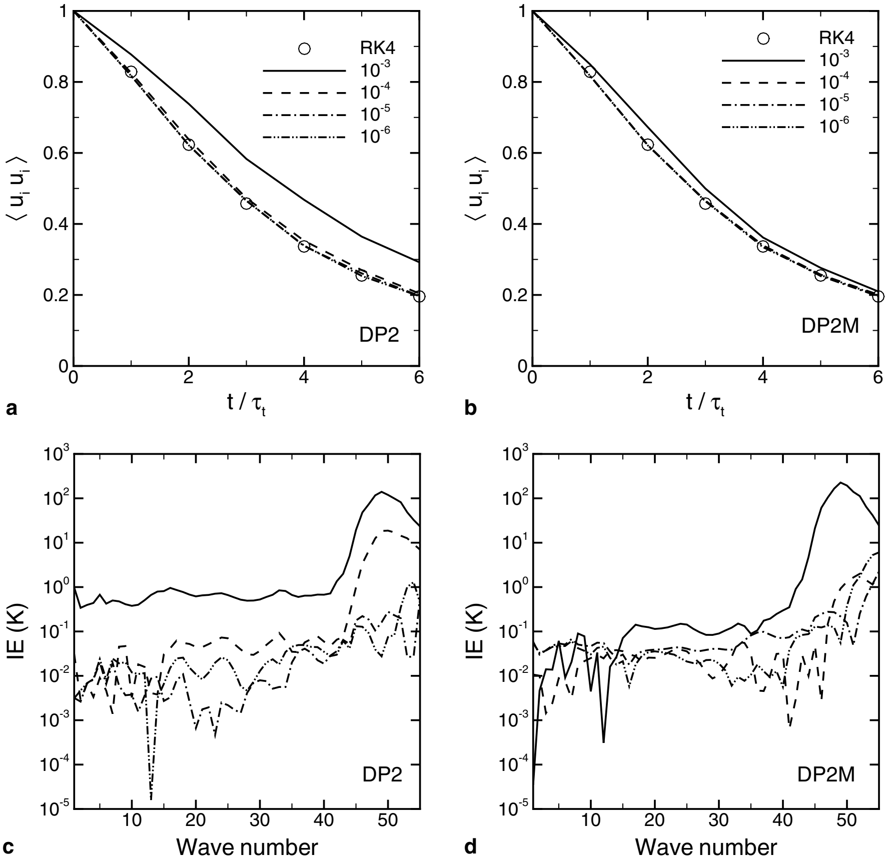

Fig. 6. Effect of error tolerance on the implicit methods for decaying isotropic turbulence with CFL = 2.8 at $Re_\lambda = 35$ and $M_t = 0.3$. Temporal evolution of the (a, b) normalized kinetic energy decay; (c, d) frequency distribution of the error, IE(k), at $t/\tau_t = 6.0$. RK4 with CFL = 0.7.

Table 2
Relative accuracy and cost index variation for implicit simulations depending on CFL number and error tolerance for decaying isotropic turbulence at $Re_\lambda = 35$ and $M_t = 0.3$

CFL	ϵ	$\mathrm{IE}_{\mathrm{DP2}}$ (%)	$\mathrm{IC}_{\mathrm{DP2}}$	$\mathrm{IE}_{\mathrm{DP2M}}$ (%)	$\mathrm{IC}_{\mathrm{DP2M}}$
0.7	10^{-3}	6.32	0.42	0.84	0.64
0.7	10^{-4}	0.97	0.59	0.43	0.68
0.7	10^{-5}	0.29	1.05	0.23	0.93
0.7	10^{-6}	0.23	1.78	0.25	1.46
1.4	10^{-3}	6.51	0.25	1.55	0.27
1.4	10^{-4}	1.60	0.39	0.74	0.48
1.4	10^{-5}	0.49	0.78	0.41	0.85
1.4	10^{-6}	0.40	1.43	0.44	1.44
2.8	10^{-3}	16.72	0.16	5.00	0.21
2.8	10^{-4}	2.06	0.37	1.00	0.37
2.8	10^{-5}	0.96	0.62	0.87	2.20
2.8	10^{-6}	0.89	1.19	0.87	2.55

CFL = 0.7 for the RK4 reference solution.

7. Compressible turbulent boundary layer

In this section, we address the questions of (1) whether the implicit methods give accurate solutions for turbulent boundary layers with increased values of CFL, relative to a RK3 method, which is sufficiently accurate for these simulations, and (2) whether the implicit methods are cost effective. The simulations of turbulent boundary layers are computationally intensive. Thus, performing parametric studies is very costly. Because of this, we cannot perform simulations for the entire range of tunable parameters, and we are guided by the isotropic simulation results.

The time step for this problem is given by Eq. (30). We consider the experimental conditions of Debiéve [2,3,7] with $M = 2.32$ and $Re_\theta = 4000$ and Eléna [4,5] with $M = 2.32$ and $Re_\theta = 4700$. The spatial derivatives are approximated using a third-order accurate, bandwidth optimized WENO scheme [17,14]. The time step is defined as in Eq. (30). The number of grid points and domain size are $328 \times 256 \times 120$ and $7.1\delta \times 2.1\delta \times 16.6\delta$ in the streamwise, spanwise, and wall-normal directions, respectively. The initial conditions are those of Martín [13] and we use rescaling boundary conditions [20] to provide the inflow. Thus, we simulate a fixed spatial location of a spatially developing boundary layer. After the initial transient, Re_θ is about 4600. We gather statistics for 70 non-dimensional time periods $\tau_t = \delta^*/U_e$, where δ^* is the displacement thickness and U_e is the velocity at the boundary layer edge. The spacing between fields is about $1.5\tau_t$. The results are validated against the experimental data and then used to assess the CPU time savings when using the implicit methods.

From the isotropic turbulence studies in the previous section, we find that $\epsilon = 10^{-6}$ is a conservative value for the error tolerance. Thus, we first consider the results using this value. Fig. 7 plots the mean flow profiles for the DNS and experimental data. Fig. 8a and b plot the normalized magnitude of velocity and temperature fluctuations, respectively, in comparison with the experiments of Debiéve. Fig. 9a and b show the magnitude of the streamwise and wall-normal velocity fluctuations in comparison to the experiments of Eléna, where the data are normalized using the edge and friction velocities respectively. The criteria for good accuracy is that the simulation data agree with the experimental data to within the scatter of the experimental data. Also accounting for the differences that are observed very closed to the wall in Fig. 9b, it should be noted that accessing the near-wall region ($z/\delta \leqslant 0.2$) is not possible experimentally. Thus, the experimental data are not accurate very close to the wall. The DNS data are within the scatter of the experimental data. Figs. 7–9 illustrate the overall good agreement among the experimental and DNS data with RK3 and DP2 at CFL = 1. For these simulations, the cost of DP2 relative to that of RK3 is $\mathrm{IC}_{\mathrm{DP2}}/\mathrm{IC}_{\mathrm{RK3}} = 1.15$.

For this problem, we find that the stability limit of the RK3 scheme is CFL = 1. Fig. 10 illustrates the effect of CFL for the simulations with the DP2 scheme. We find stable solutions up to CFL = 40. We have not performed simulations at higher CFL numbers. The accuracy of the simulations is good for CFL = 20. For

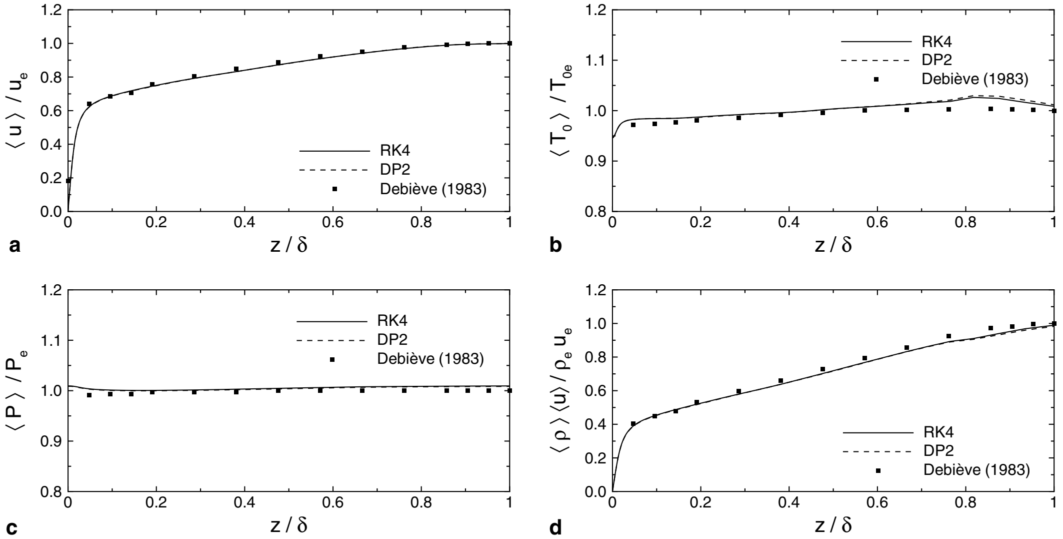

Fig. 7. RK3 and DP2 simulations of a turbulent boundary layer with CFL = 1 and comparison with experimental data [3,2,7]. Normalized values of mean: (a) velocity; (b) total temperature; (c) pressure; (d) momentum. Quantities are normalized using edge values.

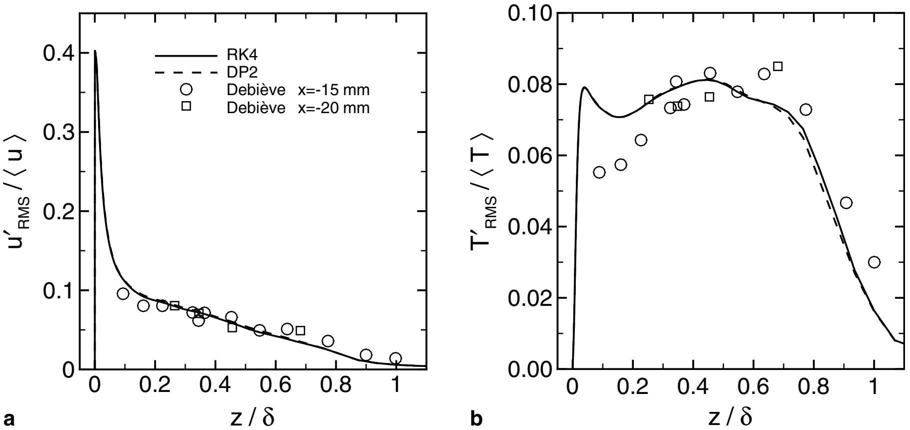

Fig. 8. RK3 and DP2 simulations of a turbulent boundary layer with CFL = 1 and comparison with experimental data [3,2,7]: (a) streamwise component of turbulent fluctuations; (b) turbulent temperature fluctuations. Quantities are normalized using mean flow variables.

CFL = 40 the solution departs slightly from the rest of the DNS and experimental data. The solution with DP2M at CFL = 20 is also shown. The statistics are similar for both DP2 and DP2M methods.

Fig. 11 plots the one-dimensional energy spectra near the wall ($z^+ = \rho_w u_\tau z / \mu_w = 8$, where u_τ is the friction velocity and subscript w indicates wall values) for the DP2 and DP2M solutions with CFL = 20 and for the RK3 simulation with CFL = 1. We observe no significant difference between the numerical results and no dispersive errors.

Next, we consider the effect of varying the error tolerance for the implicit methods with CFL = 5. Fig. 12a and b show the magnitude of streamwise turbulent intensity given by the DP2 and DP2M methods, respectively. We observe differences in the data for the DP2 method. In particular the peak magnitude for the fluctuations is inaccurate for $\epsilon = 10^{-4}$. We observe no significant difference in the DP2M simulation data. Fig. 12b and c show the near-wall energy spectra. Consistent with the previous result, we observe a departure from the rest of the data in the large-scale (low-frequency) range of the spectrum for the DP2 method with $\epsilon = 10^{-4}$. No significant differences are observed for the DP2M method.

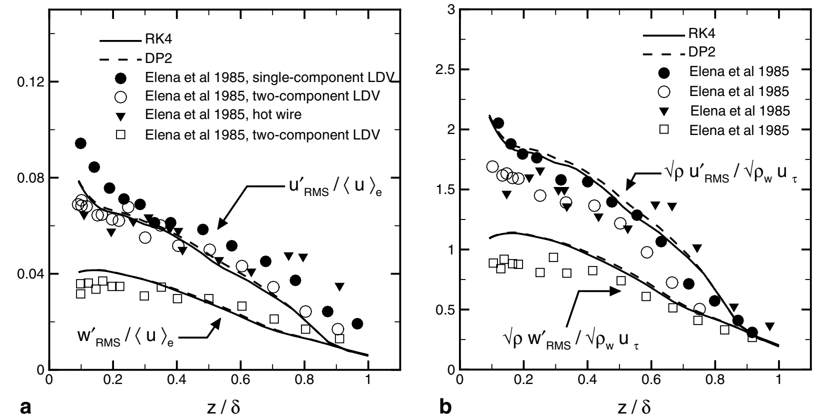

Fig. 9. RK3 and DP2 simulations of a turbulent boundary layer with CFL = 1 and comparison with experimental data [4,5]: (a) magnitude of velocity fluctuations normalized with edge velocity; (b) magnitude of velocity fluctuations normalized with Morkovin's scaling.

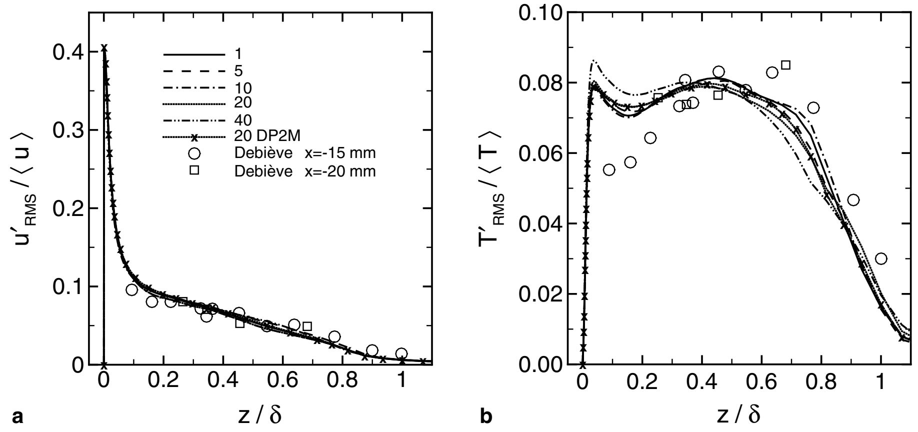

Fig. 10. Effect of CFL on DP2 solutions for the DNS of a turbulent boundary layer and comparison with experimental data [3,2,7]: (a) streamwise component of turbulent fluctuations; (b) turbulent temperature fluctuations. Quantities are normalized using mean flow variables.

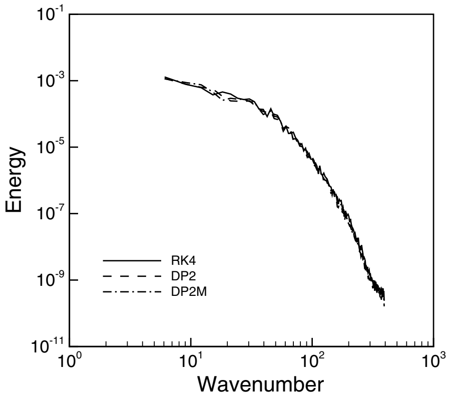

Fig. 11. Near-wall energy spectra ($z^+ = 8$) for the RK3 simulation with CFL = 1 and for the DP2 and DP2M solutions with CFL = 20.

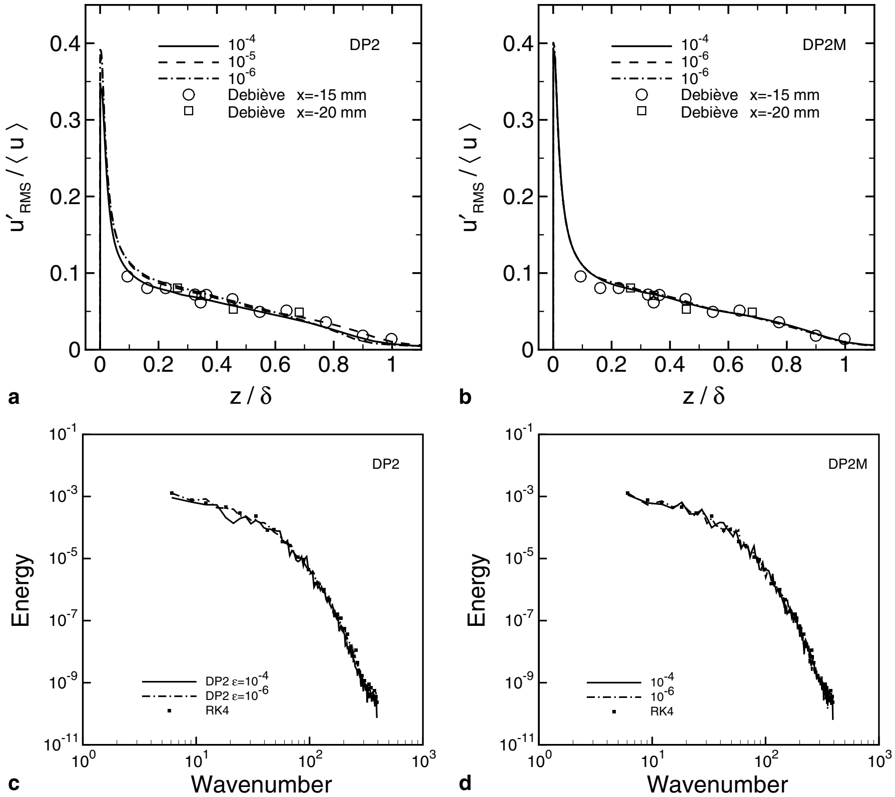

Fig. 12. Effect of ϵ on the DP2 and DP2M solutions with CFL = 5 for the DNS of a turbulent boundary layer and comparison with experimental data [3,2,7] and RK3 results with CFL = 1. (a, b) Streamwise component of turbulent fluctuations; (c, d) near-wall energy spectra ($z^+ = 8$).

We compare the results given by the implicit and RK3 schemes, monitoring the global error in the solution, which is computed with

$$\mathrm{IE} = \frac{1}{\mathrm{kbl}} \sum_{k=1}^{\mathrm{kbl}} \sqrt{\frac{(\mathrm{RMS}_{\mathrm{implicit}} - \mathrm{RMS}_{\mathrm{RK3}})^2}{\mathrm{RMS}^2_{\mathrm{RK3}}}} \times 100, \tag{33}$$

where the sum over k represents the summation of all data points within the boundary layer thickness, which is given by kbl. Since the error is found to be larger in T'_{RMS}, it is taken as the RMS quantity to measure the relative error. Fig. 13a and b plot IE for the DP2 and DP2M solutions varying CFL number and ϵ, respectively. The error increases with increasing both parameters, being roughly 4% for CFL = 40 and over 10% for $\epsilon > 10^{-5}$.

For $\epsilon = 10^{-6}$, Table 3 lists the cost of the DP2 simulations relative to the RK3 solution with CFL = 1. For accurate solutions, the DP2 and DP2M methods result in factors of about 17 and 6 reduction in computing time, respectively. Table 4 lists the cost for the DP2 and DP2M simulations varying ϵ. The cost of accurate solutions is comparable for both implicit solutions. The accuracy and cost of the DP2M solutions does not depend much on ϵ, where $\mathrm{IC}_{\mathrm{DP2M}}$ is about 3.25 to 4 times less costly than the RK3 calculations.

8. Incompressible turbulent boundary layer

We consider the cost of the implicit and explicit methods for an incompressible turbulent boundary layer. Because the simulations are computationally intensive, we only consider RK3 and DP2M results.

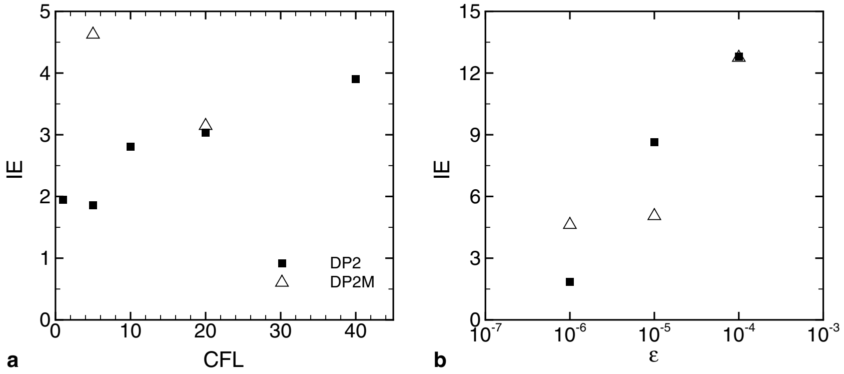

Fig. 13. Global error for the DP2 and DP2M with varying (a) CFL number and (b) error tolerance, ϵ.

Table 3
Relative cost index variation for the implicit methods depending on CFL number for the DNS of turbulent boundary layers using the experimental conditions of Debiéve et al. with $\epsilon = 10^{-6}$

CFL	$\mathrm{IC}_{\mathrm{DP2}}$	$\mathrm{IC}_{\mathrm{DP2M}}$
1	1.15	*
5	0.23	0.25
10	0.10	*
20	0.06	0.16
40	0.03	*

The reference RK3 simulation is with CFL = 1. The '*' indicates that we have not performed the simulation.

Table 4
Relative cost index variation for the implicit methods depending on error tolerance for the DNS of turbulent boundary layers using the experimental conditions of Debiéve et al. with CFL = 5

ϵ	$\mathrm{IC}_{\mathrm{DP2}}$	$\mathrm{IC}_{\mathrm{DP2M}}$
10^{-4}	0.18	0.33
10^{-5}	0.32	0.32
10^{-6}	0.23	0.25

The reference RK3 simulation is with CFL = 1.

The time step for this problem is given by Eq. (30). The simulation conditions are a $M = 0.3$ and $Re_\theta = 1750$, which are similar to those of Spalart [15]. The spatial derivatives are approximated using a third-order accurate, bandwidth optimized WENO scheme [17,14] with the adaption (shock-capturing) mechanism turned off. The time step is defined as in Eq. (30). The number of grid points and domain size are $384 \times 256 \times 116$ and $5.7\delta \times 1.4\delta \times 15\delta$ in the streamwise, spanwise, and wall-normal directions, respectively. The initial conditions are computed using the procedure of Martín [13]. We gather statistics for 138 non-dimensional time periods $\tau_t = \delta^*/U_e$. The spacing between fields is about $4.0\tau_t$. The results are validated against empirical data and then used to assess the CPU time savings when using the implicit methods.

Fig. 14a and b plot the velocity profile and skin friction for the DNS with RK3 and CFL = 0.8 against the theoretical predictions showing good agreement. For the same DNS simulation, Fig. 15a and b plot the turbulent intensity profiles in comparison to the DNS data of Spalart [15] and the experiments of Klebanoff [8] showing the good agreement. The DP2M simulations are unstable for CFL numbers greater than 2.5. Fig. 16 plots the turbulent intensities for the DP2M solutions with CFL = 2.5 and varying ϵ. Simulations with $\epsilon = 10^{-4}$ are unstable; otherwise, we find accurate results with smaller values of ϵ. The cost of the DP2M simulations relative to the RK3 solution is listed in Table 5. The DP2 simulations with CFL = 2.5 are roughly 25% more expensive than the RK3 solutions with CFL = 0.8.

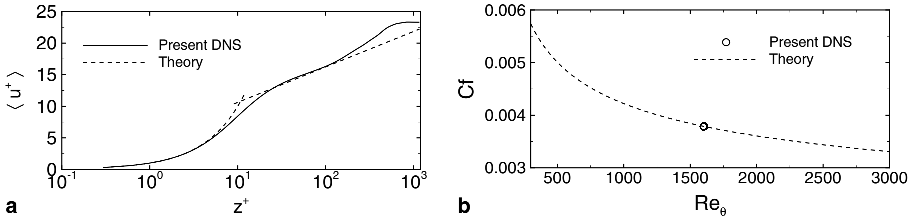

Fig. 14. RK3 simulation of a $M = 0.3$, $Re_\theta = 1600$ turbulent boundary layer with CFL = 0.8 and comparison with theory. (a) Mean velocity profile normalized with the friction velocity u_τ; and (b) skin friction coefficient.

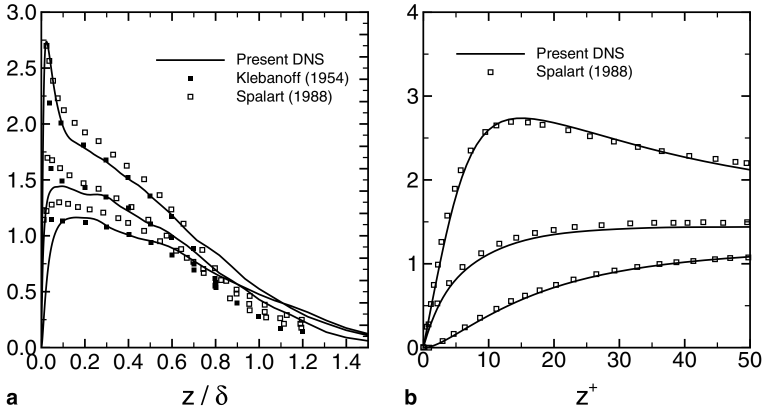

Fig. 15. RK3 simulation of an $M = 0.3$, $Re_\theta = 1600$ turbulent boundary layer with CFL = 0.8 and comparison with the DNS data of Spalart [15] with $Re_\theta = 1410$ and the experiments of Klebanoff [8] with $Re_\theta = 7500$. Upper curve u_{RMS}/u_τ; middle curve v_{RMS}/u_τ; lower curve w_{RMS}/u_τ.

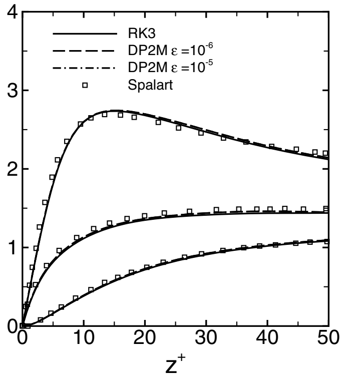

Fig. 16. Effect of ϵ on DP2M solution with CFL = 2.5 for the DNS of an incompressible turbulent boundary layer and comparison with the DNS data of Spalart [15] and the RK3 result with CFL = 0.8. Upper curve u_{RMS}/u_τ; middle curve v_{RMS}/u_τ; lower curve w_{RMS}/u_τ.

Table 5

Relative cost index variation for the DP2M method depending on error tolerance for the DNS of an incompressible turbulent boundary layer at $M = 0.3$ and $Re_\theta = 1600$ with CFL = 2.5

ϵ	$\mathrm{IC}_{\mathrm{DP2M}}$
10^{-4}	Unstable
10^{-5}	1.22
10^{-6}	1.29

The reference RK3 simulation is with CFL = 0.8.

9. Conclusions

We present two formulations of a second-order accurate, implicit method for the simulation of turbulent flows. The performance of the methods is assessed when applied to the long-range propagation of a linear one-dimensional disturbance, temporally decaying isotropic turbulence, and turbulent boundary layers. The dispersion and dissipation errors, as well as the cost of the simulations are monitored and compared to the results given by explicit Runge–Kutta schemes. The one-dimensional tests show that the original DP2 scheme leads to dispersive errors. A modified scheme, DP2M, that reduces dispersive errors is also presented. We find that both, the original and modified methods, give accurate results for the simulation of temporally decaying isotropic turbulence and turbulent boundary layers. From the isotropic turbulence problem, we observe that the dispersive behavior of the DP2 method decreases with decreasing ϵ. For the boundary layer calculation, errors are not found for $\epsilon > 10^{-5}$. This is consistent with the isotropic turbulence results. The DP2M method is more stable than the DP2. The suggested value of ϵ for DNS of compressible turbulence is 10^{-6}. We also suggest $m_{\max} = 4$ and $p_{\max} = 20$.

The stability criterion for compressible wall-bounded flows is very stringent due to the required small wall-normal grid spacing and large speed of sound. Thus, very small time steps must be used with explicit methods, resulting in very costly simulations. On the other hand, the maximum allowable time step for the implicit methods is limited by the solution accuracy. We find that this limit is much less strict than the stability limit, making the DP2M method approximately six times less costly than the RK3 method for the Mach 2.32 turbulent boundary layer test problem. In contrast, no significant cost savings are found for the incompressible boundary layer. Thus, the DP2M method is an attractive approach for the direct numerical simulation of compressible wall-bounded turbulent flows. Time-accurate results are found for compressible decaying isotropic turbulence as well, with computing saving factors of two and a half times over RK4 calculations.

The difference between the compressible and incompressible flow results is due to the extra constrain that is imposed by the variation of temperature in a compressible flow. The non-uniform temperature distribution results in sound speed and viscosity variations, which result in more stringent allowable stable time steps Eqs. (4) and (5), relative to an incompressible flow. We find that for compressible flows this limit is much less strict and the cost savings using a implicit scheme are more significant for compressible than for incompressible flows, regardless of the flow being turbulent or laminar. The method works well for the DNS of wall-bounded compressible turbulence.

Acknowledgements

This work was sponsored by the National Science Foundation under Grant number CTS-0238390 and by the Army High Performance Computing Research Center under the auspices of the Department of the Army, Army Research Laboratory cooperative agreement DAAD191-01-2-0014, the content of which does not necessarily reflect the position or policy of the government, and no official should be inferred. The computing time was provided by the CROCCO Laboratory in Princeton University.

References

[1] G.V. Candler, M.J. Wright, J.D. McDonald, AIAA Journal 32 (1994) 2380.

[2] J.F. Debiéve, H. Gouin, J. Gaviglio, Proceedings of the ICHMT/IUTAM Symposium on the Structure of Turbulence and Heat and Mass Transfer, Dubrovnik, 1981.

[3] J.F. Debiéve, Thèse Université d'Aix Marseille II, Marseille, France, 1983.

[4] M. Eléna, J.P. Lacharme, J. Gaviglio, in: A. Dybb, P.A. Pfund (Eds.), International Symposium on Laser Anemometry, ASME, 1985.

[5] M. Eléna, J.P. Lacharme, Journal Mécanique Théorique et Appliquée 7 (1988) 175.

[7] H.H. Fernholz, P.J. Finley, J.P. Dussauge, A.J. Smits, AGARDograph No. 315, 1989.

[8] P.S. Klebanoff, NACA TN-3178, 1954.

[9] S. Lee, S.K. Lele, P. Moin, Physics of Fluids 3 (1991) 657.

[10] S.K. Lele, Journal of Computational Physics 114 (1992) 265.

[11] M.P. Martín, G.V. Candler, Physics of Fluids 10 (1998) 1715.

[12] M.P. Martín, U. Piomelli, G.V. Candler, Theoretical and Computational Fluids Dynamics 13 (2000) 361–376.

[13] M.P. Martín, 34th AIAA Fluid Dynamics Conference and Exhibit, Paper No. 2004-2337, Portland, OR, 2004. Also: M.P. Martín, DNS of hypersonic turbulent boundary layers. Part I: Initialization and comparison with experiments, Journal of Fluid Mechanics, submitted for publication.

[14] M.P. Martín, E.M. Taylor, M. Wu, V.G. Weirs, Journal of Computational Physics, submitted for publication.

[15] P.R. Spalart, Journal of Fluid Mechanics 187 (1988) 61–98.

[16] T.L. Tysinger, D.A. Caughey, AIAA Paper No. 1991-0242, 1991.

[17] V.G. Weirs, G.V. Candler, AIAA Paper No. 1997-1940, 1997. Also in V.G. Weirs Ph.D. Thesis, University of Minnesota, 1998.

[18] M.J. Wright, G.V. Candler, J.D. McDonald, AIAA Paper No. 1995-1750, 1995.

[19] S. Yoon, A. Jameson, AIAA Paper No. 1987-0600, 1987.

[20] S. Xu, M.P. Martín, Physics of Fluids 16 (7) (2004). Also AIAA Paper No. 2003-3963, 2003.