INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2009; **59**:1063–1092
Published online 9 July 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1854

Study of a staggered fourth-order compact scheme for unsteady incompressible viscous flows

R. Knikker*,†

*CETHIL, CNRS, INSA-Lyon, Université Lyon1, Bât. Sadi Carnot, 9 Rue de la Physique,
F-69621 Villeurbanne, France*

SUMMARY

The objective of this paper is the development and assessment of a fourth-order compact scheme for unsteady incompressible viscous flows. A brief review of the main developments of compact and high-order schemes for incompressible flows is given. A numerical method is then presented for the simulation of unsteady incompressible flows based on fourth-order compact discretization with physical boundary conditions implemented directly into the scheme. The equations are discretized on a staggered Cartesian non-uniform grid and preserve a form of kinetic energy in the inviscid limit when a skew-symmetric form of the convective terms is used.

The accuracy and efficiency of the method are demonstrated in several inviscid and viscous flow problems. Results obtained with different combinations of second- and fourth-order spatial discretizations and together with either the skew-symmetric or divergence form of the convective term are compared. The performance of these schemes is further demonstrated by two challenging flow problems, linear instability in plane channel flow and a two-dimensional dipole–wall interaction. Results show that the compact scheme is efficient and that the divergence and skew-symmetric forms of the convective terms produce very similar results. In some but not all cases, a gain in accuracy and computational time is obtained with a high-order discretization of only the convective and diffusive terms. Finally, the benefits of compact schemes with respect to second-order schemes is discussed in the case of the fully developed turbulent channel flow. Copyright © 2008 John Wiley & Sons, Ltd.

Received 21 December 2007; Revised 25 April 2008; Accepted 29 April 2008

KEY WORDS: unsteady incompressible viscous flows; high-order compact schemes; boundary schemes; kinetic energy conservation

*Correspondence to: R. Knikker, CETHIL, CNRS, INSA-Lyon, Université Lyon1, Bât. Sadi Carnot, 9 Rue de la Physique, F-69621 Villeurbanne, France.

†E-mail: ronnie.knikker@insa-lyon.fr

Contract/grant sponsor: Institut de Développement et de Ressources en Informatique Scientifique, CNRS (IDRIS)
Contract/grant sponsor: Fédération Lyonnaise de Calcul Haute Performance (FLCHP)

1. INTRODUCTION

Simulation of turbulent or other convection-dominated unsteady incompressible flows using direct numerical simulation or large eddy simulation (LES) requires a numerical method that properly resolves all flow scales. This can be obtained by using fine grids with standard second-order methods, but the efficiency of the scheme is often compromised by large memory requirements or poor accuracy. The alternative is the use of high-order methods that allow reduction in the grid size while improving the precision of the simulation. This is already common practice in the compressible flow community where a large number of direct numerical simulations have been reported using sixth-order [1, 2] or even higher-order methods [3]. However, the incompressibility condition imposed on flows in the low Mach number limit and the resulting velocity–pressure coupling appears to be a stumbling block for the development of high-order methods. An exception to this rule are spectral methods that have proven to be very accurate and efficient for this type of problems and are therefore widely used to simulate incompressible flows [4]. Unfortunately, these methods are only efficient for constant property flows with simple geometries and boundary conditions. Another exception are methods based on a vorticity formulation of the governing equations [5, 6]. The problem of velocity–pressure coupling is circumvented, but difficulties arise in the implementation of boundary conditions and the extension to three-dimensional flow problems. In addition, these methods are still limited to constant property flows.

Accuracy and computational efficiency are among the arguments for the development of high-order schemes. Increasing the order of accuracy of the numerical method often allows the reduction of the mesh size and thereby the memory requirements of a simulation, although this mainly depends on the given flow problem. On the other hand, the computational time per grid node increases as well as the complexity of programming, debugging and the use of the software. The real benefit of high-order methods is therefore not always evident. The largest improvement can be expected when standard second-order methods are replaced by fourth-order methods, and by doing so, this paper will attempt to answer some of these questions in the context of unsteady incompressible viscous flows.

This study is also motivated by the widespread development of LES for which there has always been an interest in high-order schemes. Kravchenko and Moin [7] showed, for example, from turbulent channel flow simulations, that truncation errors from second-order discretization overwhelm the effect of the subgrid-scale model in contrast to high-order schemes. However, they also pointed out that aliasing errors may be more pronounced for high-order schemes and special care must be taken when formulating the non-linear convective terms. In addition, the interaction between subgrid-scale model and the numerical method is not always evident: models validated with second-order methods might not work correctly when combined with high-order methods due to the combined effect of numerical errors and the subgrid-scale model. Recently, Meyers *et al.* [8] discussed the issue of error and quality assessment of numerical schemes in the context of LES of homogeneous isotropic decaying turbulence.

This paper investigates the accuracy and efficiency of second-order and compact fourth-order finite-difference schemes in the framework of direct numerical simulation of viscous incompressible flows. In contrast to LES, no model is used and the difference with the exact solution is attributed only to discretization errors. Several combinations of second-order and compact discretizations of the momentum equations are tested, together with either the gradient or skew-symmetric forms of the convective terms. This paper focuses on the use of compact discretizations for incompressible flows, basing the development on recent works in this field. A numerical method is proposed

DOI: 10.1002/fld

and described in detail, including the implementation of boundary conditions and a discussion of global conservation issues. Most of the numerical tools are readily available in the literature, but, in practice, the final selection of the numerical scheme and its implementation can be a rather complex and tedious task. This work is therefore also an attempt to assemble and verify some of the established techniques.

The remainder of this paper is organized as follows. In Section 2, a brief literature review on the development of high-order schemes for incompressible flows is given, providing the theoretical basis of the method used in this paper. In Section 3, the numerical method is described in detail, including boundary implementation and global conservation issues. The performance of this method is then investigated in Section 4 from several unsteady two-dimensional and three-dimensional flow problems. Final conclusions are drawn in Section 5.

2. BACKGROUND

The objective here is to provide the reader with an overview of recent works on compact schemes for unsteady incompressible flows and to introduce the numerical method described in this paper. Compact schemes [9], sometimes called Padé schemes, are widely used for the simulation of compressible flows and have shown to be very accurate and efficient, in particular, for resolving the small scales of the flow [1]. In addition, the compact stencil allows an easy implementation of boundary conditions and therefore minimizes the use of boundary-specific schemes. This study focuses on the development of high-order compact schemes to solve the incompressible flow equations.

A staggered grid arrangement has been retained in this study in order to avoid the well-known odd–even decoupling problem on the pressure, occurring in incompressible flow solvers. In the literature, the choice of collocated grid arrangement can be found, for example, in the study of Wilson *et al.* [10, 11], who developed the fourth- and sixth-order compact schemes on a curvilinear grid. They showed that, in theory, the use of a consistent fourth-order discrete Poisson equation eliminates the odd–even decoupling problem, but the coupling between neighboring pressure nodes remains weak and the convergence rate of standard solvers is actually degraded.

The construction of high-order schemes can be based on centered or upwind stencils. Although there have been some developments on upwind-based high-order compact schemes [12, 13], this study is focused on centered schemes in order to minimize the numerical dissipation, which might degrade the quality of the solution. Central schemes have also been recommended in the framework of LES [14] because the numerical dissipation can mask the effect of the subgrid-scale model (a notable exception is implicit LES [15] where the numerical method explicitly participates in the subgrid-scale dissipation). The discrete formulation of the scheme, however, has to be carefully chosen to ensure a stable time-evolving solution. This can be obtained by the use of the so-called energy-conserving schemes that conserve globally or locally some quadratic form of any transported variable. In addition, it was shown in Kravchenko and Moin [7] that violation of energy conservation is due to aliasing errors and contaminates the solution mostly at high wavenumbers or small scales. Therefore, in the prospect of LESs, the choice of a energy-conserving scheme appears to be well justified. This property is investigated in the scheme developed here.

The concept of kinetic energy conservation has received considerable attention over the last decades, likely because it provides a natural stability criterion for incompressible flow methods; it is easily shown that the kinetic energy is a conserved quantity for inviscid flows, or that it

DOI: 10.1002/fld

decreases in time when viscous effects are present. Numerical methods that fulfill this criterion in a discrete sense are found, for example, in Morinishi *et al.* [16] and Verstappen and Veldman [17]. Most of the efforts have been focused on the non-linear convective term, which is generally the main source of aliasing errors, energy conservation violation and subsequent instability problems. To obtain specific discrete properties, the convective term is recast in either divergence, advective, skew-symmetric or rotational formulations. Although the divergence and advective expressions are the most natural forms of the convective term, it is the skew-symmetric form that is commonly recommended with respect to energy conservation. Conservation of kinetic energy for the skew-symmetric form of the convective term is easily derived by integration of the inner product of the momentum equations for u_i and the velocity components u_i over the entire flow domain. The analytical proof requires the use of *integration by parts*, which states that

$$\int_a^b u_i \frac{\partial u_i u_j}{\partial x_j} \mathrm{d}x_j = u_i u_i u_j \Big|_a^b - \int_a^b u_i u_j \frac{\partial u_i}{\partial x_j} \mathrm{d}x_j \tag{1}$$

and subsequently,

$$\int_a^b u_i \left[\frac{1}{2} \frac{\partial u_i u_j}{\partial x_j} + \frac{1}{2} u_j \frac{\partial u_i}{\partial x_j} \right] \mathrm{d}x_j = \frac{1}{2} u_i u_i u_j \Big|_a^b \tag{2}$$

This relation shows that the convective term written in skew-symmetric form (term within brackets) contributes to the kinetic energy only through the boundaries and is therefore conserved within the entire flow domain. To show conservation for the divergence form, one needs in addition to invoke the *product rule*, which is a *local* property stating that

$$u_j \frac{\partial u_i}{\partial x_j} = \frac{\partial u_i u_j}{\partial x_j} - u_i \frac{\partial u_j}{\partial x_j} \tag{3}$$

Substituting in Equation (1) and dividing by two result in

$$\int_a^b u_i \frac{\partial u_i u_j}{\partial x_j} \mathrm{d}x_j = \frac{1}{2} u_i u_i u_j \Big|_a^b + \int_a^b \frac{1}{2} u_i u_i \frac{\partial u_j}{\partial x_j} \mathrm{d}x_j \tag{4}$$

where the continuity equation makes the last term identically zero. The proof for the advective form is very similar and also requires the use of integration by parts, the product rule and the continuity equation. Since a discrete equivalent of integration by parts exists for both spectral and finite-difference methods on periodic grids, even in the presence of aliasing errors, the skew-symmetric formulation is conservative in a discrete sense [7, 18]. The *product rule* is, on the contrary, only verified by specifically designed schemes. A second-order kinetic energy-conserving scheme was developed by Harlow and Welch [19] and an extension to a fourth-order scheme can be found in Morinishi *et al.* [16]. However, in both studies, energy conservation was only provided on uniform grids. Ham *et al.* [20] developed a second-order conservative scheme that preserves its conservation properties and second-order accuracy on non-uniform rectangular grids, provided grid stretching is a smooth function of space. This was obtained by using proper interpolation operators for the advection velocity and the transported scalar in the convective term. Fukagata and Kasagi [21] noted that the same scheme can be obtained in a more elegant way using a coordinate transformation, $x_i = x_i(\xi^i)$, and by averaging the covariant velocity components. The use of coordinate transformation and discretization of the equations in computational space

DOI: 10.1002/fld

rather than physical space have recently allowed the development of energy-conserving high-order explicit schemes. Morinishi *et al.* [22], for example, proposed a conservative scheme in cylindrical coordinates. Coordinate transformation can be applied to the divergence and skew-symmetric form of the flow equations and will be retained in this study for the development of high-order compact schemes with good conservation properties. Note that a different approach to obtain high-order accuracy on non-uniform grids consists of including the metrics in the coefficients of the spatial discretization scheme. This can be found in the context of compressible flows in [23] and incompressible flows in the vorticity-streamfunction formulation in [6], although the energy conservation issue has not been addressed.

For completeness, it should be noted that to obtain energy conservation in time as well as in space, an implicit mid-point rule must be used for the discretization of all convected quantities. The time level of the advection velocity does not affect the conservation properties of the scheme and can therefore be set to the previous time level. A linear implicit system is then obtained that is solved with less effort than a fully implicit scheme [17, 24]. In this study, as well as many other applications, a semi-explicit time integration is used introducing numerical dissipation of the kinetic energy. The numerical error can be controlled by either adapting the time step or the order of accuracy of the time scheme.

Most of the previously cited works are based on second- or high-order explicit schemes to discretize the incompressible flow equations. Several recent papers can be found that developed and analyzed compact schemes for incompressible flows, based on staggered grid arrangements. In fact, most of the basic ideas retained in more recent papers can already be found in Schiestel and Viazzo [25]. In their work, the convective term is cast in a skew-symmetric form and conservation of momentum and kinetic energy in the inviscid limit was demonstrated using integration by parts. A consistent fourth-order discretization of the pressure equation was used and solved iteratively with preconditioning based on a second-order discrete Poisson solver. However, the paper did not provide much information on the implementation of the method. This gap was filled by Vedy *et al.* [26] who extended the method to smoothly non-uniform Cartesian grids. Brüger *et al.* [27, 28] discretized the advective form of flow equations using collocated and staggered compact derivatives. A staggered curvilinear grid with orthogonal transformation and local velocity components was used, but the question of energy conservation has not been treated. The pressure was solved using a preconditioning technique based on second-order discretizations. Abide and Viazzo [29] combined a fourth-order compact discretization with a two-dimensional projection decomposition method. In the work of Kampanis and Ekaterinaris [30], several explicit and implicit second- and fourth-order schemes are compared on regular Cartesian and cylindrical grids. They also compared different solvers for the pressure equation: a local pressure correction, a global pressure correction and the GMRES method. A global method similar to the one used here appeared to be the best choice for high Reynolds number flows. For completeness, we mention the works of Pereira *et al.* [31] and Piller and Stalio [32] who developed compact finite-volume schemes for incompressible flows on curvilinear grids.

The numerical method proposed in this study combines several of the established techniques presented above. For example, coordinate transformation is used to preserve kinetic energy on non-uniform rectangular grids in combination with the skew-symmetric form of the convective term. The pressure equation is solved with an iterative pressure correction method, based on a second-order accurate multigrid solver. Special attention has been paid in this study to the integration of physical boundary conditions in the numerical schemes. A detailed description of the method is given in the following section.

3. NUMERICAL METHOD

3.1. System of equations

Starting point are the governing equations for incompressible flows, which are solved on a Cartesian non-uniform grid. A curvilinear transformation is initially used to project the physical coordinates x_i on a regular computational domain with coordinates ξ^i. This transformation results in the following set of equations:

$$\begin{aligned} J\frac{\partial U^i}{\partial \xi^i} &= 0 \\ \frac{\partial u_i}{\partial t} + J\frac{\partial}{\partial \xi^j}(U^j u_i + b_i^j p - \widehat{\tau}_i^j) &= 0 \end{aligned} \tag{5}$$

where u_i are the velocity components in the ith direction, p is the pressure divided by the reference density, $J=\det(\partial\boldsymbol{\xi}/\partial\mathbf{x})$ is the Jacobian of the coordinate transformation and $b_i^j = J^{-1}\,\partial\xi^j/\partial x_i$ are the contravariant transformation vectors. These are used to define the covariant velocity and viscous stress tensor components:

$$U^j = b_k^j u_k \tag{6}$$

$$\widehat{\tau}_i^j = b_k^j \tau_{ik} \tag{7}$$

where $\tau_{ik} = \nu(\partial u_i/\partial x_k + \partial u_k/\partial x_i)$ is the viscous stress tensor in Cartesian coordinates. The velocity gradients in Cartesian coordinates are calculated from space derivatives in computational space using the transformation

$$\frac{\partial u_i}{\partial x_k} = J b_k^j \frac{\partial u_i}{\partial \xi^j} \tag{8}$$

Note that Equations (5) are expressed in a conservative variable-viscosity form, although all flow properties are kept constant throughout the paper. It is shown in Section 4 that this formulation does not conserve kinetic energy when combined with compact finite-difference schemes. For this reason, a fully conservative skew-symmetric form of the convective term is also explored in this study.

The above equations are applicable to general curvilinear transformations, but we restrict ourselves to an orthogonal transformation of the form $x_i = x_i(\xi^i)$. This implies that $b_i^j = 0$ for $i \neq j$ allowing several simplifications of the governing equations. Defining the grid scaling coefficient $\xi_x^i \equiv (\partial x_i/\partial \xi^i)^{-1}$, we can write

$$J b_i^j = \xi_x^i \delta_{ij} \tag{9}$$

The coefficients ξ_x^i are calculated from the derivatives of x_i in the computational domain. The Jacobian of the transformation is then simply $J = \xi_x^1 \xi_x^2 \xi_x^3$ and the continuity equation reduces to

$$J\frac{\partial U^j}{\partial \xi^j} = J\frac{\partial}{\partial \xi^j}\left(\frac{\xi_x^j u_j}{J}\right) = \frac{\partial u_j}{\partial \xi^j} = 0 \tag{10}$$

since $\partial\xi^i_x/\partial\xi_j = 0$ for $i \neq j$. In the same manner, the pressure and viscous terms in the momentum equation are simplified, resulting in the equation

$$\frac{\partial u_i}{\partial t} + J\frac{\partial}{\partial \xi^j}(U^j u_i) + \xi^i_x \frac{\partial p}{\partial \xi^i} = \xi^j_x \frac{\partial \tau^j_i}{\partial \xi^j} \tag{11}$$

It is important to maintain the non-linear convective term in the general curvilinear form to ensure global conservation of kinetic energy. It is also for this reason that the general curvilinear coordinates are used as a starting point. Equation (11) is strictly equivalent to the momentum equation (5) also on the discrete level. Mass and momentum are therefore conserved if the discrete difference operator is globally conservative, which is the case for many finite-difference schemes including the staggered fourth-order compact schemes used in this paper.

3.2. Discrete formulation of the governing equations

The discretization of the non-linear terms in the momentum equations is inspired from the so-called 'fully conservative' schemes designed by Morinishi *et al.* [16]. They developed explicit second- and fourth-order centered schemes, based on combinations of discrete interpolation and differentiation on a staggered grid. These schemes are fully conservative in the sense that mass, momentum and kinetic energy are conserved. It is shown here that this property does not hold for all finite-difference schemes, but their procedure leads to an elegant formulation for the development of higher-order schemes with good stability properties.

Following the notation of Morinishi *et al.* [16], the discretization of the convective terms in conservative or divergence form may be written as

$$J\frac{\partial}{\partial \xi^j}(U^j u_i) \simeq J\frac{\delta}{\delta \xi^j}\left(\overline{U^j}^i\, \overline{u_i}^j\right) = \xi^j_x \frac{\delta}{\delta \xi^j}\left(\widetilde{u_j}^i\, \overline{u_i}^j\right) \tag{12}$$

where $\delta/\delta\xi^j$ is the fourth-order compact finite-difference operator and $\overline{u_i}^j$ is the fourth-order compact interpolation operator of u_i in the direction j, which are defined in Section 3.3. The discrete convective terms can be rewritten using a weighted interpolation, defined by $\widetilde{u_j}^i \equiv J\overline{U^j}^i/\xi^j_x$, as in the last term of Equation (12).

The same notation can be used to formulate a fourth-order compact discretization of the skew-symmetric form of the convective terms,

$$\begin{aligned} &J\left[\frac{1}{2}\frac{\partial}{\partial \xi^j}(U^j u_i) + \frac{1}{2}U^j \frac{\partial u_i}{\partial \xi^j}\right] \\ &\quad \simeq J\left[\frac{1}{2}\frac{\delta}{\delta \xi^j}\left(\overline{U^j}^i\, \overline{u_i}^j\right) + \frac{1}{2}\overline{\overline{U^j}^i \frac{\delta u_i}{\delta \xi^j}}^j\right] = \xi^j_x\left[\frac{1}{2}\frac{\delta}{\delta \xi^j}\left(\widetilde{u_j}^i\, \overline{u_i}^j\right) + \frac{1}{2}\overline{\widetilde{u_j}^i \frac{\delta u_i}{\delta \xi^j}}^j\right] \end{aligned} \tag{13}$$

The remaining terms of the governing equations are discretized in a similar manner using the same fourth-order compact finite-difference operators. The semi-discretized equations with the

DOI: 10.1002/fld

convective term in divergence form are then given by

$$
\begin{aligned}
\frac{\delta u_j}{\delta \xi^j} &= 0\\
\frac{\partial u_i}{\partial t} + \xi^j_x \frac{\delta}{\delta \xi^j}\left[\widetilde{u}^i_j\, \overline{u_i}^j + p\delta_{ij} - \nu\left(\frac{\delta u_i}{\delta x_j} + \frac{\delta u_j}{\delta x_i}\right)\right] &= 0
\end{aligned}
\tag{14}
$$

In flow problems with variable viscosity, ν is required at the center and vertices of the computational cells. This can be obtained by successive application of the fourth-order compact interpolation in the directions i and j.

3.3. *Compact finite-differencing and interpolation*

The system of equations is discretized on a staggered grid, which is divided into cells of size $h \times h$ in the computational space ($h = 1$), as depicted in Figure 1. Scalars such as pressure and temperature are located at the cell center (symbol $\circ$), whereas the components of the velocity vector are defined at the mid-points of the cell faces (symbols $\triangleright$ and $\triangle$). For convenience, only the two-dimensional case is considered here and the coordinates and velocity components are noted as ξ^1, ξ^2, u and v. The equations are easily extended to three-dimensional flow problems.

3.3.1. Differentiation operations for the flow variables. All space derivatives are discretized using the staggered centered fourth-order compact scheme [9]. For example, the derivative $u' \equiv \delta u/\delta \xi^1$ in the mass conservation equation is defined by

$$
u'_{i-1} + 22u'_i + u'_{i+1} = \frac{24}{h}(u_{i+1/2} - u_{i-1/2}) + \mathcal{O}(h^4) \quad \text{for } i \in \{2, \ldots, N-1\} \tag{15}
$$

where N denotes the number of grid cells. This implicit scheme leads to a tridiagonal system of equations which is efficiently solved using the Thomas algorithm. Equation (15) is used for each first derivative defined on the interior grid points, but boundary schemes are needed to get a closed system of equations. Differences arise in the discretization of each flow variable (u, v, p) due to the staggered locations and specific boundary conditions.

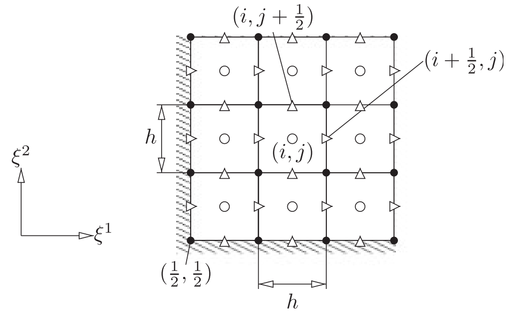

Figure 1. Staggered grid arrangement in a two-dimensional computational space.

The value of u' on the boundary cells is defined using the following third-order one-sided compact scheme:

$$u'_1 - u'_2 = \frac{1}{h}(-u_{1/2} + 2u_{3/2} - u_{5/2}) + \mathcal{O}(h^3) \tag{16}$$

Wall and inlet boundary conditions are implemented by setting the normal velocity $u_{1/2} = u^{\text{bnd}}_{1/2}$ prior to evaluation of Equation (16).

The derivative $v' \equiv \delta v/\delta\xi^1$ is evaluated at the vertices of the computational cell (symbol • in Figure 1), which are also located at the boundaries of the computational domain. In cases of no-slip wall or inlet boundary conditions, the shear velocity, $v^{\text{bnd}}_{1/2}$, that does not have a grid point at the boundary is explicitly included in the scheme. This is necessary to properly account for shear stress at the boundary. We use the third-order compact scheme defined by

$$v'_{1/2} + 3v'_{3/2} = \frac{8}{3h}(v_2 - v^{\text{bnd}}_{1/2}) + \mathcal{O}(h^3) \tag{17}$$

When the shear velocity is not explicitly imposed (slip wall or outlet), a general third-order scheme is used,

$$v'_{1/2} + 23v'_{3/2} = \frac{1}{h}(-25\,v_1 + 26v_2 - v_3) + \mathcal{O}(h^3) \tag{18}$$

The pressure gradient $p' = \delta p/\delta\xi^1$ is calculated at the cell faces using the shifted version of the compact scheme (15). In case of wall boundary, the wall-normal pressure gradient, $p'^{\text{bnd}}_{1/2}$, is calculated directly from the discrete momentum equation for u. Therefore, no special boundary treatment is necessary and the scheme remains fourth-order accurate on the entire domain. A constant pressure outlet boundary may be implemented using the third-order one-sided compact scheme,

$$p'_{1/2} + 3p'_{3/2} = \frac{8}{3h}(p_2 - p^{\text{bnd}}_{1/2}) + \mathcal{O}(h^3) \tag{19}$$

where $p^{\text{bnd}}_{1/2}$ is the outlet boundary pressure. For other boundary conditions, Equation (19) can also be used to determine the pressure at the boundary.

3.3.2. Staggered interpolation of the flow variables. The staggered interpolation appearing in the convective terms are defined by fourth-order compact schemes. For example, the velocity u shifted in the direction ξ^1, noted here as $\overline{u}$, is a face-to-cell-center interpolation obtained by solving the equation

$$\overline{u}_{i-1} + 6\overline{u}_i + \overline{u}_{i+1} = 4(u_{i-1/2} + u_{i+1/2}) + \mathcal{O}(h^4) \quad \text{for } i \in \{2, \ldots, N-1\} \tag{20}$$

At the boundary, the fourth-order scheme

$$\overline{u}_1 + \overline{u}_2 = \tfrac{1}{4}u^{\text{bnd}}_{1/2} + \tfrac{3}{2}u_{3/2} + \tfrac{1}{4}u_{5/2} + \mathcal{O}(h^4) \tag{21}$$

is used, where $u^{\text{bnd}}_{1/2}$ is the normal velocity at the boundary. The face-to-vertex interpolation of the velocity v in the direction ξ^1, noted here as $\overline{v}$, is calculated on interior points using a shifted version of Equation (20). If the shear velocity on the boundary is known, then we simply set $\overline{v}_{1/2} = v^{\text{bnd}}_{1/2}$

DOI: 10.1002/fld

and no boundary scheme is needed. In other cases, the value at the boundary is calculated with the third-order scheme

$$\overline{v}_{1/2}+5\overline{v}_{3/2}=\tfrac{15}{4}v_1+\tfrac{5}{2}v_2-\tfrac{1}{4}v_3+\mathcal{O}(h^3) \tag{22}$$

The weighted interpolation of the advection velocity, $\widetilde{u_j}^i$, defined in Equation (12) requires special attention. If the direction of interpolation and velocity component coincide ($i=j$), then $\widetilde{u_i}^j$ is strictly equivalent to the interpolation described previously. However, for $i\neq j$ the velocity is a weighted interpolation defined by

$$\widetilde{u_i}^j=\xi^i_x\overline{(\xi^i_x)^{-1}u_j}^i$$

The velocity is first multiplied by the inverse of the scale factors ξ^i_x evaluated at the mid-points of the cell face, then interpolated using the fourth-order compact scheme defined by Equation (20) and the result is multiplied with the coefficient ξ^i_x evaluated at the cell vertices. Boundary schemes are equivalent to those described previously: the shear velocity on the boundary is included in the scheme whenever possible, thereby avoiding the use of a boundary scheme, otherwise, the third-order boundary scheme of Equation (22) is used.

3.3.3. Differentiation operators for the fluxes. Differentiation of the momentum fluxes in Equation (14) is also obtained with fourth-order compact schemes. When the fluxes are located at the cell vertices and their derivatives are evaluated at cell faces (for example, for the term $\delta(\overline{u}^2\overline{v}^1)/\delta\xi^1$ in the equation for v), Equation (15) for mass conservation is again used. With $f\equiv\overline{u}^2\overline{v}^1$, this scheme is given by

$$f'_{i-1}+22f'_i+f'_{i+1}=\frac{24}{h}(f_{i+1/2}-f_{i-1/2})\quad\text{for } i\in\{2,\dots,N-1\} \tag{23}$$

The system of equations is closed using the third-order boundary scheme

$$f'_1-f'_2=\frac{1}{h}(-f_{1/2}+2f_{3/2}-f_{5/2})+\mathcal{O}(h^3) \tag{24}$$

The use of these schemes leads to a correct conservation of total mass and momentum in the computational domain. This conservation property is expressed as a weighted sum of the flux derivatives in each direction

$$\sum_{i=1}^{N}c_i f'_i h=f^{\text{bnd}}_{N+1/2}-f^{\text{bnd}}_{1/2} \tag{25}$$

The coefficients are obtained using linear algebra and defined as $c_1=c_N=\frac{26}{24}$, $c_2=c_{N-1}=\frac{21}{24}$, $c_3=c_{N-2}=\frac{25}{24}$ and $c_i=1$ for the remaining coefficients (requiring at least seven grid cells in each direction).

For the remaining fluxes, located at the cell center (for example, for the term $\delta(\overline{u}^1\overline{u}^1)/\delta\xi^1$ in the equation for u), the shifted version of Equation (15) is used. With $f\equiv\overline{u}^1\overline{u}^1$, this scheme is given by

$$f'_{i-1/2}+22f'_{i+1/2}+f'_{i+3/2}=\frac{24}{h}(f_{i+1}-f_i)\quad\text{for } i\in\{1,\dots,N-1\} \tag{26}$$

which is, of course, also the scheme used to calculate the pressure gradient. The system of equations can be closed with the third-order boundary scheme:

$$f'_{1/2}+3f'_{3/2}=\frac{8}{3h}(f_2-f^{\text{bnd}}_{1/2})+\mathcal{O}(h^3) \tag{27}$$

The use of these schemes leads to a correct conservation of the total momentum expressed by the weighted sum

$$\sum_{i=0}^{N} w_{i+1/2}\, f'_{i+1/2}\, h = f^{\text{bnd}}_{N+1/2}-f^{\text{bnd}}_{1/2} \tag{28}$$

The coefficients are given by $w_{1/2}=w_{N+1/2}=\frac{3}{8}$, $w_{3/2}=w_{N-1/2}=\frac{7}{6}$, $w_{5/2}=w_{N-3/2}=\frac{23}{24}$ and $w_{i+1/2}=1$ for the remaining coefficients. The momentum flux in Equation (14) is composed of three terms: the convective flux, pressure and viscous stress. However, the values of the last two terms are generally not known at the boundary, but only on the interior grid points. No-slip wall and inlet boundary conditions are therefore implemented using the following procedure:

- The convective flux at the boundary is calculated from the velocity at the boundary. The flux derivatives are calculated with Equations (26) and (27).
- For the viscous fluxes, the boundary scheme defined by Equation (18) is used instead of Equation (27). Derivatives at the interior points are again calculated with Equations (26).
- The normal pressure gradient ${p'}^{\text{bnd}}_{1/2}$ is calculated directly from the discrete momentum equation by requiring that the wall normal velocity does not evolve in time (i.e. $\partial u/\partial t=0$ at the boundary). A boundary scheme is therefore not needed.

Note that Equation (27) can still be used to calculate the normal stress at the boundary, in a manner that is consistent with the conservation property expressed by Equation (28). For slip-wall conditions, symmetry of the wall-normal flux is assumed and all flux derivatives are set to zero at the wall.

3.3.4. Interpolation of the convective fluxes. The skew-symmetric form of the convective term requires an additional interpolation in the last part of Equation (13). For the cell-face to cell-center interpolation, Equation (20) is used and closed using the fourth-order boundary scheme given by Equation (21). For cell-center to cell-face interpolation, values at the boundary of the computational domain are identically zero and no boundary schemes are required.

3.3.5. Evaluation of metric coefficients. In order to conserve fourth-order accuracy on smoothly varying non-uniform grids, the grid scaling coefficients $\xi^k_x\equiv(\partial x_k/\partial\xi^k)^{-1}$ require evaluation with at least the same order of accuracy. Fourth-order compact schemes are applied to estimate the first derivative of the grid coordinates x_k with respect to the computational grid coordinate ξ^k. First, the numerical grid is defined in the physical domain by defining the coordinates of the vertices of the grid cells, for example, $x_{i+1/2}$, where i denotes the ith cell as in Figure 1. The metric coefficients located at the cell centers, $(\xi_x)_i$, are then calculated from the coordinates $x_{i+1/2}$ using the compact scheme defined by Equations (23) and (24). Next, the coordinates of the cell centers, x_i, are obtained in each direction by interpolation of the coordinates $x_{i+1/2}$ using the fourth-order compact scheme defined by Equations (20) and (21). The metric coefficients located at the cell faces, $(\xi_x)_{i+1/2}$, are determined from the cell coordinates x_i and the boundary coordinates $x_{1/2}$

DOI: 10.1002/fld

and $x_{N+1/2}$ using the compact scheme defined by Equations (26) and (27). It is shown in Section 4 that fourth-order accuracy is obtained on non-uniform grids.

It should be noted that an easier approach to implement boundary conditions consists of using one-sided schemes for all variables at the boundary and enforcing physical boundary conditions separately at each time step. The approach retained in this study is to include the physical boundary conditions in the boundary scheme wherever possible. This leads to a different scheme for each flow variable and spatial direction, but in our experience gives a more stable method and global conservation properties are more easily enforced. Each discrete derivative leads to a similar tridiagonal system with slightly different coefficients. The initialization of these coefficients might be cumbersome to implement, but the tridiagonal system can be solved by the same subroutine which greatly reduces software optimization efforts.

Finally, corner points (or grid points located on edges in three-dimensional domains) do not require special treatment. They are automatically taken into account by the application of the spatial discretizations and the corresponding boundary schemes in each direction. However, it may be necessary in case of slip walls to explicitly impose zero shear stress at the wall.

3.4. *Time integration*

In order to integrate the system of equations in time, a semi-implicit Runge–Kutta/Crank–Nicolson scheme [33] is used. This scheme is briefly described in this section.

Let U and P denote the velocity and pressure solution vectors at the time t, then the semi-discrete formulation of the governing equations may be written as

$$\begin{aligned} DU &= 0 \\ \frac{\mathrm{d}U}{\mathrm{d}t} &= \mathscr{L}(U)+\mathscr{N}(U)-GP+f \end{aligned} \tag{29}$$

In this scheme $\mathscr{L}(U)$ denotes the linear diffusion terms that will be treated implicitly and $\mathscr{N}(U)$ refers to remaining terms treated explicitly in time. The term f is the discrete implementation of boundary conditions and source terms, and G and D are the discrete gradient and divergence (linear) operators. Linear terms are treated implicitly using a Crank–Nicolson scheme, whereas the explicit terms are advanced using a third-order Runge–Kutta scheme. The combined scheme proceeds in three steps. Let U^n and P^n denote the velocity and pressure solution vectors at the time t^n, then time stepping is given by

$$U' = U^n + \Delta t[\beta_1 \mathscr{L}(U'+U^n) + \sigma_1 \mathscr{N}(U^n) - 2\beta_1 GP' + \sigma_1 f]$$

$$U'' = U' + \Delta t[\beta_2 \mathscr{L}(U''+U') + \sigma_2 \mathscr{N}(U') + \gamma_2 \mathscr{N}(U^n) - 2\beta_2 GP'' + \sigma_2 f]$$

$$U^{n+1} = U'' + \Delta t[\beta_3 \mathscr{L}(U^{n+1}+U'') + \sigma_3 \mathscr{N}(U'') + \gamma_3 \mathscr{N}(U') - 2\beta_3 GP^{n+1} + \sigma_3 f]$$

where U', U'', P' and P'' are the velocity and pressure estimates obtained after the first and second Runge–Kutta steps. The velocity fields are subjected to the continuity constraint:

$$DU' = DU'' = DU^{n+1} = 0$$

The coefficients β_k, γ_k and σ_k are calculated to give third-order accuracy for the convective term and second-order accuracy for the viscous term [34], see Table I.

DOI: 10.1002/fld

Table I. Coefficients for the time integration scheme.

Sub-step	β_k	γ_k	σ_k
$k=1$	$\frac{4}{15}$	0	$\frac{8}{15}$
$k=2$	$\frac{1}{15}$	$-\frac{17}{60}$	$\frac{5}{12}$
$k=3$	$\frac{1}{6}$	$-\frac{5}{12}$	$\frac{3}{4}$

3.5. Solution method using pressure correction

Each step of the Runge–Kutta scheme is solved using a fractional step or pressure correction method. This correction scheme proceeds in two steps, which will be demonstrated for the first Runge–Kutta step, $k=1$, only. First, an intermediate velocity is obtained by advancing the momentum equation using the pressure field from the previous time step:

$$U^*=U^n+\Delta t[\beta_1\mathscr{L}(U^*+U^n)+\sigma_1\mathscr{N}(U^n)-2\beta_1 GP^n+\sigma_1 f] \tag{30}$$

The pressure is now treated explicitly, but the diffusion terms in $\mathscr{L}(U^*)$ are still treated implicitly. An iterative alternating direction implicit (ADI) method is used to solve this linear system. Details can be found in Appendix A.

In the correction step, the intermediate velocity is corrected to obtain a divergence-free velocity field by solving the system

$$U'=U^*-2\beta_1\Delta t G\phi \tag{31}$$

$$DU'=0 \tag{32}$$

where $\phi=p^{n+1}-p^n$ is the pressure correction. A discrete Poisson equation for ϕ is obtained by application of the divergence operator on Equation (31) and making use of Equation (32), leading to

$$DG\phi=\frac{1}{2\beta_1\Delta t}DU^* \tag{33}$$

The boundary conditions in Equation (33) are homogeneous Neumann conditions: normal derivatives are equal to zero at the wall. Physical boundary conditions have already been taken into account by means of the source term f. A consistent fourth-order solution of ϕ is obtained only when the divergence and gradient operators are defined by the fourth-order compact schemes, in which case DG is a full singular matrix. The discrete Poisson equation is solved iteratively. Let $\phi^{(m)}$ denote the mth estimate of the pressure correction, and let

$$U^{(m)}=U^*-2\beta_1\Delta t G\phi^{(m)} \tag{34}$$

denote the mth estimate of the velocity vector U'. The iterative method consists of solving at each iteration the following simplified discrete equation:

$$L\varepsilon=\frac{1}{2\beta_1\Delta t}DU^{(m)} \tag{35}$$

DOI: 10.1002/fld

where $\varepsilon = \phi^{(m+1)} - \phi^{(m)}$ is the update for the pressure correction and L is the usual second-order central difference approximation of the Laplace operator. The fourth-order discrete pressure equation (33) is designed in such a way that the compatibility condition for the Poisson–Neumann problem is automatically satisfied, provided that the volume integral is defined by the weighted sum of Equation (25). However, the compatibility condition is violated in Equation (35) where the discrete Laplacian is replaced by a second-order approximation. Some solvers may not converge due to this incompatibility. The resolvability of the discrete Poisson equation (35) is therefore enforced by adding a small constant to the right-hand side term [35]. The computed solution is a least-square approximation of the original equation, but the final result after several iterations is, of course, fully consistent with the fourth-order discretization.

Equation (35) is the standard 5- or 7-point stencil discretization of the elliptic Poisson equation and several ready-to-use software packages are available to solve this problem. In this study, the linear system is solved using generalized cyclic reduction [36] or the multigrid method [37], depending on the number of grid points used. The multigrid solver is based on a finite-volume formulation, which provides a more suitable discretization of the Poisson equation near the boundary on a staggered grid arrangement. This is found necessary to ensure convergence of the pressure equation on highly irregular grids. In addition, the use of a conservative finite-volume formulation allows the compatibility condition to be more easily enforced on successive coarse grids.

The convergence of the iterative method is controlled by monitoring the evolution of the mass deficit, $DU^{(m)}$. An underrelaxation of the pressure update is found to be necessary to drive the mass deficit to zero within machine precision; therefore,

$$\phi^{(m+1)} = \phi^{(m)} + \alpha\varepsilon$$

where $0 \leqslant \alpha \leqslant 1$ is the relaxation factor. From numerical experiments and linear algebra calculations on regular grids, an optimum relaxation factor is found to be $\alpha = 0.83$. This value has also shown to work very well on irregular grids. In all of the presented test cases, the maximum convergence ratio is close to the theoretical value of about 0.183, but of course convergence is much faster for the earlier iterations. To speed up the overall convergence, the relaxation factor is initially set to $\alpha = 1$ and reduced to the optimum value as soon as the convergence ratio exceeds the minimum value of 0.183. Iterations are interrupted when the mass deficit drops below a threshold, which is set to 10^{-10} for the cases reported in this study.

When the multigrid solver or any other iterative second-order Poisson solver is used, one is tempted to interrupt the inner iterations before reaching full convergence since the mass deficit can be brought down to machine zero by means of the outer iterations. In this study, however, it has been found to impede overall convergence. For this reason, the maximum tolerance for the solution of Equation (35) is set to 10^{-12}.

4. NUMERICAL TESTING

The proposed numerical method has been verified in a large number of viscous and inviscid flow problems. Results of the compact schemes are compared with those obtained with other existing schemes. Table II sums up the abbreviations used in this section. In some cases, a combined second- and fourth-order scheme is used (denoted by PADE/CDS) in which the continuity equation and pressure gradients are calculated using second-order central difference schemes, whereas the

DOI: 10.1002/fld

Table II. Overview of the numerical schemes.

ID	Description
CDS	Second-order central difference scheme
CDS4	Explicit fourth-order central difference scheme of Morinishi *et al.* [16]
PADE	Fourth-order compact scheme
•/SKEW	Convective term in skew-symmetric form, Equation (13), instead of divergence form, Equation (12)
•/CDS	Second-order scheme for the pressure gradient and the continuity equation

remaining terms (convective and viscous fluxes) are discretized using one of the fourth-order compact schemes. This combination is sometimes used to improve the accuracy of second-order methods and is much cheaper than the full fourth-order method since the discrete elliptic Poisson equation (35) is solved only once every Runge–Kutta step. All schemes are implemented in Fortran 90 and use the same elliptic solver. In the following, the Courant–Friedrichs–Lewy (CFL) number is set to 0.7 unless stated otherwise.

4.1. Two-dimensional decaying vortex

In this problem, the non-dimensional Navier–Stokes equations are solved on a two-dimensional square domain. The exact solution of the periodic vortex flow is given by

$$u(x,y,t) = -A(t)\sin(\pi x)\cos(\pi y)$$

$$v(x,y,t) = A(t)\cos(\pi x)\sin(\pi y)$$

$$p(x,y,t) = \tfrac{1}{4}A(t)^2[\cos(2\pi x)+\cos(2\pi y)]$$

where $A(t)=\exp(-2\pi^2 t/Re)$ denotes the strength of the vortex, decreasing in time due to viscous dissipation. The flow variables u and v are initialized using the exact solution at $t=0$. This problem can be solved using periodic boundary conditions with $(x,y)\in[-1,1]\times[-1,1]$, or using no-slip boundary conditions on the domain $[0,1]\times[0,1]$. Figure 2 shows the L_2 norm of the numerical error of the variables u and p plotted against the cell size obtained on a periodic grid. The Reynolds number is set to $Re=1000$ and the final non-dimensional time is $t=10$. The results are given for a non-uniform grid, but the ratio of the smallest to largest cell sizes is kept constant ($\sim\frac{1}{3}$) between different grids. Results obtained with a uniform grid are very similar.

For all schemes, the order of accuracy is correctly retrieved on the non-uniform grid for all flow variables. The fourth-order compact schemes (PADE, PADE/SKEW) give very similar results for both velocity and pressure. They also perform better in this case than the explicit fourth-order scheme (CDS4). The combined scheme PADE/CDS shows a second-order convergence on all flow variables, but the velocity field is more accurately calculated in comparison with the fully second-order scheme (CDS).

Verification of the accuracy of the time integration is often difficult since the errors resulting from the spatial scheme overwhelm the error generated by time stepping. The simulation is repeated on a 64-uniform grid, but with a Reynolds number of $Re=10$ and varying CFL number. The numerical error at $t=0.3$ is plotted in Figure 3 as a function of the time step Δt, which is kept constant during each simulation. It is observed that the time scheme is only second-order accurate, in agreement with [33]. This is due to the low Reynolds number for which viscous stresses

DOI: 10.1002/fld

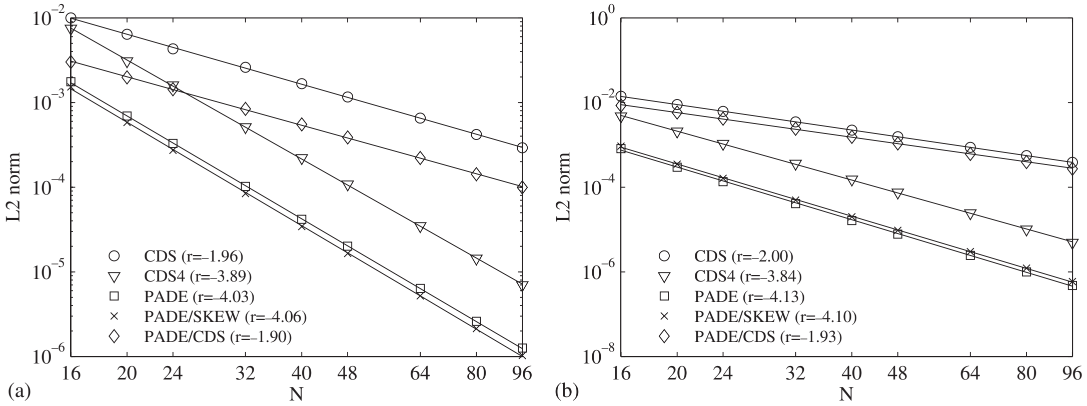

Figure 2. Vortex dissipation: numerical error for the u-velocity (a) and the pressure field (b) obtained with different numerical schemes and plotted as a function of the number of grid points in one direction. Lines indicate power law curve fittings with exponents r indicated in the legend of the figure.

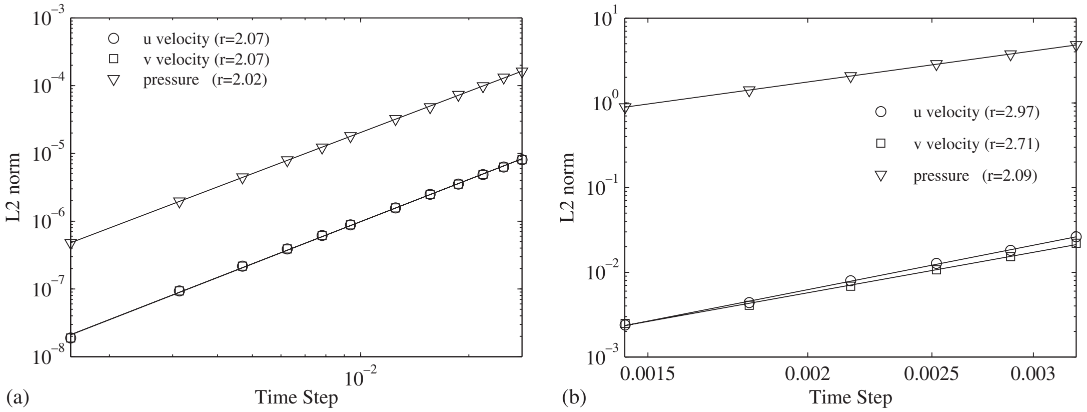

Figure 3. Time step convergence test for the L_2 norm obtained with the compact scheme and the semi-implicit time integration: (a) vortex dissipation with $Re=10$ on a 64×64 grid and (b) inviscid vortex advection on a 128×128 grid.

become the dominant source of error. Pure convective problems should retrieve the third-order accuracy.

4.2. *Vortex advection in an inviscid flow*

This benchmark is found for example in Reference [3] and describes a two-dimensional vortex convected in a uniform inviscid flow. It is an interesting problem for testing numerical methods that will be used for simulation of high Reynolds number flows or LESs, in which large-scale structures may travel through part of the computational domain while conserving their initial shape. High-order methods generally perform much better in this type of flow problems [3].

DOI: 10.1002/fld

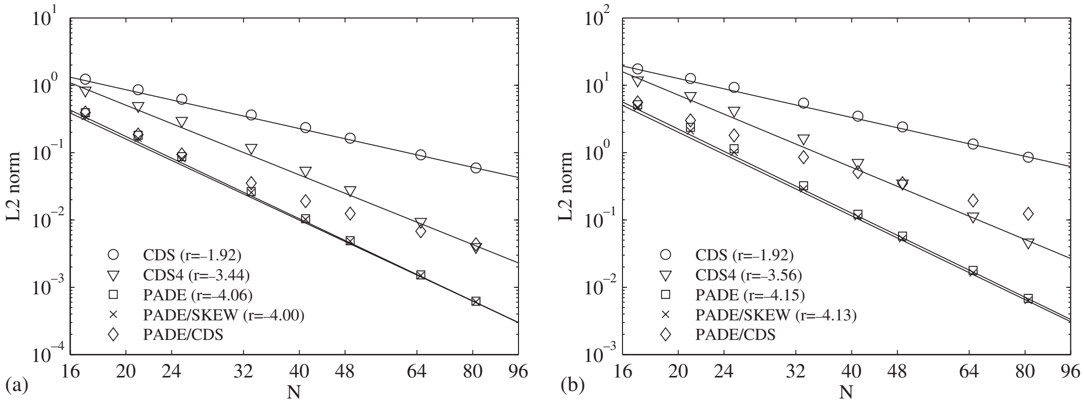

Figure 4. Inviscid vortex advection: convergence of the numerical solution to the analytical solution of the u-velocity (a) and the pressure field (b) for different numerical schemes. Lines indicate power law curves with slopes r indicated in the legend of the figure.

The flow field is initialized with a vortex centered on the origin. Velocity and pressure are defined by

$$u(x,y,0)=1-y\,C\exp(-\tfrac{1}{2}r^2)$$

$$v(x,y,0)=1+x\,C\exp(-\tfrac{1}{2}r^2)$$

$$p(x,y,0)=C^2\exp(-r^2)$$

where $r=\sqrt{x^2+y^2}$ and $C=0.6944$. The flow is solved on a periodic domain spanning the region $[-8,8]\times[-8,8]$ and the simulations are carried out until $t=2$ at which time the center of the vortex, which has been advected without decay or deformation, is located at $x=2$ and $y=2$.

Figure 4 shows the results of the convergence tests. Again the order of accuracy for pure second- and fourth-order schemes is correctly obtained. The gradient and skew-symmetric formulations of the convective term give exactly the same results. The combined scheme PADE/CDS shows a nearly fourth-order behavior for small grid sizes, but degrades to a second-order convergence for fine grids.

The efficiency of the numerical method is demonstrated by comparing the numerical error of the solution, obtained at a fixed time, and the computational time (CPU) of the simulation for various number of grid cells. Of course, the computational time depends on many parameters that are difficult to control (programming style, computer architecture and optimization, etc.). To obtain a reasonable comparison, all simulations are performed with the same software, on the same computer (an Intel-based Linux workstation) and each numerical method uses the same multigrid solver to calculate the pressure. Results are given in Figure 5 for the vortex dissipation and for the inviscid vortex advection. High-order schemes are clearly more efficient in the viscous flow problem than the second-order scheme. The differences are less pronounced in the inviscid flow problem, although the fully fourth-order schemes perform better for the highly accurate and time-consuming simulations. For coarse grids (small computational times), where the simulations are moderately resolved, the second-order scheme combined with a fourth-order discretization of the

DOI: 10.1002/fld

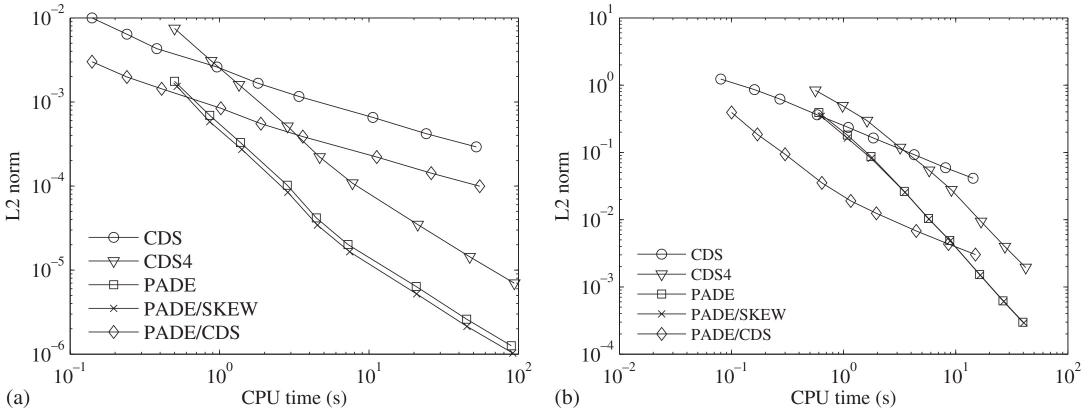

Figure 5. Numerical error on the u-velocity field plotted as a function of the computational time for different grid sizes and numerical schemes: (a) vortex dissipation and (b) inviscid vortex advection.

convection and viscous terms (PADE/CDS) seems the most efficient method. Indeed, the use of second-order scheme for the pressure gradient and continuity equation avoids costly iterations of the pressure solver, whereas the high-order treatment of convection and viscous effects improves the overall accuracy.

The time spent on the elliptic solver depends considerably on the physical problem and the grid size, but varies roughly between 80 and 95%. There is not a significant difference between the percentages found for the second- and fourth-order schemes, although the actual CPU time per time step can be as much as eight times larger for the higher-order scheme. Grid sizes are sufficiently small in these tests so that the code performance is not affected by memory access limitations. For larger grids, on the other hand, the use of a cache-optimized Gauss–Seidel smoother [38] in the multigrid solver is found to considerably reduce the computational time with respect to a straightforward implementation of the algorithm. For example, a time economy of 17% is obtained on a 512×512 grid with the PADE scheme for the problem presented in Section 4.4. Further reduction of the computational time may be obtained with more sophisticated elliptic solvers, but this has not been investigated.

4.3. Inviscid double jet flow

The double jet problem, used for example in [39], is an excellent case to verify the precision and stability of the numerical scheme at high wavenumbers. The computational domain of the size $[0,2\pi]\times[0,2\pi]$ is periodic in both directions. The initial conditions of the flow are defined by

$$u(x,y,0)=\begin{cases}\tanh[(y-\pi/2)/\delta] & \text{for } y\leqslant\pi\\ \tanh[(3\pi/2-y)/\delta] & \text{for } y>\pi\end{cases}$$

$$v(x,y,0)=\varepsilon\sin(x)$$

$$p(x,y,0)=0$$

where $\delta=\pi/15$ is the shear layer thickness and $\varepsilon=0.05$ is the amplitude of a small transverse perturbation. This perturbation will grow with time and leads to a roll up of the shear layer. As time

DOI: 10.1002/fld

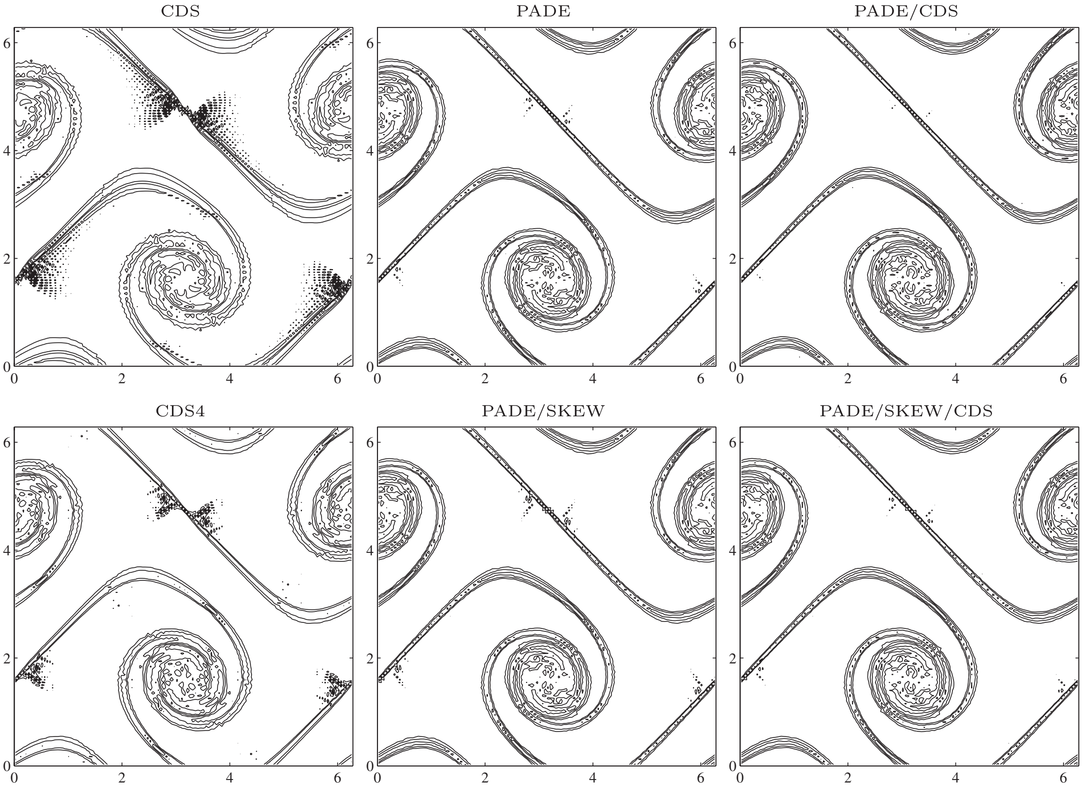

Figure 6. Double jet flow: iso-contours ($\pm 1, \pm 2, \ldots, \pm 6$) of the vorticity at $t = 8$ calculated with different second-order and fourth-order numerical methods (see Table II for details).

evolves, shear layers are stretched and grow thinner and eventually the numerical method will be unable to solve the smallest flow structures. In the absence of viscosity, the stability of the method depends considerably on the discretization of the non-linear terms.

The flow calculated at $t = 8$ on a uniform mesh of size 128×128 is shown in Figure 6. Clearly, the second-order CDS scheme has some difficulty in capturing the thin shear layers, resulting in unphysical noise visible in the shear region and from the blurring effect on shear layers in the vortex core. The explicit fourth-order CDS4 scheme performs slightly better, although the noise persists in the same shear region. The best results are obtained with the compact schemes (PADE/$\bullet$): the shear layers are thinner and better resolved than with previous schemes and numerical noise is much lower. At all times, compact schemes perform best and results obtained with the divergence (PADE) and skew-symmetric (PADE/SKEW) form of the convective term are very similar. Finally, the combined schemes PADE/CDS and PADE/SKEW/CDS seem to perform very well in this case and produce results similar to those obtained with the fully fourth-order compact scheme, although the former are much cheaper.

In order to verify the conservation properties of the compact schemes, the numerical error on the total kinetic energy is plotted in Figure 7(a) as a function of time. The fully conservative explicit central difference schemes (CDS, CDS4) exhibit a slight decrease in the kinetic energy,

DOI: 10.1002/fld

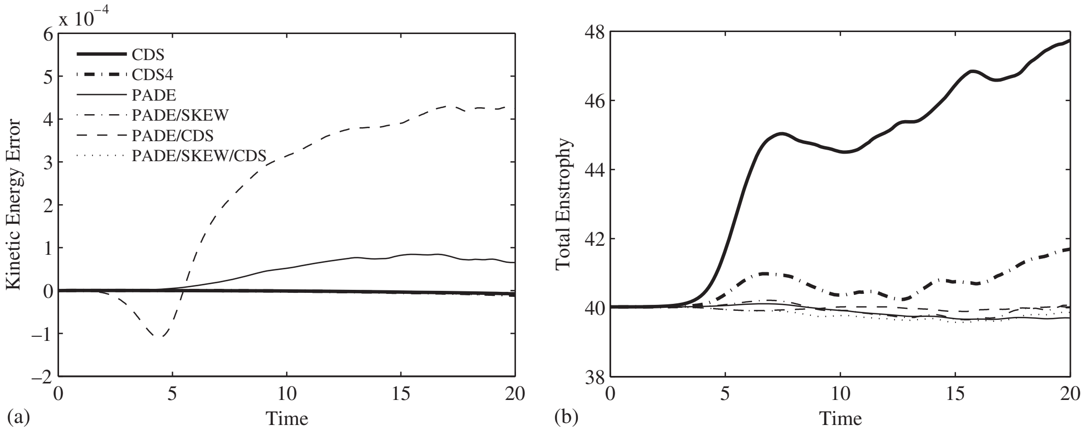

Figure 7. Double jet flow: evolution of the total kinetic energy error (a) and the total enstrophy (b) for the explicit central (CDS/•) schemes (thick lines) and central compact (PADE/•) schemes (thin lines).

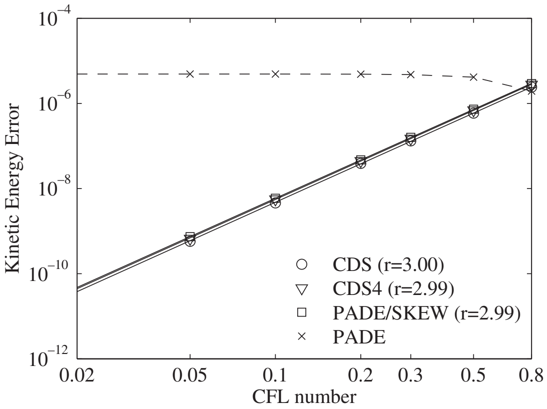

Figure 8. Double jet flow: time step convergence of the total kinetic energy error at $t = 5$ obtained with semi-implicit Runge–Kutta/Crank–Nicolson time stepping and different spatial discretization schemes. Solid lines indicate a power law fittings with indicated exponents r.

which is attributed to the numerical dissipation introduced by the Runge–Kutta time stepping (see also below). The explicit central difference schemes (CDS, CDS4) are thus stable in the L_2-norm, despite the numerical noise appearing on the flow field. The skew-symmetric version of the compact schemes (PADE/SKEW) follows nearly the same trend, which can be expected considering the fully conservative property of these schemes. In contrast, the divergence formulation of the convective terms (PADE, PADE/CDS) is clearly not conservative and the energy initially increases, although the simulation does not seem to 'blow up' for longer times. Note that the flow structures after roughly $t = 10$ are highly underresolved, the grid cells being too coarse to capture the finest gradients of the solution; the simulation is continued to observe the long time stability of the numerical methods. Even at this point, the error on the total energy remains relatively small and results obtained with all compact schemes are very similar, which is confirmed by the evolution of the total enstrophy shown in Figure 7(b). Note, however, that the worse violation of energy conservation is obtained by the combined scheme PADE/CDS, despite the promising results in Figures 6 and 7(b).

In order to show that the decrease in total kinetic energy is related to the time integration scheme, the simulations are repeated for different values of the time step by varying the CFL number. Figure 8 shows the time step convergence of the total kinetic energy error at $t=5$. For the fully conservative schemes (CDS, CDS4 and PADE/SKEW), the third-order convergence of the Runge–Kutta time stepping scheme is correctly recovered. Other results (not shown here) obtained using the mid-point rule for time integration also confirm the conservation property of these schemes in both space and time since a constant kinetic energy at all times is obtained within machine accuracy. On the other hand, results shown in Figure 8 obtained with the PADE scheme do not converge to zero since the spatial scheme introduces a conservation error that is not related to the time integration scheme.

4.4. *Two-dimensional dipole–wall interaction*

The problem of the dipole–wall interaction was proposed in [40] as a benchmark for the two-dimensional unsteady viscous flow simulations. The initial flow field is given by

$$u(x,y,0)=\tfrac{1}{2}|\omega_e|[-(y-y_1)\exp(-(r_1/r_0)^2)+(y-y_2)\exp(-(r_2/r_0)^2)]$$

$$v(x,y,0)=\tfrac{1}{2}|\omega_e|[(x-x_1)\exp(-(r_1/r_0)^2)-(x-x_2)\exp(-(r_2/r_0)^2)]$$

$$p(x,y,0)=0$$

where $r_0=0.1$ and $r_i^2=(x-x_i)^2+(y-y_i)^2$, $i=1,2$. This flow field represents two counter-rotating vortices centered at the locations $(x_1,y_1)=(0,0.1)$ and $(x_2,y_2)=(0,-0.1)$. The vortex strength $|\omega_e|\simeq 300$ is calculated from the condition that the initial total kinetic energy is equal to 2. The computational grid is spanning the domain defined by $-1\leqslant x\leqslant 1$ and $-1\leqslant y\leqslant 1$ with no-slip wall boundary conditions at all sides. Calculations are carried out at the Reynolds number of $Re=2500$. The vortex dipole initially splits up into two dipoles, the strongest of which propagates towards the right-hand side boundary, located at $x=1$. The collision of the dipole with the wall generates a complicated flow pattern. An accurate simulation of the near-wall interaction is necessary in order to reproduce accurately these flow structures. In Reference [40], a uniform grid is used and grid convergence is obtained on a 2048×2048 grid with a second-order finite-difference method. In this study, the grid is refined in the vicinity of the four walls in order to reduce the total number of grid points. The objective here is to find the numerical configuration that requires a minimum of computational time. Several parameters are varied: the order of the numerical scheme, the number of grid points and the grid spacing at the wall. Table III sums up the most important simulations, indicating also the grid spacing at the wall, the CFL number and the computational time of the calculation. Simulations with a CFL number below 0.7 are calculated using a fully explicit time scheme in order to reduce the computational time.

Several criteria are used to conclude whether a simulation is resolved or not. Figure 9 shows the vorticity field on a part of the domain at $t=1$ for the simulations S22, S33 and S44. The third simulation is calculated on a fine grid with fourth-order accuracy and gives results identical to those obtained in [40]. From a visual comparison it is seen that the simulation S22 is clearly not well resolved. For example, the vortex structure located around $(x,y)=(0.72,0.24)$ in Figure 9(c) is not observed in Figure 9(a). Furthermore, the location of the secondary vortex pair at $x\simeq 0.48$ on

DOI: 10.1002/fld

Table III. Overview of the simulations of the dipole–wall interaction.

ID	Scheme	Grid size	Δx_{wall}	CFL	CPU time (h)	Resolved
S21	CDS	1024×1024	1.95×10^{-3}	0.7	13.9	(a)
S22	CDS	1024×1024	1.04×10^{-3}	0.52	16.5	(a)
S23	CDS	1024×1024	0.54×10^{-3}	0.7	43.8	(b)
S24	CDS	1536×1536	0.69×10^{-3}	0.34	65.4	(b)
S25	CDS	1536×1536	0.47×10^{-3}	0.7	109	(c)
S31	PADE/CDS	512×512	1.45×10^{-3}	0.48	2.8	(a)
S32	PADE/CDS	512×512	1.12×10^{-3}	0.28	5.9	(b)
S33	PADE/CDS	768×768	1.39×10^{-3}	0.7	11.2	(b)
S34	PADE/CDS	1024×1024	1.04×10^{-3}	0.49	23.1	(b/c)
S41	PADE	512×512	3.91×10^{-3}	0.7	8.0	(a)
S42	PADE	512×512	2.09×10^{-3}	0.7	8.4	(b)
S43	PADE	512×512	1.45×10^{-3}	0.48	9.4	(c)
S44	PADE	768×768	1.39×10^{-3}	0.67	25.3	(c)

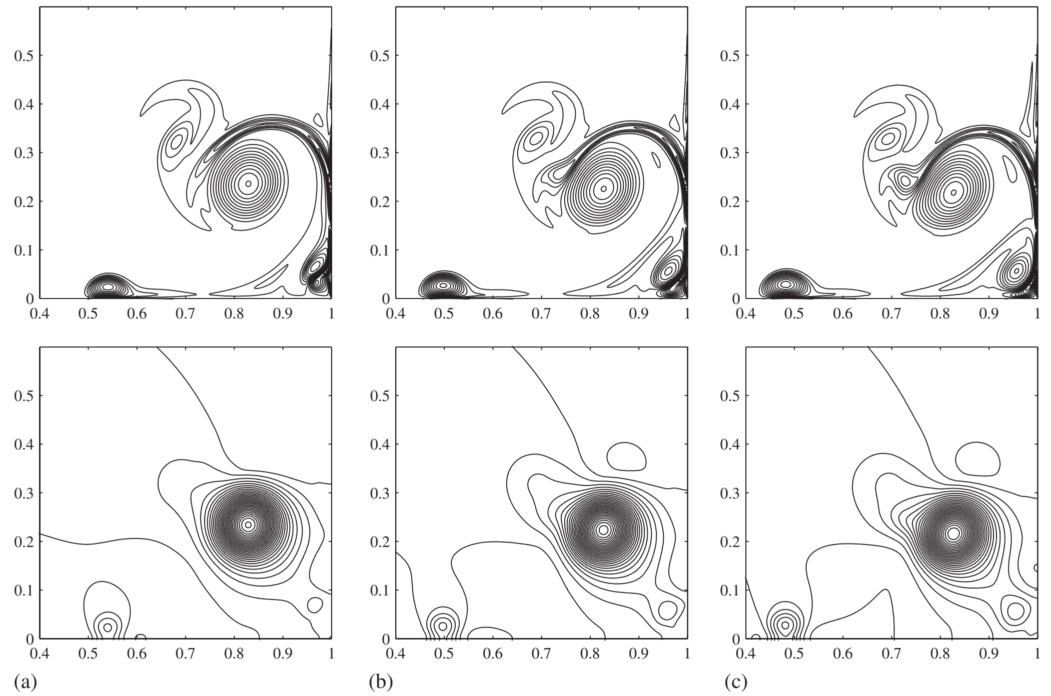

Figure 9. Dipole–wall interaction: iso-contours of the vorticity (upper) and the pressure (lower) at $t=1$ for the cases S22 (a), S33 (b) and S44 (c). The vorticity and pressure contour lines are drawn for every 20 and 1 dimensionless units, respectively.

DOI: 10.1002/fld

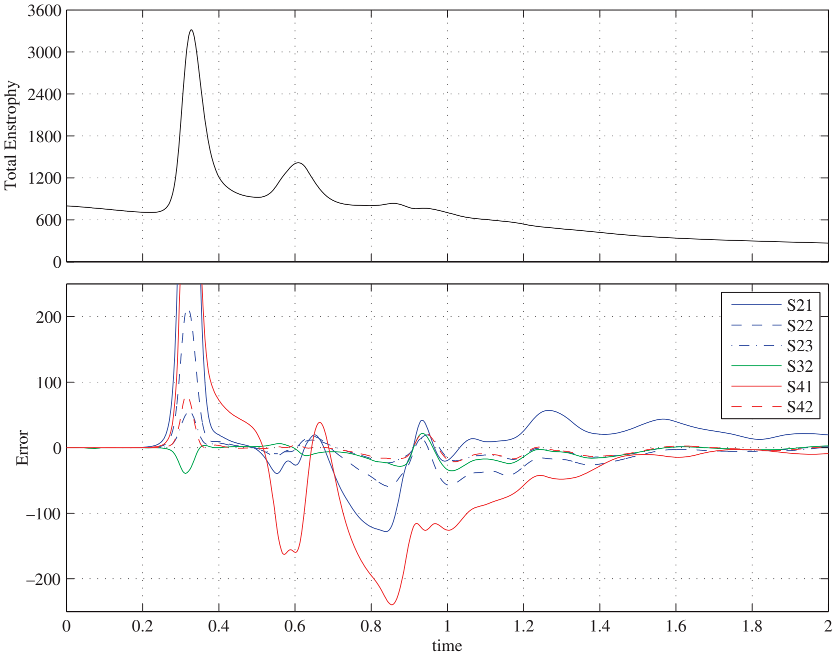

Figure 10. Dipole–wall interaction: evolution of the total enstrophy obtained on the finest grid with the fourth-order method (upper figure) and a comparison of the differences obtained with other simulations (lower figure).

the centerline is substantially different in Figure 9(a). The simulation S33 in Figure 9(b) reproduces many of the characteristic flow structures of the resolved simulation S44, but small differences can nonetheless be observed, for example, on the pressure field near the centerline. The last column in Table III indicates whether the numerical solution is sufficiently resolved (c), underresolved (a) or with only slight differences such as in the case S33 (b). The differences between simulations can also be observed in Figure 10, which shows the evolution of the total enstrophy.

Results in Table III indicate that for the second-order (CDS) scheme, a grid spacing smaller than $\Delta x \simeq 0.7\times 10^{-3}$ is necessary to resolve the finest structures near the wall, whereas for the compact scheme a grid spacing of $\Delta x = 1.45\times 10^{-3}$ is sufficient. Furthermore, a fairly accurate simulation (b) is obtained with the compact scheme on a grid containing four times less grid points and requiring only a fifth of the computational time compared with the second-order method. The fourth-order (PADE) scheme is therefore clearly more efficient for this flow problem. The mixed second- and fourth-order scheme (PADE/CDS) performs well in this case on a 512×512 grid and requires, therefore, even less computational time than the fourth-order scheme. However, to obtain a very accurate simulation (c), the fourth-order scheme is clearly the most efficient.

4.5. Two-dimensional linear instability in plane channel flow

This test case demonstrates the accuracy of the compact scheme for the calculation of the linear instability in plane channel flow. Orszag [41] evaluated the critical Reynolds number for this flow at $Re_{\mathrm{c}} = 5772.22$. Flow disturbances appear in the form of the well-known Tollmien–Schlichting

DOI: 10.1002/fld

waves. The first unstable mode appears at the non-dimensional wavenumber of $\alpha = 1.02056$ and is shown in Figure 11(b). It is seen that the unstable mode contains sharp gradients near the channel walls, requiring an accurate solution of the flow in this region. The undisturbed flow is the Poiseuille flow given by

$$\begin{aligned} u(x,y) &= 1-(y/\delta)^2 \\ v(x,y) &= 0 \\ p(x,y) &= -2x/\delta Re \end{aligned} \tag{36}$$

where $Re_c = U_c\delta/v$ is the Reynolds number based on the centerline velocity and the channel half width δ. For this case, a periodic domain of non-dimensional size $L_x = 2\pi/\alpha$ is used in the streamwise direction x, therefore, only modes with wavenumbers of α or multiples thereof are allowed to develop. The mean flow is sustained by adding a constant pressure gradient of $\partial p/\partial x = -2/\delta Re_c$ to the x-momentum equation. The grid in the streamwise direction x is regular, whereas the grid point distribution in the wall-normal direction y is given by

$$y_{i+1/2} = \frac{\delta \tanh(\gamma(2i-N_y)/N_y)}{\tanh(\gamma)} \tag{37}$$

with $0 \leqslant i \leqslant N_y$ and $\gamma = 2.2$. The initial flow consists of a random noise of amplitude 10^{-4} which is added to the laminar solution of Equation (36). Main parameters of the two cases considered in this paper are summed up in Table IV. The case R1 corresponds to the best approximation of the critical Reynolds number, for which the growth rate of the most unstable mode is nearly zero. The second case is for the supercritical Reynolds number $Re_c = 10\,000$. Figure 11(a) shows a typical result of the evolution of the energy of the disturbance for the critical Reynolds number and the supercritical $Re_c = 10\,000$. After roughly 60 time units, most of the random noise energy is dissipated and only the most unstable mode remains. The growth rate of the amplitude of this mode is in excellent agreement with the value obtained from linear stability theory. Figure 11(b) shows iso-contours of the u-velocity disturbance for the case R1.

Figure 12 shows the error on the growth rate obtained with the different second- and fourth-order schemes as a function of the number of grid cells N in the spanwise direction. The ratio of the numbers of grid cells in the spanwise direction to streamwise direction is equal to 2 for all calculations. The growth rate is estimated from the slope of the linear part of the $\langle v\rangle^2$ curve in Figure 11(a). The result illustrates the efficiency of the fourth-order compact scheme for this case with respect to second-order schemes. The scheme PADE/SKEW seems to have a slight advantage over the non-conservative scheme PADE. The error curve is rather erratic, but in the mean close to the expected fourth-order behavior. Note that the minimum error on the growth rate obtained with the compact schemes is of the order of 10^{-7}, which is close to the round-off error on the theoretical value. Results obtained with the second-order central scheme (CDS) are rather

Table IV. Overview of the channel flow instability simulations.

Case	Re_c	α	Amplitude growth rate [41]
R1	5772.22	1.02056	$<10^{-8}$
R2	10000	1	3.73967×10^{-3}

DOI: 10.1002/fld

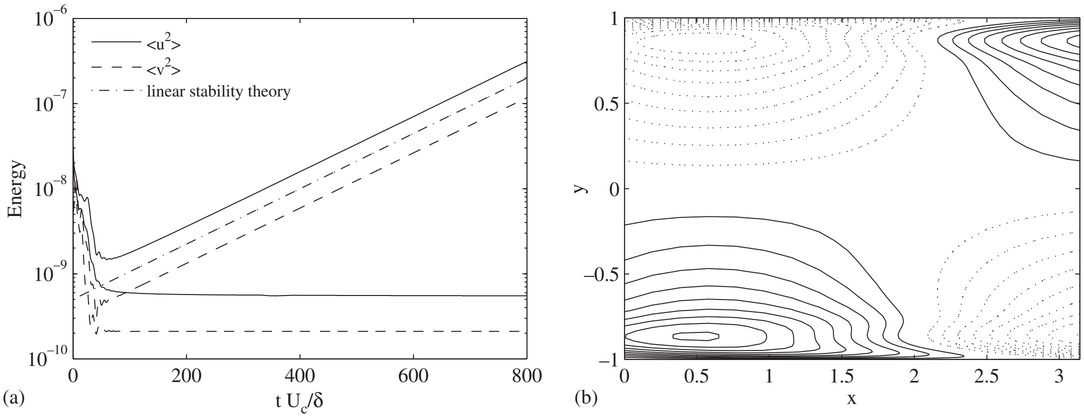

Figure 11. Linear instability in plane channel flow: (a) time evolution of the energy of the velocity disturbances for cases R1 (lower curves) and R2 (upper curves) and (b) iso-contours of the instantaneous u-velocity disturbance at the critical Reynolds number. Results are obtained with the PADE scheme on a 32×64 grid.

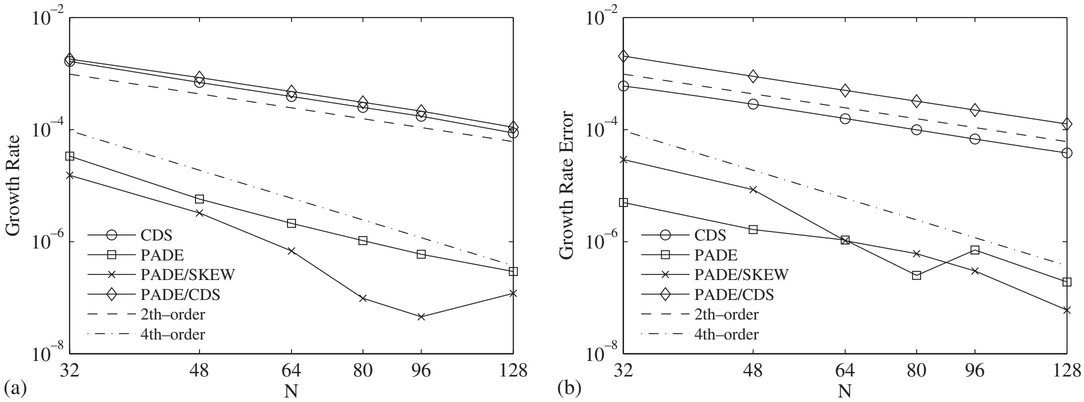

Figure 12. Linear instability in plane channel flow: error on the growth rate of the most unstable mode for cases R1 (a) and R2 (b), plotted as a function of the number of grid cells in the spanwise direction. Dashed and dashed-dotted lines indicate slopes for second- and fourth-order convergences.

disappointing. The second-order behavior is correctly recovered, but the numerical error remains unacceptable large even on the finest grids. Finally, the combined scheme PADE/CDS leads to an error which is even larger than the second-order scheme. This might be the result of the large energy conservation error generated by this scheme, already observed in Section 4.3.

4.6. *Turbulent channel flow*

Several plane channel flow simulations are performed at a Reynolds number based on the mean velocity and the channel height of $Re_b = 5600$, which corresponds to a Reynolds number based on the friction velocity of about $Re_\tau \simeq 180$. The objective is to put the numerical method in

DOI: 10.1002/fld

a more realistic DNS or LES type of framework where the most important turbulent flow features are captured with a minimum of computer resources on relative coarse grids. The size of the computational domain is given by $L_x = \pi\delta$, $L_y = 2\delta$ and $L_z = 0.3\pi\delta$, which corresponds to the minimal flow unit with unforced sustained turbulence according to Jiménez and Moin [42]. The grid in the stream- and spanwise directions x and z is regular and periodic, whereas the grid point distribution in the wall-normal direction is given by Equation (37). The flow is solved on $32\times128\times24$ grid cells, corresponding to the coarsest grid on which turbulence could be maintained for all schemes without forcing the flow. The grid spacing in terms of wall units is $\Delta x^+ = 18$, $\Delta z^+ = 7$, $\Delta y^+ = 0.3$ near the wall and $\Delta y^+ = 6.3$ at the center of the channel. The resolution in the wall-normal direction is in agreement with other channel studies [43], but significantly coarser than the near wall value of $\Delta y^+ = 0.05$ used in References [20, 42]. This could explain why these authors used 16 cells in the spanwise direction, whereas it resulted in a laminar channel flow when using our numerical method.

The total mass flux in the streamwise direction $\dot{M}_x$ is kept nearly constant throughout the simulation by imposing a uniform volume force f_x. This force is calculated from the total force balance given by

$$\frac{\mathrm{d}\dot{M}_x}{\mathrm{d}t} = \sum f_x^{\mathrm{bnd}} + f_x V = \frac{\dot{M}_0 - \dot{M}_x}{\tau} \tag{38}$$

where f_x^{bnd} is the total shear force at the upper or lower wall, V is the total volume, $\dot{M}_0$ is the initial total mass flux and τ is a relaxation parameter. An exact discrete equivalence of Equation (38) is obtained by integration of the discrete momentum equation using the weighted sum defined by Equation (25). A relaxation of $\tau = 10\delta/U_b$ is used in this study, which keeps the total mass flux within 0.0015% of the initial value $\dot{M}_0$. The first simulation is started from an artificial pseudo-turbulent flow field. A statistically steady turbulent flow is obtained after $t \simeq 100\delta/U_b$ and flow statistics are collected for $t > 1000\delta/U_b$. The time step is fixed at $\Delta t = 0.031\delta/U_b$, corresponding to $\Delta t^+ = 0.4$, and the time is advanced using the RK/CN scheme with implicit treatment of the viscous fluxes in the wall-normal direction. Identical results have been obtained with smaller time steps, but are not shown for the sake of brevity.

Table V summarizes the properties of the channel flow simulations and indicates the shear Reynolds number calculated from the mean shear stress at the wall. The shear Reynolds number is slightly low compared with the values obtained by Ham *et al.* [20] who used a conservative second-order scheme, but the difference could be attributed to their use of an implicit fully conservative time scheme and a finer grid spacing near the wall of $\Delta y^+ = 0.05$.

The results of turbulence intensities, Reynolds stresses and mean velocity are plotted in Figure 13. The large-scale turbulent correlations are not completely resolved due to the size of the

Table V. Overview of the turbulent channel simulations at $Re_b = 5600$.

Scheme	L_x	L_z	Grid size	Re_τ	U_b/u_τ
CDS	$\pi\delta$	$0.3\pi\delta$	$32\times128\times24$	182.5	15.34
PADE	$\pi\delta$	$0.3\pi\delta$	$32\times128\times24$	174.8	16.02
PADE/CDS	$\pi\delta$	$0.3\pi\delta$	$32\times128\times24$	175.9	15.92
PADE	$\pi\delta$	$0.3\pi\delta$	$64\times128\times48$	176.0	15.91
KMM [44]	$4\pi\delta$	$\frac{4}{3}\pi\delta$	$128\times129\times128$	178.13	—

DOI: 10.1002/fld

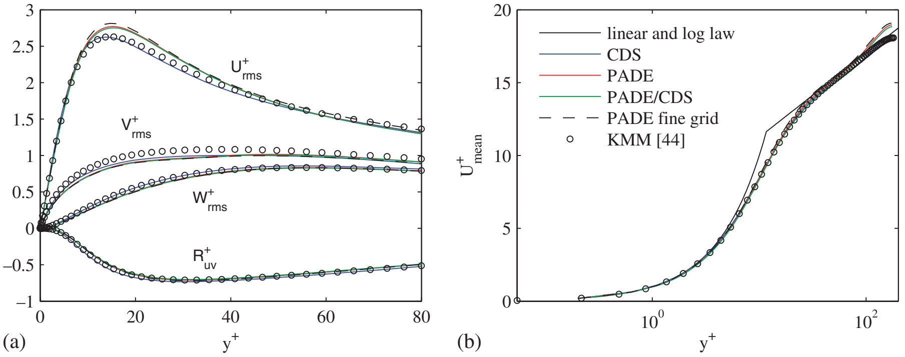

Figure 13. Turbulent channel flow: (a) velocity fluctuations and Reynolds shear stress profiles obtained with different schemes and grid sizes and (b) mean velocity profile near the wall. The solid black line is plotted from $U^{+}_{\mathrm{mean}}=\min\{y^{+}, 2.5\log y^{+}+5.5\}$.

computational domain, but nonetheless all schemes are able to capture the small-scale turbulent physics and many features of a fully developed turbulent channel flow are retrieved in Figure 13; for comparison, the DNS data from Kim, Moin and Moser (KMM) [44] are also shown in the figure. The circles indicate results obtained on the finest grid with fourth-order accuracy. Considering these data as the most reliable, it is seen that most results collapse and only the second-order central schemes deviate slightly from these results, in particular for the streamwise velocity fluctuations and the mean velocity profile. The discrepancy in the mean velocity profile is mainly due to the difference in the shear velocity $u_{\tau}=Re_{\tau}v/\delta$.

It is clear from previous test cases that the second-order central difference schemes require much less computational effort than the fourth-order compact scheme. It seems here that no particular benefit is obtained with the more accurate compact schemes: the same grid size is used, main turbulent flow structures are also reproduced with the lower-order scheme and the differences in one-point statistical correlations are small. Similar conclusions have already been drawn by Vedy *et al*. [26] from LES simulations of sudden expansion plane channel and the backward facing step flows.

5. CONCLUSIONS

A fourth-order compact scheme for unsteady incompressible viscous flows is presented and tested against other existing schemes. The results confirm the efficiency of the compact schemes with respect to other explicit finite-difference schemes for the simulation of incompressible flows on staggered grids. The numerical method used in this study is accurate and robust, and a fully conservative scheme is obtained with the skew-symmetric formulation of the convective terms. The divergence formulation of the compact scheme, which does not strictly conserve kinetic energy, has proven to be stable in practical simulations and shown to give nearly the same results. It has the advantage of being easily extended to flow simulations with variable properties.

In the framework of DNS of turbulent channel flows, the standard second-order discretization seems to capture correctly the turbulent physics even on relative coarse grids. Apparently, no

DOI: 10.1002/fld

clear advantage is obtained using a fourth-order compact scheme. However, applying fourth-order compact discretization for the convection and viscous terms and second-order accuracy for the continuity equation and pressure gradient does seem to increase the accuracy of the method on moderately resolved grids without increasing exceedingly the computational time. In the vortex dissipation and advection problems, this combined scheme allows the use of much coarser grids than pure second-order methods, but this method seems to lack the accuracy of the fully fourth-order compact scheme in the dipole–wall interaction and plane channel instability problems.

APPENDIX A

A.1. Solution of Equation (30) using ADI method

The ADI method used in this study is demonstrated below for the first stage of the time stepping procedure on a two-dimensional computational domain. We seek the solution of the following implicit linear system:

$$U'-\Delta t\beta_1\mathscr{L}(U')=U^n+\Delta t[\beta_1\mathscr{L}(U^n)+\sigma_1\mathscr{N}(U^n)-2\beta_1 GP^n+\sigma_1 f]=F^n$$

where F^n denotes the sum of the explicit terms, which are known from the previous time step. The linear operator $\mathscr{L}(U)$ contains the discrete second derivatives of the velocity components in each spatial direction, i.e. $\partial^2 u_i/\partial x_j^2$, which appear in the viscous terms of the momentum equation. The system is solved iteratively. Let U^m denote the approximation of U' after m iterations, then a new approximation U^{m+1} is obtained by solving

$$\begin{aligned} R^m &= F^n - U^m\Delta t\beta_1\mathscr{L}(U^m)\\ \Delta U^{m+1} &= (\mathscr{I}-\Delta t\beta_1\mathscr{L}_{11})^{-1}(\mathscr{I}-\Delta t\beta_1\mathscr{L}_{22})^{-1}R^m \\ U^{m+1} &= U^m+\Delta U^{m+1}\end{aligned} \tag{A1}$$

where $\mathscr{L}_{11}$ and $\mathscr{L}_{22}$ denote the second-order approximation of the second derivatives in, respectively, the ξ^1 and ξ^2 directions. For example, the discrete operator $\mathscr{L}_{11}$ for the velocity component u in the direction ξ^1 is given by

$$\mathscr{L}^u_{11,i+1/2}=\nu\xi^1_{x,i+1/2}[\xi^1_{x,i+1}(u_{i+3/2}-u_{i+1/2})-\xi^1_{x,i}(u_{i+1/2}-u_{i-1/2})] \tag{A2}$$

Equation (A1) then leads to a set of decoupled tridiagonal systems for each velocity component and spatial direction that are solved using the Thomas algorithm. In the actual method, the order of solving the tridiagonal systems in different directions is alternated at each Runge–Kutta stage and time step to avoid numerical errors to accumulate in one direction. Iterations are continued until the maximum absolute value of the residual R^m drops below a threshold of typically 10^{-10}.

ACKNOWLEDGEMENTS

The author acknowledges the support from IDRIS (Institut de Développement et de Ressources en Informatique Scientifique, CNRS) and FLCHP (Fédération Lyonnaise de Calcul Haute Performance) where part of the computations were performed.

REFERENCES

1. Ladeinde F, Cai X, Visbal MR, Gaitonde DV. Turbulence spectra characteristics of high order schemes for direct and large eddy simulation. *Applied Numerical Mathematics* 2001; **36**(4):447–474.
2. Nagarajan S, Lele SK, Ferziger JH. A robust high-order compact method for large eddy simulation. *Journal of Computational Physics* 2003; **191**:392–419.
3. Visbal MR, Gaitonde DV. On the use of higher-order finite-difference schemes on curvilinear and deforming meshes. *Journal of Computational Physics* 2002; **181**(1):155–185.
4. Peyret R. *Spectral Methods for Incompressible Viscous Flow*. Springer: New York, 2002.
5. Meitz HL, Fasel HF. A compact-difference scheme for the Navier–Stokes equations in vorticity–velocity formulation. *Journal of Computational Physics* 2000; **157**(1):371–403.
6. Shukla RK, Tatineni M, Zhong X. Very high-order compact finite difference schemes on non-uniform grids for incompressible Navier–Stokes equations. *Journal of Computational Physics* 2007; **224**(2):1064–1094.
7. Kravchenko AG, Moin P. On the effect of numerical errors in large eddy simulations of turbulent flows. *Journal of Computational Physics* 1997; **131**(2):323–326.
8. Meyers J, Geurts BJ, Sagaut P. A computational error-assessment of central finite-volume discretizations in large-eddy simulation using a Smagorinsky model. *Journal of Computational Physics* 2007; **227**(1):156–173.
9. Lele SK. Compact finite difference schemes with spectral-like resolution. *Journal of Computational Physics* 1992; **103**(1):16–42.
10. Wilson RV, Demuren AO, Carpenter M. Higher-order compact schemes for numerical simulation of incompressible flows. *Technical Report NASA/CR-1998-206922 ICASE Report No. 98-13*, Institute for Computer Applications in Science and Engineering, 1998.
11. Demuren AO, Wilson RV, Carpenter M. Higher-order compact schemes for numerical simulation of incompressible flows. Part I: Theoretical development. *Numerical Heat Transfer*, *Part B* 2001; **39**(3):207–230.
12. Yanwen M, Dexun F, Kobayashi T, Taniguchi N. Numerical solution of the incompressible Navier–Stokes equations with an upwind compact difference scheme. *International Journal for Numerical Methods in Fluids* 1999; **30**:509–521.
13. Zhu Q-Y, Chan CK. High-order upwind compact scheme and simulation of turbulent premixed V-flame. *International Journal for Numerical Methods in Fluids* 2005; **49**(7):701–720.
14. Mittal R, Moin P. Suitability of upwind-biased finite difference schemes for large-eddy simulation of turbulent flows. *AIAA Journal* 1997; **35**(8):1415–1417.
15. Grinstein FF, Fureby C. Recent progress on miles for high Reynolds number flows. *Journal of Fluids in Engineering* 2002; **124**(4):848–861.
16. Morinishi Y, Lund TS, Vasilyev OV, Moin P. Fully conservative higher order finite difference schemes for incompressible flow. *Journal of Computational Physics* 1998; **143**:90–124.
17. Verstappen RWCP, Veldman AEP. Symmetry-preserving discretization of turbulent flow. *Journal of Computational Physics* 2003; **187**(1):343–368.
18. Mansour NN, Moin P, Reynolds WC, Ferziger JH. Improved methods for large eddy simulations of turbulence. *Turbulent Shear Flows* 1979; **1**:368–401.
19. Harlow FH, Welch JE. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids* 1965; **8**:2182–2189.
20. Ham FE, Lien FS, Strong AB. A fully conservative second-order finite difference scheme for incompressible flow on nonuniform grids. *Journal of Computational Physics* 2002; **177**:117–133.
21. Fukagata K, Kasagi N. Highly energy-conservative finite difference method for the cylindrical coordinate system. *Journal of Computational Physics* 2002; **181**:478–498.
22. Morinishi Y, Vasilyev OV, Takeshi O. Fully conservative finite difference scheme in cylindrical coordinates for incompressible flow. *Journal of Computational Physics* 2004; **197**:686–710.
23. Gamet L, Ducros F, Nicoud F, Poinsot T. Compact finite difference schemes on non-uniform meshes. Application to direct numerical simulations of compressible flows. *International Journal for Numerical Methods in Fluids* 1999; **29**(2):159–191.
24. Van Os JJAM, Uittenbogaard RE. Towards the ultimate variance-conserving convection scheme. *Journal of Computational Physics* 2004; **197**(1):197–214.
25. Schiestel R, Viazzo S. A Hermitian–Fourier numerical method for solving the incompressible Navier–Stokes equations. *Computers and Fluids* 1995; **24**(6):739–752.
26. Vedy E, Viazzo S, Schiestel R. A high-order finite difference method for incompressible fluid turbulence simulations. *International Journal for Numerical Methods in Fluids* 2003; **42**(11):1155–1188.

DOI: 10.1002/fld

27. Brüger A, Nilsson J, Kress W. A compact higher order finite difference method for the incompressible Navier–Stokes equations. *Journal of Scientific Computing* 2002; **17**:551–560.
28. Brüger A, Gustafsson B, Lotstedt P, Nilsson J. High order accurate solution of the incompressible Navier–Stokes equations. *Journal of Computational Physics* 2005; **203**(1):49–71.
29. Abide S, Viazzo S. A 2D compact fourth-order projection decomposition method. *Journal of Computational Physics* 2005; **206**(1):252–276.
30. Kampanis NA, Ekaterinaris JA. A staggered grid, high-order accurate method for the incompressible Navier–Stokes equations. *Journal of Computational Physics* 2006; **215**(2):589–613.
31. Pereira JMC, Kobayashi MH, Pereira JCF. A fourth-order-accurate finite volume compact method for the incompressible Navier–Stokes solutions. *Journal of Computational Physics* 2001; **167**:217–243.
32. Piller M, Stalio E. Finite-volume compact schemes on staggered grids. *Journal of Computational Physics* 2004; **197**:299–340.
33. Spalart PR, Moser RD, Rogers MM. Spectral methods for the Navier–Stokes equations with one infinite and two periodic directions. *Journal of Computational Physics* 1991; **96**(2):297–324.
34. Le H, Moin P. An improvement of fractional step methods for the incompressible Navier–Stokes equations. *Journal of Computational Physics* 1991; **92**(2):369–379.
35. Pozrikidis C. A note on the regularization of the discrete Poisson–Neumann problem. *Journal of Computational Physics* 2001; **172**(2):917–923.
36. Swarztrauber PN. Direct method for the discrete solution of separable elliptic equations. *SIAM Journal on Numerical Analysis* 1974; **11**:1136–1150.
37. Wesseling P. *An Introduction to Multigrid Methods*. R T Edwards, Inc.: Philadelphia, 2004.
38. Kowarschik M, Rüde U, Weiß C, Karl W. Cache-aware multigrid methods for solving Poisson's equation in two dimensions. *Computing* 2000; **64**:381–399.
39. Wilson RV, Demuren AO, Carpenter M. Higher-order compact schemes for numerical simulation of incompressible flows. Part II: Applications. *Numerical Heat Transfer*, *Part B* 2001; **39**(3):231–255.
40. Clercx HJH, Bruneau C-H. The normal and oblique collision of a dipole with a no-slip boundary. *Computers and Fluids* 2006; **35**(3):245–279.
41. Orszag SA. Accurate solution of the Orr–Sommerfeld stability equation. *Journal of Fluid Mechanics* 1971; **50**(4):689–703.
42. Jiménez J, Moin P. The minimal flow unit in near-wall turbulence. *Journal of Fluid Mechanics* 1991; **225**:213–240.
43. Nicoud F. Conservative high-order finite difference schemes for low-Mach number flows. *Journal of Computational Physics* 2000; **158**:71–97.
44. Kim J, Moin P, Moser R. Turbulent statistics in fully developed channel flow at low Reynolds number. *Journal of Fluid Mechanics* 1987; **177**:133–166.

DOI: 10.1002/fld