Journal of Computational Physics 228 (2009) 5989–6015

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

High-order compact schemes for incompressible flows: A simple and efficient method with quasi-spectral accuracy

Sylvain Laizet [a,b], Eric Lamballais [a,*]

[a] Laboratoire d'Etudes Aérodynamiques UMR 6609, Université de Poitiers, ENSMA, CNRS Téléport 2 – Bd. Marie et Pierre Curie B.P. 30179 86962 Futuroscope Chasseneuil Cedex, France

[b] Turbulence, Mixing and Flow Control Group, Department of Aeronautics and Institute for Mathematical Sciences, Imperial College London, London SW7 2AZ, United Kingdom

ARTICLE INFO

Article history:
Received 8 February 2008
Received in revised form 6 February 2009
Accepted 8 May 2009
Available online 18 May 2009

Keywords:
Direct numerical simulation
Compact schemes
Spectral methods
Incompressible flows
Immersed boundary method

ABSTRACT

In this paper, a finite difference code for Direct and Large Eddy Simulation (DNS/LES) of incompressible flows is presented. This code is an intermediate tool between fully spectral Navier–Stokes solvers (limited to academic geometry through Fourier or Chebyshev representation) and more versatile codes based on standard numerical schemes (typically only second-order accurate). The interest of high-order schemes is discussed in terms of implementation easiness, computational efficiency and accuracy improvement considered through simplified benchmark problems and practical calculations. The equivalence rules between operations in physical and spectral spaces are efficiently used to solve the Poisson equation introduced by the projection method. It is shown that for the pressure treatment, an accurate Fourier representation can be used for more flexible boundary conditions than periodicity or free-slip. Using the concept of the modified wave number, the incompressibility can be enforced up to the machine accuracy. The benefit offered by this alternative method is found to be very satisfactory, even when a formal second-order error is introduced locally by boundary conditions that are neither periodic nor symmetric. The usefulness of high-order schemes combined with an immersed boundary method (IBM) is also demonstrated despite the second-order accuracy introduced by this wall modelling strategy. In particular, the interest of a partially staggered mesh is exhibited in this specific context. Three-dimensional calculations of transitional and turbulent channel flows emphasize the ability of present high-order schemes to reduce the computational cost for a given accuracy. The main conclusion of this paper is that finite difference schemes with quasi-spectral accuracy can be very efficient for DNS/LES of incompressible flows, while allowing flexibility for the boundary conditions and easiness in the code development. Therefore, this compromise fits particularly well for very high-resolution simulations of turbulent flows with relatively complex geometries without requiring heavy numerical developments.

1. Introduction

For fundamental flows in academic configuration, the usefulness of highly accurate numerical schemes for DNS/LES is fully recognized. For very simplified geometries, in terms of accuracy and computational efficiency, the most spectacular gain is obtained using spectral methods based on Fourier or Chebyshev representation [5]. The various combinations between these two types of spectral discretization have allowed numerous authors to consider efficiently some fundamental

* Corresponding author.
E-mail addresses: s.laizet@imperial.ac.uk (S. Laizet), lamballais@univ-poitiers.fr (E. Lamballais).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.05.010

turbulent problems in the context of incompressible flows. Among these fundamental problems, one could cite homogeneous turbulence (full Fourier representation), transitional or turbulent channel flow (mixed Fourier–Chebyshev representation) or turbulence in a cavity (full Chebyshev representation), etc. Unfortunately, for fundamental problems in slightly more complex geometry, the full spectral approach in Fourier or Chebyshev space is not appropriate. Furthermore, despite the recent progress in computational fluid dynamics (CFD) methods, the use of sophisticated meshes combined with high-order schemes remains a challenging task that requires important numerical developments to accommodate accuracy and efficiency. For a general presentation of this ambitious numerical strategy, see [8] where the authors focus on collocation and spectral elements for the simulation of incompressible flows.

To better understand a wide range of transitional or turbulent phenomena [25], many DNS/LES studies of fundamental flows in non-academic geometries remain to be performed. The term non-academic means here that the geometry is not as complex as in a real industrial configuration, while not being simple enough to allow the straightforward use of high-order methods, spectral or not. To treat favourably that kind of flows, this paper presents an intermediate approach leading to a compromise between accuracy of spectral methods and versatility of industrial codes. To our knowledge, the potential of such an incompressible Navier–Stokes solver has not been investigated in the literature. In a complementary way to collocation and spectral elements, the high-order strategy considered in this paper is based on finite difference schemes implemented on a Cartesian mesh, the main advantages of this specific numerical configuration being its simplicity and efficiency. It will be shown that such a numerical code can be significantly more versatile than a fully spectral code while offering in practice a comparable accuracy for the DNS of turbulent flows. Furthermore, to increase the potential of the code, the interest of its combination with an immersed boundary method (IBM) will also be evaluated, so that any solid wall geometry can be freely imposed despite the use of a simple Cartesian mesh.

It is well known that in terms of numerical development and computational efficiency, the enforcement of incompressibility can be critical, especially for high-order methods. In this paper, using a projection method, a fully spectral Poisson solver is proposed, in the three spatial directions, even if the boundary conditions do not seem to suit well for a Fourier representation. The advantages of this technique are discussed in terms of computational efficiency, accuracy and simplicity. Several validations are proposed to show the potential of this approach through the use of high-order schemes, the boundary condition treatment and the choice of the mesh organisation (collocated or partially staggered configuration). The results are analysed in order to target the general context for which quasi-spectral accurate schemes could be really attractive.

The organisation of the paper is as follows. After an overview of the general framework of present numerical methods in Sections 2 and 3, the treatment of the pressure is detailed in Section 4 where the benefit of the modified spectral formalism is argued. Furthermore, an extension toward non-regular meshes is proposed in Section 5. Finally, a set of formal and practical validations is presented and analysed in Section 7 to evaluate the actual benefit of present high-order method.

2. General framework of the numerical method

2.1. Governing equations

The governing equations are the forced incompressible Navier–Stokes equations

$$\frac{\partial \mathbf{u}}{\partial t} = -\nabla p - \frac{1}{2}[\nabla(\mathbf{u}\otimes\mathbf{u}) + (\mathbf{u}.\nabla)\mathbf{u}] + \nu\nabla^2\mathbf{u} + \mathbf{f} \tag{1}$$

$$\nabla.\mathbf{u} = 0 \tag{2}$$

where $p(\mathbf{x},t)$ is the pressure field (for a fluid with a constant density $\rho = 1$) and $\mathbf{u}(\mathbf{x},t)$ the velocity field. In these forced Navier–Stokes equations, the forcing field $\mathbf{f}(\mathbf{x},t)$ is used through an immersed boundary method or sometimes to perform numerical tests where analytical solutions are available thanks to a relevant definition of $\mathbf{f}$. More details about the exact expression of $\mathbf{f}$ will be given in the following, depending on the physical problem considered. Note that convective terms are written in the skew-symmetric form. This specific form is used in this study as it allows the reduction of aliasing errors while remaining energy conserving for the spatial discretization permitted in the code [17].

2.2. Time advancement

The time advancement of Eq. (1) can be expressed as

$$\frac{\mathbf{u}^* - \mathbf{u}^k}{\Delta t} = a_k\mathbf{F}^k + b_k\mathbf{F}^{k-1} - c_k\nabla\tilde{p}^k + c_k\tilde{\mathbf{f}}^{k+1} \tag{3}$$

$$\frac{\mathbf{u}^{**} - \mathbf{u}^*}{\Delta t} = c_k\nabla\tilde{p}^k \tag{4}$$

$$\frac{\mathbf{u}^{k+1} - \mathbf{u}^{**}}{\Delta t} = -c_k\nabla\tilde{p}^{k+1} \tag{5}$$

with

$$\mathbf{F}^k = -\frac{1}{2}[\nabla(\mathbf{u}^k \otimes \mathbf{u}^k) + (\mathbf{u}^k.\nabla)\mathbf{u}^k] + \nu\nabla^2\mathbf{u}^k \tag{6}$$

and

$$\tilde{p}^{k+1} = \frac{1}{c_k\Delta t}\int_{t_k}^{t_{k+1}} p\ dt, \quad \tilde{\mathbf{f}}^{k+1} = \frac{1}{c_k\Delta t}\int_{t_k}^{t_{k+1}} \mathbf{f}\ dt \tag{7}$$

for a Runge–Kutta scheme with coefficient couples a_k, b_k (and $c_k = a_k + b_k$ for simplicity) on n_k sub-time steps, $k = 1, \ldots, n_k$ with $t_1 = t_n$ and $t_{n_k} = t_{n+1}$ ($\Delta t = t_{n+1} - t_n$ being the full time step). In this paper, depending on the flow configuration, two temporal schemes can be used: a third-order Runge–Kutta scheme with $n_k = 3$ (see [42] for the values of a_k, b_k allowing this low-storage) or a second-order Adams–Bashforth scheme with simply $n_k = 1$ and $(a_1, b_1) = (3/2, -1/2)$. In this writing, the pressure and forcing terms are expressed through their time-averaged values on a given sub-step $c_k\Delta t$, indicated by the tilde in $\tilde{p}^{k+1}$ and $\tilde{\mathbf{f}}^{k+1}$.

2.3. Boundary conditions of the computational domain

The governing Eqs. (1) and (2) are directly solved in a computational domain $L_x \times L_y \times L_z$ discretized on a Cartesian mesh of $n_x \times n_y \times n_z$ nodes. On each limit of the domain $x = \pm L_x/2, y = \pm L_y/2$ and $z = \pm L_z/2$, periodic, free-slip, no-slip or open conditions can be applied depending on the flow configuration considered. Periodic or free-slip boundary conditions can be imposed directly via the spatial discretization without specific care in the time advancement. In contrary, the use of Dirichlet conditions on the velocity (for no-slip or open conditions) needs to be defined according to the time advancement procedure. Here, as a three-step fractional step method is used, the Dirichlet boundary conditions on the velocity can be imposed just after the first step (3) directly on $\mathbf{u}^*$, allowing a Δt^2 accuracy on the final velocity $\mathbf{u}^{k+1}$ [6]. In addition, conventional homogeneous Neumann conditions are used to solve the pressure (see the next section for more details about the specific Poisson equation solved to compute the pressure). The interest of the homogeneous Neumann condition for the pressure will be discussed in the Section 4 dedicated to the pressure discretization.

2.4. Pressure treatment and immersed boundary method

In a classical fractional step method, the incompressibility condition (2) can be verified at the end of each sub-time step

$$\nabla.\mathbf{u}^{k+1} = 0 \tag{8}$$

through the solving of a Poisson equation

$$\nabla.\nabla\tilde{p}^{k+1} = \frac{\nabla.\mathbf{u}^{**}}{c_k\Delta t} \tag{9}$$

that provides the estimation of $\tilde{p}^{k+1}$ required to perform the correction (5). In the present work, this conventional incompressibility treatment is used only for academic flows. In order to simulate flows in presence of solid walls with relatively complex geometries, the numerical code can be combined with an immersed boundary method (IBM). IBM is very attractive by avoiding sophisticated meshes and their associated loss of accuracy and increase of computational cost. In this study, a direct forcing method is considered, consisting in the imposition of a target velocity $\mathbf{u_0}(\mathbf{x}, t)$ on the velocity solution $\mathbf{u}(\mathbf{x}, t)$ in the solid body region. In a general IBM, the forcing term $\mathbf{f}$ of Eq. (1) is defined in order to verify more or less carrefully the expected boundary condition at the wall of the solid body.[1] In the present study, using a direct method, the forcing term in (3) can be expressed as

$$c_k\tilde{\mathbf{f}}^{k+1} = \varepsilon\left(-a_k\mathbf{F}^k - b_k\mathbf{F}^{k-1} + c_k\nabla\tilde{p}^k + \frac{\mathbf{u_0}^{k+1} - \mathbf{u}^k}{\Delta t}\right) \tag{10}$$

with $\varepsilon = 1$ in the solid body region and $\varepsilon = 0$ everywhere else. As for a conventional Dirichlet condition, the present definition of $\tilde{\mathbf{f}}^{k+1}$ allows the exact prescription of $\mathbf{u}^*$ in the forcing region so that the final error on $\mathbf{u}^{k+1}$ is second-order in time, namely $\mathbf{u}^{k+1} = \mathbf{u_0}^{k+1} + O(\Delta t^2)$ when $\varepsilon = 1$. The target velocity $\mathbf{u_0}(\mathbf{x}, t)$ is calibrated in order to satisfy the no-slip condition at the wall of the solid body while ensuring the regularity of the velocity field across the immersed surface (see [34,33] for more details). As the target velocity is not necessarily divergence free [34,33] have proposed to solve a specific pressure equation

$$\nabla\ \cdot\ \nabla\tilde{p}^{k+1} = \frac{\nabla.[(1-\varepsilon)\mathbf{u}^{**}]}{c_k\Delta t} \tag{11}$$

where the conventional Poisson Eq. (9) is recovered for $\varepsilon = 0$ whereas inside the solid body, the condition $\varepsilon = 1$ yields the Laplace equation. It is therefore possible to prescribe freely the level of divergence inside the solid body by satisfying a modified divergence condition expressed as

[1] See for instance [2,9,12,35,38,40] to have a non-exhaustive view about the diversity of the techniques already successfully developed, and also [27] for a review.

$$\nabla \cdot \mathbf{u}^{k+1} = \nabla \cdot (\varepsilon \mathbf{u}_0^{k+1}) \tag{12}$$

This internal motion (given by the target velocity) introduces a mass source/sink inside the body, this artificial flow being useful for the regularity of the solution when high-order schemes are used for the spatial differentiation [34,33].

3. Spatial discretization of convective and viscous terms

Let us consider a uniform distribution of n_x nodes x_i on the domain $[0, L_x]$ with $x_i = (i-1)\Delta x$ for $1 \leqslant i \leqslant n_x$. The approximation of values $f'_i = f'(x_i)$ of the first derivative $f'(x)$ of the function $f(x)$ can be related to values $f_i = f(x_i)$ by a finite difference scheme of the form

$$\alpha f'_{i-1} + f'_i + \alpha f'_{i+1} = a\,\frac{f_{i+1} - f_{i-1}}{2\Delta x} + b\,\frac{f_{i+2} - f_{i-2}}{4\Delta x} \tag{13}$$

By choosing $\alpha = 1/3, a = 14/9$ and $b = 1/9$, this approximation is sixth-order accurate while having a so-called "quasi-spectral behaviour" [24] due to its capabilities to represent accurately a wide range of scales. The compromise of the sixth-order accuracy has been chosen to maintain a compact formulation (a three- and five-node stencil for f'_i and f_i, respectively) via the use of a Hermitian structure of the scheme with $\alpha \neq 0$. Note however that the present approach can be straightforwardly adapted to centred finite difference schemes of higher order if necessary.

Analogous relations can be established for the approximation of the second derivative values $f''_i = f''(x_i)$ with

$$\alpha f''_{i-1} + f''_i + \alpha f''_{i+1} = a\,\frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2} + b\,\frac{f_{i+2} - 2f_i + f_{i-2}}{4\Delta x^2} + c\,\frac{f_{i+3} - 2f_i + f_{i-3}}{9\Delta x^2} \tag{14}$$

By choosing $\alpha = 2/11, a = 12/11, b = 3/11$ and $c = 0$, this approximation is sixth-order accurate with the same favourable properties of the first derivative in terms of "spectral-like" resolution [24]. To control the aliasing errors via the viscous term, this type of schemes can also be defined to be over-dissipative at the highest wave numbers (in the spectral range where even a high-order finite difference scheme becomes inaccurate), through the use of a less compact formulation, with for instance $\alpha = 0.47959871686180711$, $a = 0.42090288706093404$, $b = 1.7020738409366740$ and $c = -0.16377929427399390$ that preserves the sixth-order accuracy of the scheme.

For the present code, the partial differentiation of convective and viscous terms represented by $\mathbf{F}^k$ in definition (6) are performed with high-order compact schemes (13) and (14) for the first and second derivatives. As already stressed, four different boundary conditions can be considered in each spatial direction. The periodic and free-slip (equivalent to symmetric and antisymmetric conditions) boundary conditions allow the use of schemes (13) and (14) for all the nodes considered. More precisely, the schemes (13) and (14) just need to be relevantly modified near the borders $i = 1$ and $i = n_x$ through the substitution of the extra-node (sometimes called "ghost") values $f_0, f_{-1}, f_{n_x+1}, f_{n_x+2}$ by their counterparts $f_1, f_2, f_{n_x-1}, f_{n_x-2}$. For a periodic boundary condition, this type of substitution for f, f' and f'' can be written

$$f_0 \to f_{n_x},\ f_{-1} \to f_{n_x-1},\ f'_0 \to f'_{n_x},\ f''_0 \to f''_{n_x}, \tag{15}$$

while symmetric and antisymmetric conditions lead to

$$f_0 \to f_2,\ f_{-1} \to f_3,\ f'_0 \to -f'_2,\ f''_0 \to f''_2, \tag{16}$$

and

$$f_0 \to -f_2,\ f_{-1} \to -f_3,\ f'_0 \to f'_2,\ f''_0 \to -f''_2, \tag{17}$$

respectively. For simplicity, only relations on the left boundary condition (near $i = 1$) are given here, their right counterparts (near $i = n_x$) being easily deduced.

When no-slip or open (i.e. Dirichlet for velocity) boundary conditions are used, nothing is assumed concerning the flow outside the computational domain. Single sided formulations are used for the approximation of first and second derivatives for these types of boundaries using relations of the form

$$\begin{aligned} f'_1 + 2f'_2 &= \frac{1}{2\Delta x}(-5f_1 + 4f_2 + f_3) \\ f''_1 + 11f''_2 &= \frac{1}{\Delta x^2}(13f_1 - 27f_2 + 15f_3 - f_4) \end{aligned} \tag{18}$$

that are third-order accurate [24]. At the adjacent nodes, because a three-point formulation must be used, Padé schemes are employed with

$$\begin{aligned} \frac{1}{4}f'_1 + f'_2 + \frac{1}{4}f'_3 &= \frac{3}{2}\frac{f_3 - f_1}{2\Delta x} \\ \frac{1}{10}f''_1 + f''_2 + \frac{1}{10}f''_3 &= \frac{6}{5}\frac{f_3 - 2f_2 + f_1}{\Delta x^2} \end{aligned} \tag{19}$$

these schemes being fourth-order accurate.

In terms of computational techniques, present compact Hermitian schemes (with $\alpha \neq 0$) for convective and viscous terms just need to inverse tridiagonal matrices. The most expensive derivative calculation is obtained for a periodic direction that needs to inverse a cyclic matrix. However, the computation of derivatives remains cheaper than what is required by pseudo-spectral methods mainly because of the non-linear nature of the convective terms. This difference can be significantly enlarged if additional non-linear terms need to be treated through the use of LES subgrid models for instance. For this reason, even if spectral methods can be used for the treatment of periodic directions, a better compromise can be found in terms of computational efficiency with the compact schemes described above. On the contrary, the spectral approach is clearly more efficient for linear equations. The main subject of the following section deals with the opportunity to take advantage of this favourable property of Fourier methods for the treatment of the Poisson equation.

4. Spatial discretization of the pressure

It is well known that the treatment of incompressibility is a real difficulty to obtain solutions of incompressible Navier–Stokes equations. The unavoidable solving of the Poisson Eqs. (9) and (11) introduced by the fractional step method can be computationally very expensive, especially when high-order schemes are used in combination with iterative techniques. This point is so delicate that in terms of computational cost, the use of second-order schemes could be finally more efficient. Based on high-order finite difference schemes and expressed in physical space, the inversion of a Poisson equation requires sophisticated methods that can be computationally expensive (see [26] for an example of 3D solver of Poisson's equation based on compact schemes). On the contrary, performed in Fourier space, the equivalent operation is much cheaper and easy to implement with the help of Fast Fourier Transform (FFT) routines. In first analysis, it seems logical to think that the Fourier approach for the pressure treatment is restricted to the same set of boundary conditions used for the remaining part of the Navier–Stokes equations, namely periodic or free-slip boundary conditions. However, it is well known that the Fourier formalism can be used advantageously in order to obtain a second-order accurate solution from a Poisson equation with homogeneous Neumann conditions (see for instance [41,39,16]). The strategy suggested in this study is to follow this idea, by evaluating its interest in a numerical code based on compact high-order finite difference schemes. To emphasize the simplicity of the method, its principle is directly presented in the 3D configuration described in Section 4.5.

In the numerical strategy proposed in this paper, the velocity components are always located at the same position. The only staggering concerns the pressure field that can be located at the velocity mesh nodes (collocated mesh, see Fig. 1, left) or shifted by a half-mesh in each spatial direction (staggered mesh, see Fig. 1, right). Such a partial staggering, initially proposed by [10] and further developed by [1,11] using second-order schemes, is easy to implement because it does not require extensive use of mid-point interpolations, these procedures being computationally expensive when combined with high-order schemes. Another point is that any numerical code based on collocated meshes can be easily modified into this partially staggered approach. Indeed, only the pressure treatment needs to be modified.

4.1. Mid-point operators

To evaluate values $f'_{i+1/2}$ of the first derivative at the staggered nodes by a half-mesh $\Delta x/2$, a sixth-order finite difference scheme can be expressed as

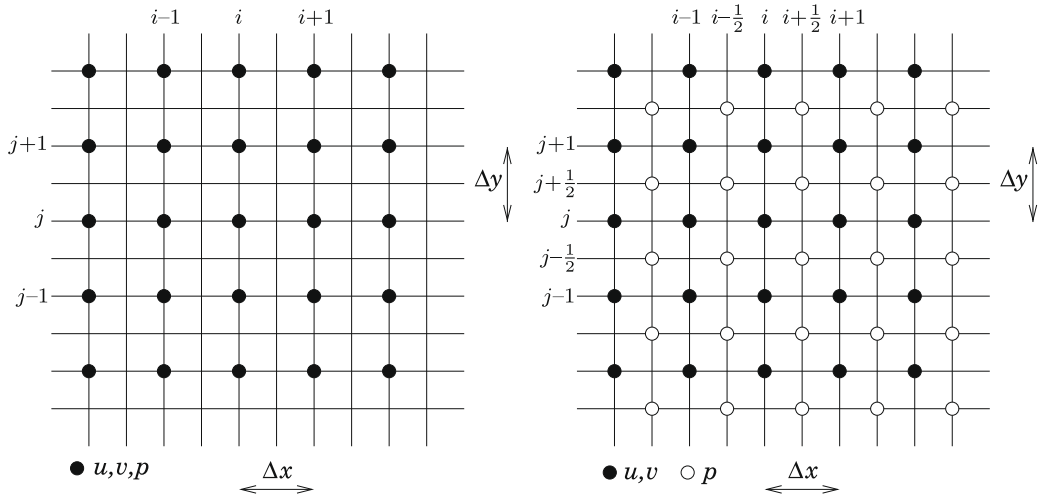

Fig. 1. Arrangement of variables in two dimensions for collocated (left) and partially staggered (right) meshes.

$$\alpha f'_{i-1/2} + f'_{i+1/2} + \alpha f'_{i+3/2} = a\frac{f_{i+1} - f_i}{\Delta x} + b\frac{f_{i+2} - f_{i-1}}{3\Delta x} \tag{20}$$

with $\alpha = 9/62, a = 63/62$ and $b = 17/62$. The spectral behaviour of this scheme is better than its collocated counterpart (13), but to evaluate its overall behaviour, it is necessary to consider its combination with a mid-point interpolation procedure given by

$$\alpha f^I_{i-1/2} + f^I_{i+1/2} + \alpha f^I_{i+3/2} = a\frac{f_{i+1} + f_i}{2} + b\frac{f_{i+2} + f_{i-1}}{2} \tag{21}$$

that provides an approximation of values of $f_{i+1/2}$ [30], this estimation being sixth-order accurate when $\alpha = 3/10, a = 3/4$ and $b = 1/20$.

4.2. Relevant Fourier transforms for periodic or symmetric boundary conditions

Assuming that f is periodic over the domain $[0, L_x]$, namely

$$f(x + L_x) = f(x) \tag{22}$$

the discrete Fourier transform may be defined as

$$\hat{f}_l = \frac{1}{n_x}\sum_{i=1}^{n_x} f_i e^{-\imath k_x x_i} \quad \text{for} \quad -n_x/2 \leqslant l \leqslant n_x/2 - 1 \tag{23}$$

with its inverse expression

$$f_i = \sum_{l=-n_x/2}^{n_x/2-1} \hat{f}_l e^{\imath k_x x_i} \tag{24}$$

where $\imath = \sqrt{-1}$ and where $k_x = 2\pi l/L_x$ is the wave number. Equivalent shifted transforms (based on mid-point locations) are simply obtained by the operation $i \to i + 1/2$ and can be computed using conventional FFT routines slightly pre- and post-processed.

Assuming now that f is symmetric in $x = 0$ and $x = L_x$, namely

$$f(x) = f(-x) \text{ and } f(x + L_x) = f(L_x - x) \tag{25}$$

for $x \in [0, L_x]$, the discrete cosine Fourier transform may be defined as

$$\hat{f}_l = \frac{2}{n_x - 1}\left[\frac{1}{2}f_1 + \sum_{i=2}^{n_x-1} f_i \cos(k_x x_i) + \frac{1}{2}(-1)^l f_{n_x}\right] \text{ for } 0 \leqslant l \leqslant n_x - 1 \tag{26}$$

with its inverse expression

$$f_i = \frac{\hat{f}_0}{2} + \sum_{l=1}^{n_x-2} \hat{f}_l \cos(k_x x_i) + \frac{1}{2}(-1)^{i+1}\hat{f}_{n_x-1} \tag{27}$$

where $k_x = \pi l/L_x$ is the wave number. The equivalent shifted transforms differ slightly from the previous expressions with

$$\hat{f}_l = \frac{2}{n_x - 1}\sum_{i=1}^{n_x-1} f_{i+1/2} \cos(k_x x_{i+1/2}) \text{ for } 0 \leqslant l \leqslant n_x - 2 \tag{28}$$

and its inverse expression

$$f_{i+1/2} = \frac{\hat{f}_0}{2} + \sum_{l=1}^{n_x-2} \hat{f}_l \cos(k_x x_{i+1/2}) \tag{29}$$

In practice, these discrete cosine Fourier transforms in their collocated (26) and (27) or shifted (28) and (29) versions can be easily computed using standard FFT routines through a relevant pre- and post-processing [41,5].

4.3. Spectral equivalence

It is easy to show (see for instance [28]) that the Fourier coefficients $\hat{f}'_l$ associated with the approximation (13) are linked to the Fourier coefficients $\hat{f}_l$ given by (23) by the simple spectral relation

$$\hat{f}'_l = \imath k'_x \hat{f}_l \tag{30}$$

where k'_x is the modified wave number related to the actual wave number k_x by

$$k'_x\Delta x = \frac{a\sin(k_x\Delta x) + (b/2)\sin(2k_x\Delta x)}{1 + 2\alpha\cos(k_x\Delta x)}. \tag{31}$$

The same modified wave number can be introduced for symmetric and antisymmetric conditions in order to establish the relation

$$\hat{f}'_I = k'_x\hat{f}_I \tag{32}$$

when they are given by cosine expansions (26).

Naturally, the concept of the modified wave number still holds in the staggered formulation, the expression of k'_x associated with the scheme (20) becoming

$$k'_x\Delta x = \frac{2a\sin(k_x\Delta x/2) + (2b/3)\sin(3k_x\Delta x/2)}{1 + 2\alpha\cos(k_x\Delta x)}. \tag{33}$$

In the same way, the relation between coefficients $\hat{f}^I_I$ associated with the mid-point interpolation scheme (21) and initial Fourier coefficients $\hat{f}_I$ can be easily established with

$$\hat{f}^I_I = T_x(k_x\Delta x)\hat{f}_I \tag{34}$$

where $T_x(k_x\Delta x)$ is the transfer function related to the wave number k_x by

$$T_x(k_x\Delta x) = \frac{2a\cos(k_x\Delta x/2) + (2b/3)\cos(3k_x\Delta x/2)}{1 + 2\alpha\cos(k_x\Delta x)} \tag{35}$$

Basically, the well known principle of equivalence between multiplication in Fourier space and derivation/interpolation in physical space is just recalled here, at the discrete level and in the framework of the approximation associated with centred finite difference schemes (13) and (20) or (21). Let us emphasize that this equivalence is exact. Hence, for example, the computation of a derivative in physical space using (20) with relevant boundary conditions (15) and (16) or (17) must lead strictly to the same result obtained with the use of (33) in spectral space.

4.4. Extension toward homogeneous Neumann conditions

For non-periodic and non-symmetric boundary conditions, due to the well known Gibbs phenomenon (see [5] for instance), the use of Fourier transforms (23), (24), (26) and (27) must be avoided to evaluate spatial derivatives using the exact wave number. The situation is almost the same using the modified wave numbers (31) or (33) that would lead to wiggles near the border of the computational domain. Naturally, according to the equivalence discussed in the previous section, the direct evaluation of the first derivative using (13) or (20) combined with any set of boundary conditions (15) and (16) or (17) would provide exactly the same border wiggles. In other words, no accurate and smooth differentiation can be expected using periodicity or symmetrical assumptions when the function itself is neither periodic nor symmetric.

However, noting that the symmetrical condition (25) leads to

$$f'(0) = 0 \text{ and } f'(L_x) = 0 \tag{36}$$

it is tempting to use this assumption to consider functions submitted to these two boundary conditions while not ensuring the full requirement (25). To illustrate how this intermediate treatment works and what type of error it generates, let us consider the seventh-order polynomial function

$$f(x) = \frac{x^7}{7} - \frac{x^6}{2} + \frac{17x^5}{25} - \frac{9x^4}{20} + \frac{274x^3}{1875} - \frac{12x^2}{625} \tag{37}$$

over the domain $L_x = 1$. This function obviously does not respect the symmetrical condition (25) while ensuring the boundary conditions (36). The computation of its first derivative using (20) combined with ghost boundary conditions (16) can be compared with its exact value

$$f'(x) = (x-1)\left(x-\frac{4}{5}\right)\left(x-\frac{3}{5}\right)\left(x-\frac{2}{5}\right)\left(x-\frac{1}{5}\right)x \tag{38}$$

The resulting convergence is presented in Fig. 2 where the error corresponds to the maximum deviation between the exact value f' and its approximation using (20) and (16). This error is found to be maximum at borders $x = 0$ and L_x, as it can be expected due to ghost boundary conditions (16). In terms of convergence, despite the use of a sixth-order scheme (20), it is easy to demonstrate (see Appendix A) that the additional approximation associated with the border treatment (16) introduces a second-order error in the evaluation of the values near the borders, so that the maximal error is of second-order. This behaviour can be checked in Fig. 2 (left). As a reference, the behaviour of the error associated with the use of the standard second-order scheme

$$f'_{i+1/2} = \frac{f_{i+1} - f_i}{\Delta x} \tag{39}$$

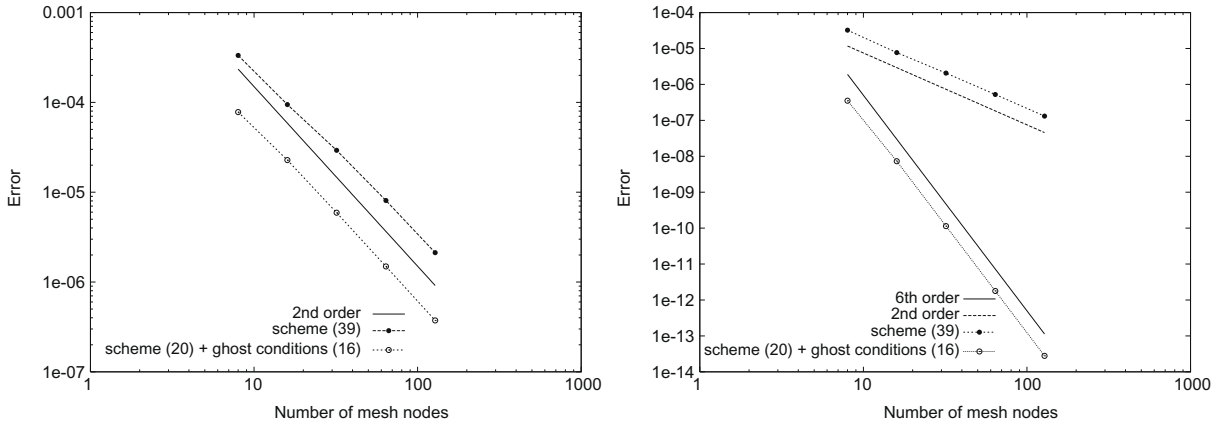

Fig. 2. Decrease with n_x of the error associated with the differentiation of function f (37) using the sixth-order scheme (20) combined with ghost boundary conditions (16) or using the standard second-order scheme (39). Left: maximal error at the border $x = 0$. Right: error at the centre $x = L_x/2$.

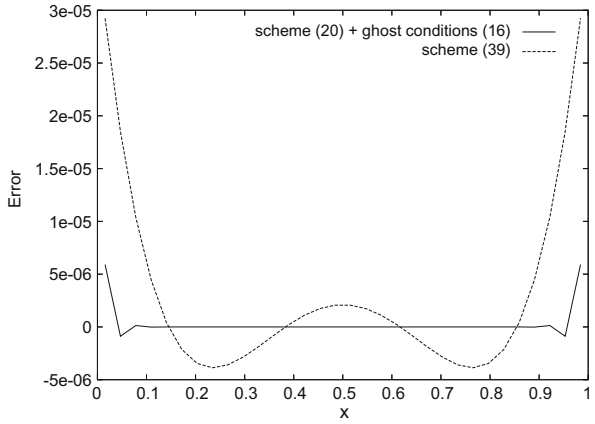

Fig. 3. Local error associated with the differentiation of f (37) using the sixth-order scheme (20) combined with ghost boundary conditions (16) or using the standard second-order scheme (39). Tests performed with $n_x = 32$ mesh nodes.

is also presented in Fig. 2. The advantage of this scheme is its very compact stencil that avoids any assumption near the border through the use of ghost boundary conditions like those given by (16). Despite this attractive property (already widely exploited in the literature), it is interesting to observe in Fig. 2 that this second-order scheme is clearly less accurate than its sixth-order counterpart (20) combined with ghost boundary conditions (16). Outside from the border neighbourhood, a sixth-order order convergence can even be recovered with this latter combination (20) and (16) as shown by Fig. 2 (right). To have a better idea of the errors introduced by these two schemes, their local error (as a function of x) is presented in Fig. 3 for $n_x = 32$ mesh nodes. For the scheme (20) combined with ghost boundary conditions (16), the occurrence of maximal errors near the borders is clearly exhibited while being significantly reduced compared with the error associated with the standard scheme (39). Note that, for the present seventh order polynomial function, the error introduced by the second-order scheme (39) is scaled on the third derivative f''' leading to the specific curve presented in Fig. 3 where no singular behaviour can be observed near the borders. For the sixth-order scheme (20) combined with ghost boundary conditions (16), the error is also found to be dependent on x while exhibiting a slight Gibbs effect, a behaviour that can be accepted regarding the low error level.

To summarize, this very simple test based on the function f (37) shows that the sixth-order scheme (20) combined with ghost boundary conditions (16) can provide an accurate estimation of the first derivative.[2] Despite the formal second-order of this combination, the accuracy obtained is significantly improved compared with the standard second-order scheme (39). However, the condition required to obtain such a favourable behaviour is that the function submitted to the differentiation

[2] Note that this point has been illustrated using only staggered schemes, but similar conclusions can be drawn for collocated schemes, except for the standard second-order scheme $f'_i = (f_{i+1} - f_{i-1})/2\Delta x$ that cannot avoid completely ghost boundary conditions (16) so that border errors are generated contrary to the standard second-order staggered scheme (39).

must verify the boundary conditions (36).[3] The drawback of ghost boundary conditions (16) is that they limit any scheme to the second-order accuracy, but their important advantage is that they allow the discretization to remain homogeneous in the whole computational domain. Then, it is possible to define exactly and easily the differentiation in spectral space. Hence, as for periodic or free-slip boundary conditions, the use of compact sixth-order scheme (13) or (20) combined with the ghost conditions (16) in physical space remains strictly equivalent to the multiplication by the modified wave number (31) or (33) in spectral space defined using the cosine Fourier transforms (26) and (27) or (28) and (29), respectively.

For the incompressible Navier–Stokes equations, the restriction (36) is found to be well adapted to a numerical treatment where homogeneous Neumann conditions are applied on the pressure field in the present fractional step method (see Section 2.3). In this work, this property is exploited for the full treatment of the pressure that can be defined in the physical or spectral space depending on the stage of the solving. The overall pressure treatment procedure in the present numerical code is presented in the next section.

4.5. Numerical procedure of the pressure treatment

To simplify the writing, a generic discrete 3D Fourier transform can be defined as

$$\hat{p}_{lmn} = \frac{1}{n_x n_y n_z} \sum_i \sum_j \sum_k p_{ijk} W_x(k_x x_i) W_y(k_y y_j) W_z(k_z z_k) \tag{40}$$

with its inverse expression

$$p_{ijk} = \sum_l \sum_m \sum_n \hat{p}_{lmn} W_x(-k_x x_i) W_y(-k_y y_j) W_z(-k_z z_k) \tag{41}$$

where the sums, the base functions (W_x, W_y, W_z) and the wave numbers $(k_x, k_y,\ k_z)$ can correspond to standard FFT (for periodic boundary conditions) or cosine FFT (for free-slip or **u**-Dirichlet/*p*-Neumann boundary conditions) in their collocated (23), (24), (26) and (27) or shifted versions (28) and (29). In practice, the 3D direct (40) and inverse (41) transforms can be easily performed with any efficient FFT routines available in scientific Fortran or C libraries for instance. Consequently, this stage in the implementation of the present method to solve the Poisson equation does not require significant numerical developments.

The first stage in the solving of the Poisson (11) consists in the computation of its right hand side. In order to preserve compatibility between the discrete differentiation operators in physical and spectral spaces, the calculation of $\mathbf{\nabla}.[(1-\varepsilon)\mathbf{u}^{**}]$ in (11) must be done with collocated (13) or staggered (20) schemes preserving the set of boundary conditions considered. Note that for the partially staggered case, the divergence is computed on pressure nodes represented by empty circles in Fig. 1.

Having performed the relevant Fourier transform (40) to $D = \mathbf{\nabla} \cdot [(1-\varepsilon)\mathbf{u}^{**}]$, the solving of the Poisson equation consists in a single division of each Fourier mode $\widehat{D}_{lmn}$ by a factor F_{lmn} with

$$\hat{\tilde{p}}^{k+1}_{lmn} = \frac{\widehat{D}_{lmn}}{F_{lmn}} \tag{42}$$

The expression of this factor depends on the mesh configuration. For a fully collocated configuration, it takes the following form

$$F_{lmn} = -\left(k_x'^2 + k_y'^2 + k_z'^2\right) c_k \Delta t \tag{43}$$

where (k_x', k_y', k_z') are modified wave numbers defined in (31). In the partially staggered approach, the factor F_{lmn} must take the mid-point interpolation into account through the use of the transfer function (35) with the following form

$$F_{lmn} = -[(k_x' T_y T_z)^2 + (k_y' T_x T_z)^2 + (k_z' T_x T_y)^2] c_k \Delta t \tag{44}$$

For the collocated case, the expression of (43) leads to a significant number of Fourier modes (k_x, k_y, k_z) where F_{lmn} is zero in the expression (42). These specific modes correspond to the well known singular modes of pressure for which $k_x'^2 + k_y'^2 + k_z'^2 = 0$. For a given direction, *x* for instance, the modified wave number (31) of a collocated scheme can be zero for $k_x = 0$ but also for $k_x \Delta x = \pi$, the so-called cutoff wave number. Consequently, in addition to the expected singular mode $(k_x, k_y, k_z) = (0,0,0)$ that corresponds to an undefined constant of the pressure for an incompressible flow, seven singular modes can be identified in 3D[4] due to the simultaneous cancellation of the three modified wave number $(k_x', k_y', k_z') = (0,0,0)$. Note that in the context of purely spectral methods, it can be shown that these modes can only be avoided through the use of a pressure mesh based on $N-2$ nodes when the velocity mesh is based on N nodes in each spatial

[3] For more general functions, it can be shown that the standard second-order (39) is clearly more accurate by avoiding the border assumptions (16) and their resulting strong Gibbs phenomena.

[4] For simplicity, only the modes with positive wave numbers are counted, the other ones being deduced by symmetry with respect to the plane $k_x = 0, k_y = 0$ and $k_z = 0$ while taking Hermicity into account.

discretization [8]. When the Poisson equation is solved in physical space, these singular modes can affect significantly the efficiency of the solver while creating non-physical oscillations on the pressure field. Here, because the Poisson equation is expressed in spectral space, singular modes can be easily eliminated by simply skipping all the modes where $k_x'^2 + k_y'^2 + k_z'^2 = 0$. However, as it will be observed in Section 7.3, in the spectral region near-singular conditions $k_x'^2 + k_y'^2 + k_z'^2 \approx 0$, the division by F_{lmn} tends to amplify the corresponding modes so that pressure oscillations can appear.

For the partially staggered case, the situation is less critical because the modified wave number (33) is zero only for $k_x = 0$ while being maximal for $k_x\Delta x = \pi$ (as the actual wave number). However, four additional singular modes can be identified due to the cancellation of the transfer functions T_x, T_y and T_z for $k_x\Delta x = \pi, k_y\Delta y = \pi$ and $k_z\Delta z = \pi$, respectively. These singular modes correspond to spurious oscillations with a zigzag pattern in oblique directions with respect to the Cartesian mesh. In Section 7.3, it will be shown that in the neighbourhood of singular conditions where $\left(k_x' T_y T_z\right)^2 + \left(k_y' T_x T_z\right)^2 + \left(k_z' T_x T_y\right)^2 \approx 0$, the resulting amplification of the corresponding pressure modes does not create significant wiggles contrary to the fully collocated case.

In order to perform the correction (5), the most efficient method is to compute the pressure in physical space using the inverse Fourier transform (41). Then, the pressure gradients $\nabla\tilde{p}^{k+1}$ can be calculated using a collocated (13) or a staggered (20) scheme depending on the type of mesh used. After this correction, depending on the use or not of an IBM, the divergence free conditions (8) or (12) can be ensured up to the machine accuracy, the strict equivalence between all the discrete operators (in physical as well as in spectral space) being preserved throughout the full pressure treatment.

5. Extension toward a stretched mesh in one direction

At this stage, the overall procedure for the discretization of the pressure is valid only for a regular mesh in the three spatial directions. This constraint can be overcome through the use of a mapping easily usable in physical space, the basic principle being based on an equally spaced mesh in the mapped space. However, expressed in spectral space, a general mapping can introduce a serious difficulty in terms of algorithm and computational cost. Indeed, the main problem is linked to the use of a product in physical space that corresponds to a convolution product in spectral space. For this reason, the spectral equivalence discussed in Section 4.3 needs to be reconsidered. For instance, the simple solving of the Poisson equation based only on a single division in spectral space is lost. However, following the stretching technique introduced by [4] and extended by [3], a slight modification of the Poisson solver is proposed, based on a specific function mapping and expressed using only few Fourier modes. Thus, the spectral and non-iterative nature of the pressure treatment can be preserved without significant loss of accuracy.

To simply explain this method, the general analysis of the mesh technique is restricted to a one-dimensional problem. Let y be the physical coordinate[5] and s the computational coordinate introduced by

$$y = h(s),\ 0 \leqslant s \leqslant 1,\ 0 \leqslant y \leqslant L_y \tag{45}$$

where $h(s)$ is the mapping from the equally spaced coordinate s to the stretched spaced coordinate y. The derivatives with respect to y can be estimated using the chain rule. The first derivative can be expressed as

$$\frac{\partial f}{\partial y} = \frac{\partial f}{\partial s}\frac{ds}{dy} = \frac{1}{h'}\frac{\partial f}{\partial s} \tag{46}$$

and the second derivative as

$$\frac{\partial^2 f}{\partial y^2} = \frac{\partial^2 f}{\partial s^2}\left(\frac{ds}{dy}\right)^2 + \frac{\partial f}{\partial s}\frac{d^2 s}{dy^2} = \frac{1}{h'^2}\frac{\partial^2 f}{\partial s^2} - \frac{h''}{h'^3}\frac{\partial f}{\partial s} \tag{47}$$

Expressed in physical space, these rules are very easy to implement for schemes like (13), (14) and (20) where the finite differences are simply performed on the regular coordinate s (instead of x). The main problem comes from the treatment of the Poisson equation proposed in this study that requires to define similar operations in spectral space. Since ensuring the divergence free condition, the Laplacian in Poisson equation (11) has to be discretized using an iterated first derivative operator, based on (45). In practice, the multiplication $\frac{1}{h'}\frac{\partial f}{\partial s}$ has to be expressed in spectral space. Assuming than f can be represented by one of the discrete Fourier transforms ((23) and (24), (26) and (27), (28) and (29)) where the regular coordinate is s instead of x, the point is to define appropriately the Fourier representation of the metric $1/h'$. For clarity, let us consider the periodic case[6] where f can be expressed as

$$f_i = \sum_{m=-n_y/2}^{n_y/2-1} \hat{f}_m e^{ik_m s_i} \tag{48}$$

[5] Only stretched meshes in y will be considered in this paper as well as in flow configurations considered in the validation section, but present methodology of stretching remains valid for any single direction whatever the type of boundary conditions.

[6] The present description of the method can be adapted straightforwardly to symmetric or/and shifted Fourier transforms.

the metric $1/h'$ as

$$\frac{1}{h'_i} = \sum_{m=-n_y/2}^{n_y/2-1} \hat{a}_m e^{\imath k_m s_i} \tag{49}$$

and the y-derivative of f by

$$\left.\frac{\partial f}{\partial y}\right|_i = \sum_{m=-n_y/2}^{n_y/2-1} \left.\frac{\widehat{\partial f}}{\partial y}\right|_m e^{\imath k_m s_i} \tag{50}$$

where $k_m = 2\pi m$ is the wave number associated with the uniform distribution of n_y points $s_i = (i-1)/n_y$ on the regular coordinate s.

Expressed in spectral space, the product

$$\left.\frac{\partial f}{\partial y}\right|_i = \frac{1}{h'_i}\left.\frac{\partial f}{\partial s}\right|_i \tag{51}$$

leads to the convolution product

$$\left.\frac{\widehat{\partial f}}{\partial y}\right|_m = \sum_{p=-n_y/2}^{n_y/2-1} \hat{a}_p \left.\frac{\widehat{\partial f}}{\partial s}\right|_{m-p} \quad \text{for } -n_y/2 \leqslant m \leqslant n_y/2-1 \tag{52}$$

requiring to perform n_y^2 multiplications. Then, in terms of computational cost, the use of a such mapping appears to be very penalizing. Moreover, the convolution product (52) does not mimic exactly the multiplication in physical space (51) due to the truncation of the modes outside from the range $-n_y/2 < m < n_y/2-1$. This truncation eliminates the aliasing errors, but because they are present when the operation is performed in physical space as (51), the physical/spectral equivalence is lost. For this specific point, it would be possible to mimic exactly the aliasing error in spectral space (through a relevant "contamination" of corresponding modes) but with significant damages on the result depending on the decay of the Fourier coefficients $\hat{f}_m$ and $\hat{a}_m$. An illustration of aliasing introduction in spectral space (to preserve compatibility between derivation in physical and spectral space) is presented in Appendix B.

Following the technique introduced by [4], a metric, expressed with only three Fourier modes in spectral space, is proposed here with

$$\frac{1}{h'} = \frac{1}{L_y}\left\{\frac{\alpha}{\pi} + \frac{1}{\pi\beta}\sin^2(\pi(\gamma s+\delta))\right\} = \frac{1}{L_y}\left\{\frac{\alpha}{\pi} + \frac{1}{2\pi\beta}\left[1 - \frac{e^{\imath 2\pi(\gamma s+\delta)} + e^{-\imath 2\pi(\gamma s+\delta)}}{2}\right]\right\} \tag{53}$$

so that the mapping (45) can be written as

$$h = \frac{L_y\sqrt{\beta}}{\gamma\sqrt{\alpha}\sqrt{\alpha\beta+1}}\left\{\tan^{-1}\left[\frac{\sqrt{\alpha\beta+1}\tan(\pi(\gamma s+\delta))}{\sqrt{\alpha}\sqrt{\beta}}\right] + \pi\left[H\left(s-\frac{1-2\delta}{2\gamma}\right) + H\left(s-\frac{3-2\delta}{2\gamma}\right)\right] - \tan^{-1}\left[\frac{\sqrt{\alpha\beta+1}\tan(\pi\delta)}{\sqrt{\alpha}\sqrt{\beta}}\right]\right\} \tag{54}$$

where H is the Heaviside step function. The present mapping is a generalization of those proposed by [4] and [3]. It preserves the accuracy while avoiding the expensive computation of a full convolution and ensuring the strict physical/spectral equivalence. When $\alpha = 0, \gamma = 1$ and $\delta = 0$, Eq. (54) degenerates to Cain et al.'s [4] mapping function that leads to a fine mesh in the centre of an infinite domain $L_y = \infty$. For $\alpha \neq 0$, $\gamma = 1$ and $\delta = 0$, present function corresponds to the one of [3] that allows a central mesh refinement in a finite domain. Using $\gamma = 1$ and $\delta = 1/2$, the concentration of mesh nodes is performed near the boundaries of the domain at $y = 0$ and L_y. Finally, the choice $\gamma = 1/2$ and $\delta = 1/2$ leads to mesh refinement only at one boundary $y = 0$. Note that this latter case is not compatible with periodic boundary conditions because $1/h'$ is not periodic over L_y. Consequently, when $\gamma = 1/2$, only cosine Fourier transforms (26)–(29) can be used (for present code, Neumann conditions for the pressure and free-slip or Dirichlet conditions for the velocity), whereas for the other cases, both periodic and Neumann conditions can be considered. Note that α needs to be adjusted depending on the set of parameters (β, γ, δ) used to ensure $h(1) = L_y$, with for instance $\alpha = \frac{-1+\sqrt{1+4\beta^2\pi^2}}{2\beta}$ when $\gamma = 1$.

By identification between the expression (53) and its Fourier expansion (49), it is easy to deduce that only three coefficients of the metric are non-zero with

$$\hat{a}_0 = \frac{1}{L_y}\left(\frac{\alpha}{\pi} + \frac{1}{2\pi\beta}\right), \quad \hat{a}_1 = \hat{a}_{-1} = -\frac{1}{L_y}\left(\frac{\cos 2\pi\delta}{4\pi\beta}\right) \tag{55}$$

for $\gamma = 1$ and $\delta = 0$ or $1/2$ (the case $\gamma = 1/2$ needs to consider cosine expansion as already stated). The main advantage of this very compact expression in spectral space is that the convolution product of the metric by the first derivative with respect to the regular coordinate s requires only $3n_y$ multiplications with

$$\left.\frac{\widehat{\partial f}}{\partial y}\right|_m = \sum_{p=-1}^{1} \hat{a}_p \left.\frac{\widehat{\partial f}}{\partial s}\right|_{m-p} \tag{56}$$

Using the modified wave numbers k'_m associated with k_m through a relation like (31) or (33) based on the computational coordinate s instead of x, the same relation including the differentiation in s can be written

$$\left.\frac{\widehat{\partial f}}{\partial y}\right|_m = \imath \sum_{p=-1}^{1} \hat{a}_p \; k'_{m-p} \hat{f}_{m-p} \tag{57}$$

so the resulting full discretization leads formally to

$$\hat{\mathbf{f}}' = \mathbf{A}\hat{\mathbf{f}} \tag{58}$$

where $\hat{\mathbf{f}}$ and $\hat{\mathbf{f}}'$ are n_y vectors of components $\hat{f}_m$ and $\left.\frac{\widehat{\partial f}}{\partial y}\right|_m$, respectively and $\mathbf{A}$ a $n_y \times n_y$ tridiagonal matrix of components

$$A_{m,m-1} = \imath \hat{a}_1 k'_{m-1}, \qquad A_{m,m} = \imath \hat{a}_0 k'_m, \qquad A_{m,m+1} = \imath \hat{a}_{-1} k'_{m+1} \tag{59}$$

This operation, applied once again to estimate the second derivative of f, leads to

$$\hat{\mathbf{f}}'' = \mathbf{B}\hat{\mathbf{f}} \tag{60}$$

where $\hat{\mathbf{f}}''$ is a n_y vector of components $\left.\frac{\widehat{\partial^2 f}}{\partial y \partial y}\right|_m$ and $\mathbf{B} = \mathbf{A}\mathbf{A}$ a $n_y \times n_y$ pentadiagonal matrix of components

$$\begin{aligned}
b_{m,m-2} &= -\hat{a}_1^2 k'_{m-1} k'_{m-2} \\
b_{m,m-1} &= -\hat{a}_0 \hat{a}_1 k'_{m-1} (k'_m + k'_{m-1}) \\
b_{m,m} &= -\hat{a}_0^2 {k'_m}^2 - \hat{a}_1 \hat{a}_{-1} k'_m (k'_{m+1} + k'_{m-1}) \\
b_{m,m+1} &= -\hat{a}_0 \hat{a}_{-1} k'_{m+1} (k'_m + k'_{m+1}) \\
b_{m,m+2} &= -\hat{a}_{-1}^2 k'_{m+1} k'_{m+2}
\end{aligned} \tag{61}$$

For the solving of the Poisson equation (using 3D Fourier transforms (40) and (41) where y needs to be substituted by s for the present y-stretched approach), the counterpart of the integration scheme (42) becomes

$$\hat{\mathbf{p}}_{ln}^{k+1} = \mathbf{B}^{-1} \hat{\mathbf{D}}_{ln} \tag{62}$$

where $\hat{\mathbf{p}}_{ln}^{k+1}$ and $\hat{\mathbf{D}}_{ln}$ are n_y vectors of components $\hat{p}_{lmn}^{k+1}$ and $\hat{D}_{lmn}$, respectively and $\mathbf{B}$ a $n_y \times n_y$ pentadiagonal matrix of components

$$\begin{aligned}
b_{m,m-2} &= -\hat{a}_1^2 k'_{m-1} k'_{m-2} \\
b_{m,m-1} &= -\hat{a}_0 \hat{a}_1 k'_{m-1} (k'_m + k'_{m-1}) \\
b_{m,m} &= -(k'_x + k'_z)^2 - \hat{a}_0^2 {k'_m}^2 - \hat{a}_1 \hat{a}_{-1} k'_m (k'_{m+1} + k'_{m-1}) \\
b_{m,m+1} &= -\hat{a}_0 \hat{a}_{-1} k'_{m+1} (k'_m + k'_{m+1}) \\
b_{m,m+2} &= -\hat{a}_{-1}^2 k'_{m+1} k'_{m+2}
\end{aligned} \tag{63}$$

for the fully collocated case. For the partially staggered case, the equivalent matrix takes the form

$$\begin{aligned}
b_{m,m-2} &= -\hat{a}_1^2 T_x^2 T_z^2 k'_{m-1} k'_{m-2} \\
b_{m,m-1} &= -\hat{a}_0 \hat{a}_1 T_x^2 T_z^2 k'_{m-1} (k'_m + k'_{m-1}) \\
b_{m,m} &= -(k'_x T_y T_z)^2 - (k'_z T_y T_z)^2 - \hat{a}_0^2 T_x^2 T_z^2 {k'_m}^2 - \hat{a}_1 \hat{a}_{-1} T_x^2 T_z^2 k'_m (k'_{m+1} + k'_{m-1}) \\
b_{m,m+1} &= -\hat{a}_0 \hat{a}_{-1} T_x^2 T_z^2 k'_{m+1} (k'_m + k'_{m+1}) \\
b_{m,m+2} &= -\hat{a}_{-1}^2 T_x^2 T_z^2 k'_{m+1} k'_{m+2}
\end{aligned} \tag{64}$$

Naturally, these two matrices are diagonal for a regular y-coordinate (with $a_1 = a_{-1} = 0$) so that the simplified expressions (43) and (44) can be recovered. In the other cases, the computation of pressure nodes $\hat{\mathbf{p}}_{ln}^{k+1}$ requires to invert $n_x \times n_z$ linear systems based on $n_y \times n_y$ pentadiagonal matrices. The corresponding computational cost is, as for the regular case, proportional to $n_x \times n_y \times n_z$ so that the solver Poisson can be direct without any iterative process.

Finally, note that some of the coefficients of $\mathbf{A}$ and $\mathbf{B}$ given by (59), (61), (63) and (64) need to be slightly modified near the borders of the matrices for two reasons. First, any coefficient with subscript outside from the range $-n_y/2 < m < n_y - 1$ has to be set to zero due to the truncation of present Fourier series. When this truncation is performed, the resulting matrices $\mathbf{A}$ and $\mathbf{B}$ are free from any aliasing error. Defined in spectral space, the corresponding operations are aliased. In consequence, based on spectral de-aliased operators, the physical/spectral equivalence is not exactly satisfied. As already discussed in this section, this equivalence can be re-established through the relevant contamination of the coefficients of $\mathbf{A}$ and $\mathbf{B}$ in order to mimic exactly the aliasing produced naturally in physical space by the product (51). The resulting modifications concern only the first and last lines of the matrices $\mathbf{A}$ and $\mathbf{B}$ due to the use of present metric (53) that limits drastically the aliasing process. An example of aliasing implementation through the modification of the matrix $\mathbf{A}$ can be found in Appendix B.

6. Concluding remarks and computational cost

As the strict equivalence between the spatial discretization in physical and spectral spaces is always maintained, the present technique allows the verification of the incompressibility condition up to the machine accuracy. In practice, this property seems to be important in terms of stability. If the matching between the physical and spectral discretization is not strictly satisfied, the resulting residual divergence is found to increase during the calculation whatever its initial level. For instance, the use of a non-aliased operator in spectral space for the stretched case leads to a very low but remaining divergence[7] that tends to accumulate during the simulation and finally leads to a strong numerical instability at small scales.

In terms of computational cost, the direct inversion of the Poisson equation needs to perform one direct and one inverse 3D FFT. For a fully regular mesh in the three spatial directions, these two FFT are the main part of the computational cost of the Poisson stage which is about 10% of the total treatment of the Navier–Stokes equations. For meshes with one stretching direction, the cost of the incompressibility treatment increases but remains about 15% of the overall cost for a given simulation. Such relative costs can be considered to be satisfactory regarding the use of high-order schemes.

Naturally, the Fourier formalism suffers from drawbacks linked to implicit assumptions about the boundary conditions. For periodic or free-slip boundary conditions, the use of Fourier transforms (23) or (26) for the pressure is perfectly adapted and does not introduce additional errors. On the contrary, for no-slip or open (i.e. Dirichlet for velocity) boundary conditions, the development of the pressure in cosine expansion like (26) needs to assume that homogeneous Neumann conditions are verified. This implicit assumption introduces an error that is only second-order accurate in space. In the following results, the interest of high-order schemes in such a limited framework will be discussed by analysing validation results in various computational configurations.

7. Validation and results

In this section, six computational configurations are considered, with different sets of boundary conditions and different flow motions. In the Section 7.1, periodic or free-slip conditions are used for preliminary verifications in 2D problems. Then, the ability of the present code to treat satisfactory no-slip boundary conditions (despite the use of a modified spectral approach for the pressure) is evaluated in Section 7.2 for a stationary flow. Note that in the same section, the error introduced by the IBM is exhibited. Furthermore, a combination of present numerical methods with IBM is also used in Section 7.3 where the simulation of the simple steady flow over a circular cylinder emphasizes the interest of a partially staggered approach. Finally, DNS of a transitional channel flow, a vortex-dipole rebound near a wall and a turbulent channel flow are presented in Sections 7.4, 7.5 and 7.6, respectively, in order to compare the efficiency of sixth-order compact schemes with respect to conventional second-order schemes.

To better understand the following comparisons and conclusions, it could be useful to clarify what terms like sixth-order schemes and collocated/staggered approaches will refer to. A numerical code based on sixth-order schemes means that (i) convective and viscous terms are discretized using (13) and (14) (ii) pressure term is discretized using the same scheme for a collocated approach or using mid-point operators (20) and (21) for a partially staggered approach. In addition, for Dirichlet conditions on the velocity, near-border formulations (18) and (19) are used only for the computation of convective and viscous terms.

In preliminary calculations, a partially staggered second-order version of the present code has been tested. For all the following validation tests, comparisons between the partially second-and sixth-order versions have clearly shown the improvement offered by the use of compact sixth-order schemes. However, this spectacular benefit has been found somewhat artificial because it was partly due to the poor accuracy of a second-order approach based on a partially staggered mesh. To our knowledge, the best strategy to use second-order finite schemes is based on a fully staggered arrangement of the mesh as initially proposed by [13] to eliminate all singular modes in the pressure field. The efficiency and accuracy of this approach, known as the MAC method, has been clearly shown by [32] for the DNS of turbulent flows. In consequence, to show the interest of the present partially staggered sixth-order approach, comparisons with a fully staggered second-order code will be presented for several validation tests presented in this section. Note that in practice, the numerical code available in the book of [32] has been used. A version adapted to the flow configurations considered for comparisons with present code can be found on the CD that comes with the book. It is important to note that this version has been directly used to produce second-order results without performing code development while checking the consistency of these present results with those reported in [32] for the turbulent channel flow.

7.1. Taylor-Green vortices

In a square domain $L_x = L_y = 1$ with periodic or free-slip conditions at the boundaries $x = 0, x = 1, y = 0$ and $y = 1$, starting from initial conditions

[7] Typically, the reduction of the divergence is more than ten order of magnitude.

$$u(x,y,0) = \sin(2\pi x)\cos(2\pi y)$$
$$v(x,y,0) = -\cos(2\pi x)\sin(2\pi y)$$

the analytic solution of the incompressible Navier–Stokes (1) and (2) equations without any forcing ($\mathbf{f} = \mathbf{0}$) is given by

$$u(x,y,t) = \sin(2\pi x)\cos(2\pi y)e^{-4\pi^2 \nu t}$$
$$v(x,y,t) = -\cos(2\pi x)\sin(2\pi y)e^{-4\pi^2 \nu t} \tag{65}$$

In order to confirm the perfect compatibility between the spectral Poisson solver presented here and the rest of the discretization defined in physical space, the exact solution (65) was compared with its approximation for this benchmark. The evolution of the flow is simulated from $t_0 = 0$ to $t_1 = 0.5$, using a very small time step $\Delta t = 0.0005$ in such a way that time discretization errors can be considered as negligible with respect to the spatial ones. The square computational domain is discretized on $n_x \times n_y$ mesh nodes with $n_x = n_y$. The Reynolds number defined by $Re = 1/\nu$ is equal to 1000.

The error is considered through the standard deviation of the computed velocity component u_c from its exact solution u_e defined as

$$\text{Error} = \sqrt{\frac{1}{n_x}\frac{1}{n_y}\sum_{i=1}^{n_x}\sum_{j=1}^{n_y}(u_c(x_i,y_j,t_1) - u_e(x_i,y_j,t_1))^2} \tag{66}$$

with $x_i = (i-1)/n_x$ and $y_j = (j-1)/n_y$.

For each case, four calculations were performed using 8, 16, 32 and 64 mesh nodes in each direction. The error decrease is presented in Fig. 4 for four cases combining regular/stretched and collocated/staggered meshes.

It can be clearly seen that for each case, the error follows a decrease corresponding to a sixth-order accuracy as expected when compact schemes (13) and (14) or (20) are used. For the present solution (65) based on sine and cosine functions, stretched meshes lead logically to a significant increase of the error, the regular mesh being optimal whereas stretching introduces local (in unrefined regions) and aliasing errors as already discussed in Section 5. However, the use of present metric allows the preservation of the sixth-order convergence as shown in Fig. 4. Finally, it is worth noting that no significant difference can be observed between collocated or partially staggered approaches, both mesh organisations leading virtually to the same accuracy.

7.2. Burggraf flow

The Burggraf flow corresponds to a steady recirculating flow submitted to a forcing in a square cavity $L_x = L_y = 1$. The velocity boundary conditions are no-slip conditions with

$$\mathbf{u}(x,0) = \mathbf{u}(0,y) = \mathbf{u}(1,y) = \mathbf{0} \tag{67}$$

while at the top boundary, a Dirichlet condition is applied

$$u(x,1) = 16(x^4 - 2x^3 + x^2) \tag{68}$$

This set of boundary conditions avoids the presence of discontinuities at the top corners of the cavity. If a forcing of the form

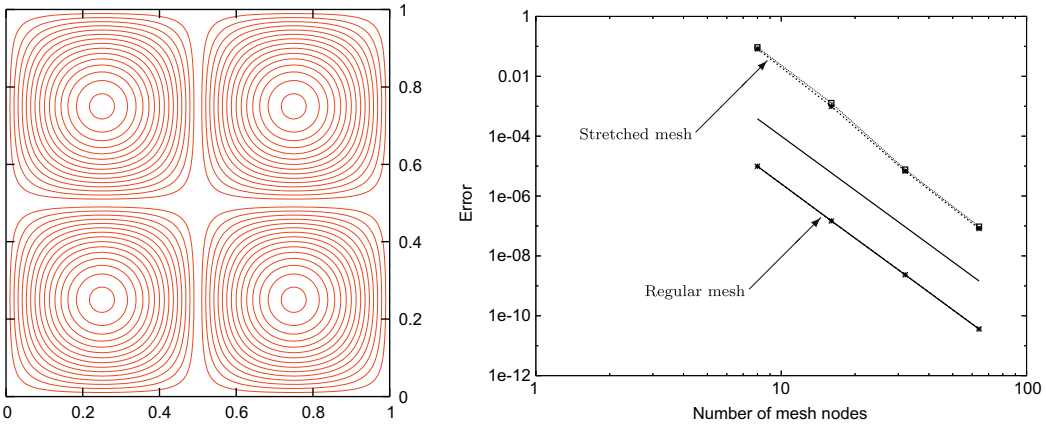

Fig. 4. Left: Taylor–Green vortices (periodic or free-slip boundary conditions) represented by streamlines. Right: Error decrease with the same number of mesh nodes $n_x = n_y$ in each direction. The solid line corresponds to a sixth-order decrease in n_x^{-6} while dashed/dotted lines designate the computed solutions based on a fully collocated or a partially staggered mesh, respectively (dashed and dotted lines are superimposed for the regular one).

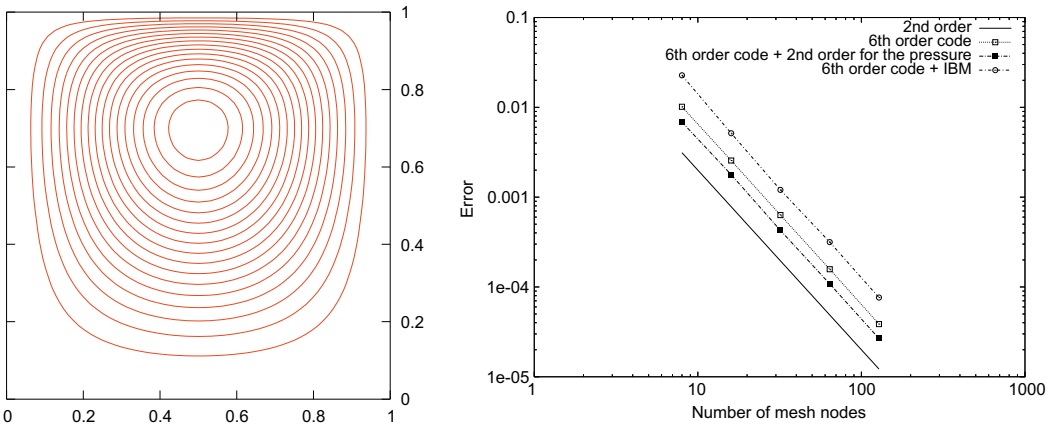

Fig. 5. Left: Burggraf flow (no-slip boundary conditions) represented by streamlines. Right: Error decrease with the number of mesh nodes $n_x = n_y$ in each direction. Results obtained using a regular and partially staggered mesh.

$$\mathbf{f} = -\left\{\frac{8}{Re}[24g + 2g''h'' + g^{(4)}h] - 64\left[\frac{g'^2}{2}(hh''' - h'h'') - hh'(g'g''' - g''^2)\right]\right\}\mathbf{e_x} \tag{69}$$

with

$$g(x) = \frac{x^5}{5} - \frac{x^4}{2} + \frac{x^3}{3}, \quad h(y) = y^4 - y^2 \tag{70}$$

is used in equations (1), the exact solution is (see for instance [37])

$$\begin{aligned} u(x,y) &= 8(x^4 - 2x^3 + x^2)(4y^3 - 2y) \\ v(x,y) &= -8(4x^3 - 6x^2 + 2x)(y^4 - y^2) \end{aligned} \tag{71}$$

The streamlines illustrating this cavity flow are plotted in Fig. 5. This benchmark problem allows us to evaluate the behaviour of our code when only boundary conditions of Dirichlet type are applied on the velocity field. The procedure used in the previous section is followed here and five calculations at $Re = 10$ are performed to investigate the behaviour of the error estimated using (66).

For this flow, Fig. 5 shows clearly that only a second-order accuracy is obtained despite the use of sixth-order schemes.[8] Such behaviour can be expected if the implicit assumption about the pressure boundary conditions is taken into account. In fact, as already stated in Section 4.4 (see also Appendix A), the spectral treatment of the pressure through a cosine expansion (26) introduces a second-order error that is logically reported on the velocity field. For the present flow dominated by wall effects, this assumption inhibits by construction the global accuracy of the solution to second-order. Moreover, the comparison of the solution based on a second-order differentiation for the pressure (while keeping the discretization based on sixth-order compact schemes for the convective and viscous terms) with its counterpart obtained using only sixth-order compact schemes (except near the boundary) reveals that the former is more accurate than the latter. This rather unexpected result can be explained by recalling that the use of the standard second-order scheme (39) does not require ghost conditions like (15) or (16) as already stated in Section 4.4. For the present cavity flow, the use of ghost conditions, unavoidable when sixth-order schemes are used, seems to dominate the overall error by taking a slight advantage to the standard scheme (39) based on a minimum stencil. In first conclusion, for this laminar and steady cavity flow, the use of quasi-spectral schemes combined with ghost conditions for the pressure appears to be a useless strategy with no improvement of the convergence order and even a slight degradation of the overall accuracy. This conclusion will have to be reconsidered in next sections where less formal tests are performed in the context of numerical prediction of near-wall transition and turbulence.

Finally, Fig. 5 reports the convergence behaviour observed when the velocity Dirichlet conditions (67) are imposed with an IBM combined with sixth-order differentiation. In present flow configuration, the forcing field $\mathbf{f}$ corresponds to the expression (69) in the fluid region ($\varepsilon = 0$) whereas in the immersed region ($\varepsilon = 1$), it is calibrated to ensure the no-slip boundary conditions (67) via the expression (10). This test shows that a second-order accuracy can be obtained with an additional slight increase of the error that can be accepted regarding the crude approximation of no-slip conditions allowed by

[8] For clarity, only the results obtained with a regular and staggered mesh are presented in Fig. 5. Similar second-order convergence is observed using a collocated mesh but with an increase of the error of about 100%, showing that the use of a partially staggered mesh leads to a drastic improvement of the solution accuracy in presence of no-slip walls.

IBM. Note in particular that the pressure is not at all constrained in IBM since only the velocity is forced in the immersed region without any care about the pressure boundary condition (see the Appendix of [9] for this point). The behaviour of the pressure for a less simple flow configuration (where the mesh does not fit the immersed boundary region) is investigated in the following section.

7.3. Steady flow around a circular cylinder

In this section, the ability of present code combined with an IBM is tested to provide relevant physical predictions. The forcing field $\mathbf{f}$ is again calibrated in order to model the presence of a circular cylinder by the prescription of a target velocity $\mathbf{u_0}$ ensuring no-slip boundary conditions and regularity at its surface (see Section 2.4 and [34,33] for more details). A DNS of a steady wake flow is performed, created by a constant velocity U flow over a circular cylinder of diameter D at $Re = UD/\nu = 40$. The computational domain $(L_x, L_y) = (20D, 12D)$ is discretized on $n_x \times n_z = 361 \times 217$ mesh nodes equally spaced. The streamwise location of the cylinder is $x_{cyl} = 5D$. At $y = \pm L_y/2$, free-slip boundary conditions are assumed. In the streamwise direction, inflow/outflow boundary conditions are imposed as velocity Dirichlet conditions. At the inlet, $\mathbf{u}(0, y, t) = U\,\mathbf{e_x}$ is prescribed while at the outlet, the imposed exit velocity $\mathbf{u}(L_x, y, t)$ is deduced from the simplified equation

$$\left.\frac{\partial \mathbf{u}}{\partial t}\right|_{x=L_x} + U \left.\frac{\partial \mathbf{u}}{\partial x}\right|_{x=L_x} = \mathbf{0} \tag{72}$$

As for a non-slip boundary condition, a homogeneous Neumann condition is assumed at $x = 0$ and $x = L_x$ so that pressure can be defined in spectral space through a cosine expansion. In consequence, the full spectral treatment of the pressure proposed in Section 4 can be used for the present combination of free-slip (in y) and inflow/outflow (in x) boundary conditions.

To investigate the effect of the pressure mesh staggering, only two calculations are performed, based on sixth-order compact schemes used in their collocated (13) and staggered (20) versions for the pressure. By comparison with reference results [43], the main characteristic length scales are satisfactory recovered for both calculations. For instance, the length of the wake bubble is well predicted with a value of $2.30D$ ($\pm 0.03D$) in good agreement with [43] who found $L_w = 2.27\ D$. A more extensive investigation of the velocity or vorticity fields does not allow a clear distinction between the two calculations based on a collocated or a partially staggered approach. However, some discrepancies are found for the pressure field with oscillations for the collocative approach (Fig. 6) whereas the partial staggered approach seems to cancel efficiently this phenomenon. The zigzag pattern of the collocated pressure does not correspond, strictly speaking, to pressure modes because these singular modes are easily cancelled in spectral space during the stage (42) of the Poisson solving. However, as already discussed in Section 4.5, pressure modes near the singular conditions can be highly amplified by the division performed in (42) so that pressure wiggles can be created. Here, this phenomenon is not observed for the partially staggered mesh where singular modes are, by construction, less numerous than in the collocated case while corresponding to oblique waves of very small length. For the collocated case, Fig. 6 reveals that small waves aligned with the mesh are excited, these non-physical oscillations corresponding to "almost singular pressure modes". Note that this unrealistic behaviour of the collocated pressure is only observed for calculations using IBM. Indeed, the use of IBM tends to excite pressure oscillations in the

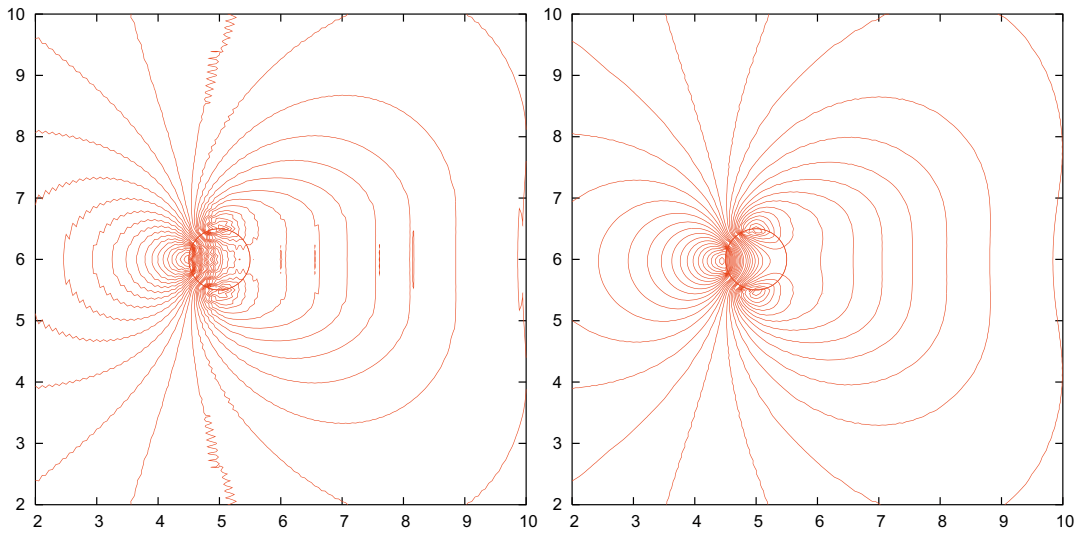

Fig. 6. Pressure field isocontours of a flow around a circular cylinder modelled with an IBM on collocated (left) and staggered (right) meshes.

neighbourhood of the immersed boundary because in this region, the velocity field is not as smooth as in the core fluid zone. This residual non-regularity is then strongly amplified by the division in (42) for near-singular pressure modes for which the factor F_{lmn} tends to be very small. The relation (42) allows the straightforward identification of these almost singular modes, but naturally, similar effects would be recovered if the pressure solving was performed in physical space, with additional difficulties associated with the treatment of actual singular modes that cannot be easily skipped in this case.

To conclude this section, the potential of IBM to excite unrealistic pressure wiggles leads us to renounce to a fully collocated mesh. A partially staggered mesh seems to be able to avoid this phenomenon without introducing any additional difficulties in terms of code development or computational cost. More precisely, in physical space, the cost of a staggered scheme (20) instead of a collocated one (13) is equivalent. In spectral space, the same can be said for the solving of (42) using (43) or (44). The single significant additional computational effort is related to the use of mid-point interpolation scheme (21) for the computation of the term $\nabla \cdot [(1-\varepsilon)\mathbf{u}^{**}]$ in (11), with a resulting increase of the cost of the incompressibility treatment less than 1%. Consequently, our recommendation is to always prefer partially staggered meshes that do not exhibit significant drawbacks compared with collocated meshes while offering a more realistic pressure prediction. For this reason, only results obtained with a partially staggered sixth-order approach will be presented for the remaining benchmark problems.

7.4. Transitional channel flow

In this section, as well as in the remaining ones, comparisons between quantitative predictions obtained from present partially staggered code and those provided by a more conventional code based on second-order schemes and a fully staggered approach will be shown. First, the temporal development of an unstable wave in a plane channel is considered. For this purpose, an initial random perturbation is superimposed on a Poiseuille profile

$$\mathbf{u}(y) = \left(1 - \frac{y^2}{h^2}\right) U_c \ \mathbf{e_x} \tag{73}$$

where h is the half-width of the channel while U_c is the centreline velocity, x, y and z being the longitudinal, normal to the wall and spanwise directions, respectively. By considering a supercritical Reynolds number $Re = U_c h/\nu = 8000$ and a very low amplitude of the perturbation $u' \approx 10^{-10} U_c$, it can be expected that the most amplified Tollmien–Schlichting wave (predicted by the linear stability) dominates all the other waves through an exponential increase during the time. Because this most amplified wave has no z-dependence, present calculations are only 2D. The computational domain matches exactly the expected most amplified mode of stability with $L_x \times L_y = 2\pi h \times 2h$ discretized using $n_x \times n_y$ mesh nodes. A refined mesh near the walls could be slightly less computationally expensive, but because present tests are cheap, a regular mesh is used in both directions with no stretching error. Calculations are performed using a very small time step $\Delta t = 0.0075\ h/U_c$ so that time errors are always negligible with respect to those associated with the spatial discretization. Periodic boundary conditions are used in x-direction while no-slip boundary conditions are imposed at the two walls $y = \pm h$.

As reported by Zang and Krist [44], the amplification rate $\omega_i = 0.002664$[9] of the most unstable Tollmien–Schlichting wave can be recovered with a 6-digit agreement using $n_y = 64$ Chebyshev modes in a purely spectral method. Note that the present problem is known to be numerically stringent. The computation of the temporal evolution of a linearly unstable perturbation for viscous stability is known to require the use of high-order methods in order to capture satisfactory the balance between the convective and viscous terms in Navier–Stokes equations. In this context, purely spectral methods allowing the exponential convergence can be considered as optimal. Here, because only second-order can be formally achieved, our comparisons are restricted to the prediction of the amplification rate with 4 characteristic digits $\omega_i = 0.0026$ with a relative error less than 2%. For the second-order version of present code of [32], it should be noted that $n_y = 391$ mesh nodes are required to obtain this accuracy, while $n_y = 201$ are necessary if sixth-order code is used. This observation shows that a clear improvement of the accuracy can be obtained if sixth-order schemes are used, although the use of ghost conditions implies that both codes are formally second-order accurate. Note that a simulation of the Tollmien–Schlichting wave using $n_y = 201$ mesh nodes with the fully staggered second-order code leads to an amplification rate significantly underestimated with $\omega_i = 0.0024$.

Concerning the discretization in x, the 4-digit accuracy can be achieved using $n_x = 32$ with the fully staggered second-order code against $n_x = 12$ with the present partially staggered sixth-order code. This difference can be expected because in this case, due to the use of periodic boundary conditions, the formal sixth-order accuracy is preserved. Combining the present requirements in x- and y-directions, it can be concluded that $n_x \times n_y = 12 \times 201$ mesh nodes are necessary to satisfactory compute the evolution of a Tollmien–Schlichting wave using the present sixth-order code, while about 5 times more mesh nodes need to be used to have an equivalent accuracy with the second-order code. This conclusion could appear contradictory to the numerical tests in the cavity presented in Section 7.2 where second-order schemes were found to lead to slightly more accurate results. However, this previous test corresponds to a stationary flow where the smooth character of the solution tends to emphasize the errors associated with the discretization of the pressure near the boundaries. Such a configuration seems to penalize a high-order approach requiring ghost conditions. In the present unsteady problem, the sensitivity of

[9] Using the normalisation $h = 1$ and $U_c = 1$.

the solution to numerical errors seems to be of different nature, the "local" sixth-order accuracy being found to be very attractive for the accuracy at marginal resolution.

7.5. Vortex-dipole rebound from a wall

As an intermediate unsteady problem between the prediction of a linearly unstable wave and the description of fully developed turbulence in a channel flow (next section), the dynamics of a vortex-dipole rebound is considered here as a relevant benchmark problem. The main features of the successive events associated with the dipole rebound from a wall are reported by [31]. In this previous study, the physical complexity of this fully non-linear problem is clearly exhibited, the vortex interactions involving multi-scale processes like strong vorticity production at the wall, vorticity sheet detachment and pairing. The combination of these effects leads to several impinging of the structures on the wall where secondary and tertiary structures are created. More recently [7] have emphasized the difficulty to compute accurately the full dynamics of a vortex-dipole rebound. The requirement for this problem, in terms of spatial resolution, is found to be unexpectedly high, even when the spectral accuracy is reached through the use of Fourier or Chebyshev modes. From this point of view, the most demanding phenomena correspond to high-vorticity patches created at the wall through the no-slip boundary condition. A slightly misrepresentation of these patches can lead to erroneous predictions in the vortex interactions that follow the first impinging of the vortex-dipole. Due to its high sensitivity to the numerical accuracy near the wall, the present problem of vortex-dipole rebound can be considered as a very stringent benchmark to evaluate the impact of the present use of ghost conditions near the boundaries. The methodology proposed by [7] is followed to initiate a dipole between two walls through the superimposition of two isolated counter-rotating monopoles. The computational configuration of the previous section is

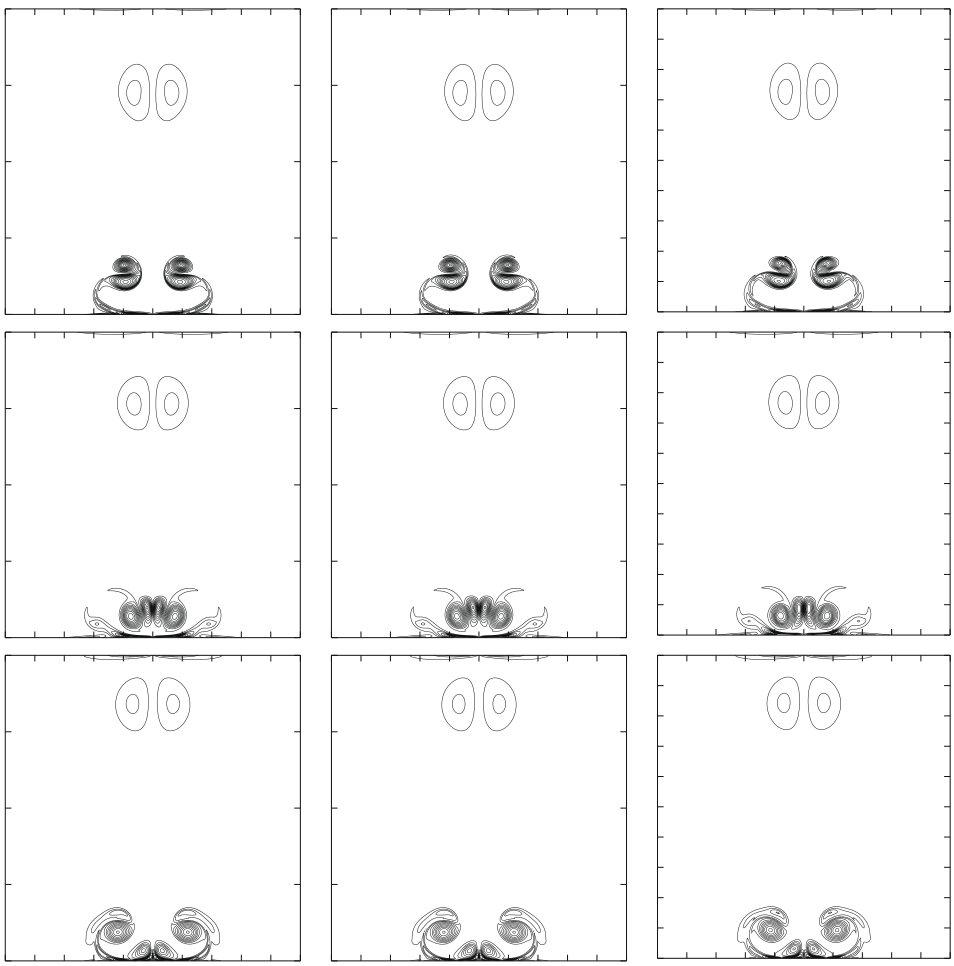

Fig. 7. Vorticity isocontours of the vortex-dipole rebound at three characteristic times. From top to bottom: $t = 0.5, 0.6, 0.8\,h/U$. Left: spectral code of [15] using 2048×1024 Fourier–Chebyshev modes. Centre: present partially staggered sixth-order code using 256×513 mesh nodes. Right: fully staggered second-order code of [32] using 256×513 mesh nodes.

applied with no-slip conditions at $y = \pm h$ and periodicity in x-direction. The computational domain is a simple square $L_x \times L_y = 2h \times 2h$ using $n_x \times n_y$ mesh nodes. The vortex rebound obtained in this exact domain configuration is reported in [15] where the authors used a Fourier–Chebyshev code with a spatial resolution up to 2048×1024 spectral modes. To evaluate the accuracy of the present code for the vortex rebound problem, the highly accurate solution obtained by [15] is considered as the reference one. The initial condition is exactly the same as in [15] where the counter-rotating monopoles of radius $0.1h$ are located in the centre of the channel ($y = 0$) with an horizontal separation of $0.2\,h$. The Reynolds number $Re = Uh/\nu$ is 1000, where U is a velocity scale defined through the initial value $E(0) = 2U^2$ of the total kinetic energy $E(t)$ defined as

$$E(t) = \frac{1}{2h^2}\int_{-h}^{h}\int_{-h}^{h}\mathbf{u}^2(\mathbf{x},t)\,dxdy \tag{74}$$

Note that for the vorticity distribution used to define the monopoles, the total enstrophy defined as

$$Z(t) = \frac{1}{2h^2}\int_{-h}^{h}\int_{-h}^{h}\omega_z^2(\mathbf{x},t)\,dxdy \tag{75}$$

takes the initial value $Z(0) = 800U^2/h^2$. For more details about the initial vorticity field used here (that is virtually zero at the two walls $y \pm h$), see the reference studies [7,15].

In this section, the accuracy of the present code is compared with its spectral counterpart [15]. Additionally, a simulation using the second-order code of [32] is also performed. The purpose is to compare the present partially staggered approach with the more standard fully staggered approach while investigating the interest of sixth-order schemes despite the

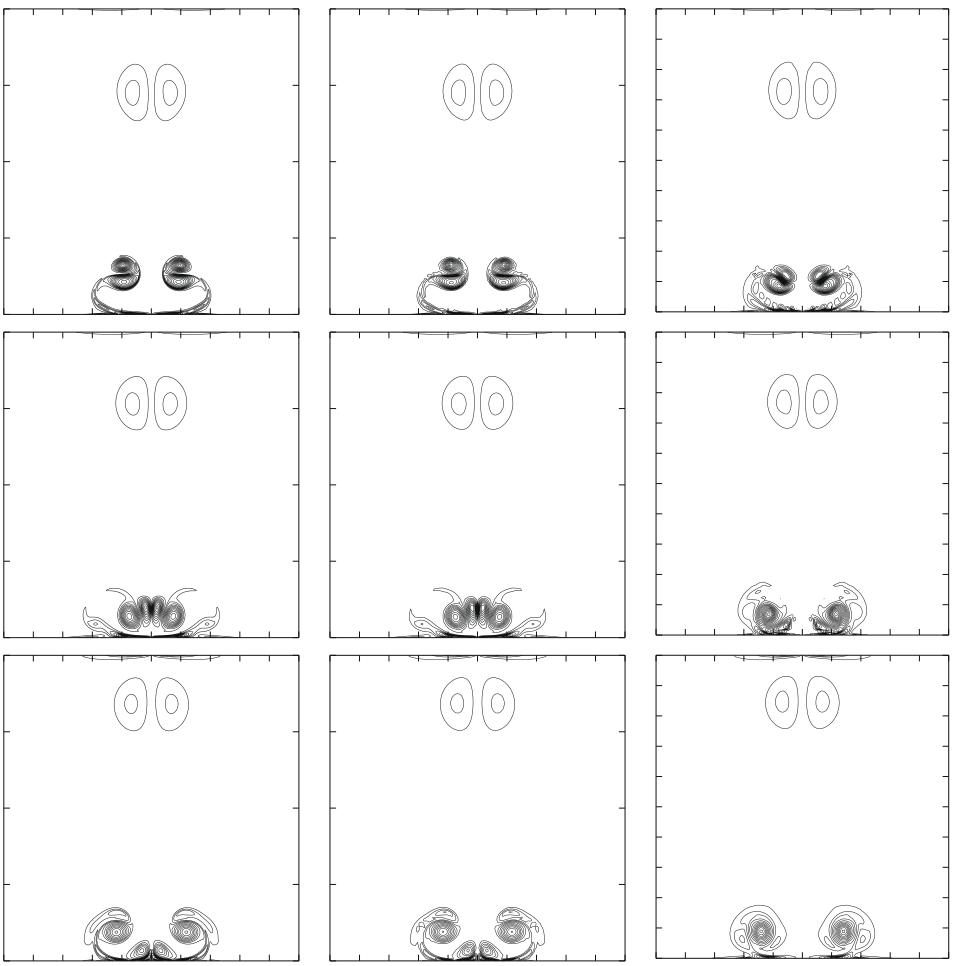

Fig. 8. Vorticity isocontours of the vortex-dipole rebound at three characteristic times. From top to bottom: $t = 0.5, 0.6, 0.8\,h/U$. Left: spectral code of [15] using 2048×1024 Fourier–Chebyshev modes. Centre: present partially staggered sixth-order code using 128×257 mesh nodes. Right: fully staggered second-order code of [32] using 128×257 mesh nodes.

introduction of second-order accuracy near the wall, the more sensitive region with respect to numerical accuracy for this physical problem. Two different refinements are used with $n_x \times n_y = 256 \times 513$ and $n_x \times n_y = 128 \times 257$. These two spatial resolutions are significantly coarser than the reference one [15]. Indeed, the present purpose is to evaluate the behaviour of each code at marginal resolution, the typical situation for which the present sixth-order code is dedicated. For both codes, the mesh is refined near the walls. In practice, the mapping function (54) has been used for the present sixth-order code, using $\gamma = 1$ and $\delta = 1/2$ while adjusting the couple of stretching parameters (α, β) to ensure $\Delta y = 1.5 \times 10^{-3}\ h$ at $y = \pm h$. A comparable stretching was used for the fully staggered second-order code but with the help of an hyperbolic tangent as explained in [32].

For qualitative comparisons, vorticity isocontours obtained at three typical instants of the vortex rebound are presented in Figs. 7 and 8 for the two codes under study with the data of [15] as reference results. For the highest resolution presented here (see Fig. 7), both sixth- and second-order codes allow a physically realistic description of the vortex dynamic associated with the vortex-dipole collision. Compared with the very accurate results of [15], the only visible differences concern the results issued from the second-order code, the present sixth-order code leading to virtually identical results. However, a more quantitative analysis shows that while the time evolution of the kinetic energy is well predicted by the two codes (see Fig. 9, top right), significant errors can be observed concerning the total enstrophy (see Fig. 9, bottom right), especially for the first and second peak (associated with the first and second rebound, respectively) that are too early predicted and underestimated. These discrepancies are strongly reduced using the sixth-order code for which a satisfactory agreement with the spectral results of [15] can be noticed in Fig. 9 (bottom left).

As expected, the use of a coarser mesh reinforces these trends. For the vorticity dynamics, the low resolution cases presented in Fig. 8 emphasize the interest of present sixth-order schemes that allow us to simulate realistically the vortex rebound despite the use of a very coarse mesh. Only some slight wiggles can be observed in the corresponding vorticity plots with minor influence on the vortex dipole collision process. The good behaviour of the present sixth-order code can be confirmed by examining the total enstrophy evolution that remains accurately predicted for this computationally cheap calculation. In contrary, the second-order code leads to unrealistic prediction from the first rebound, with a continual increase of the discrepancy with reference results of [15] when the flow is evolving in time. This loss of accuracy can been directly seen

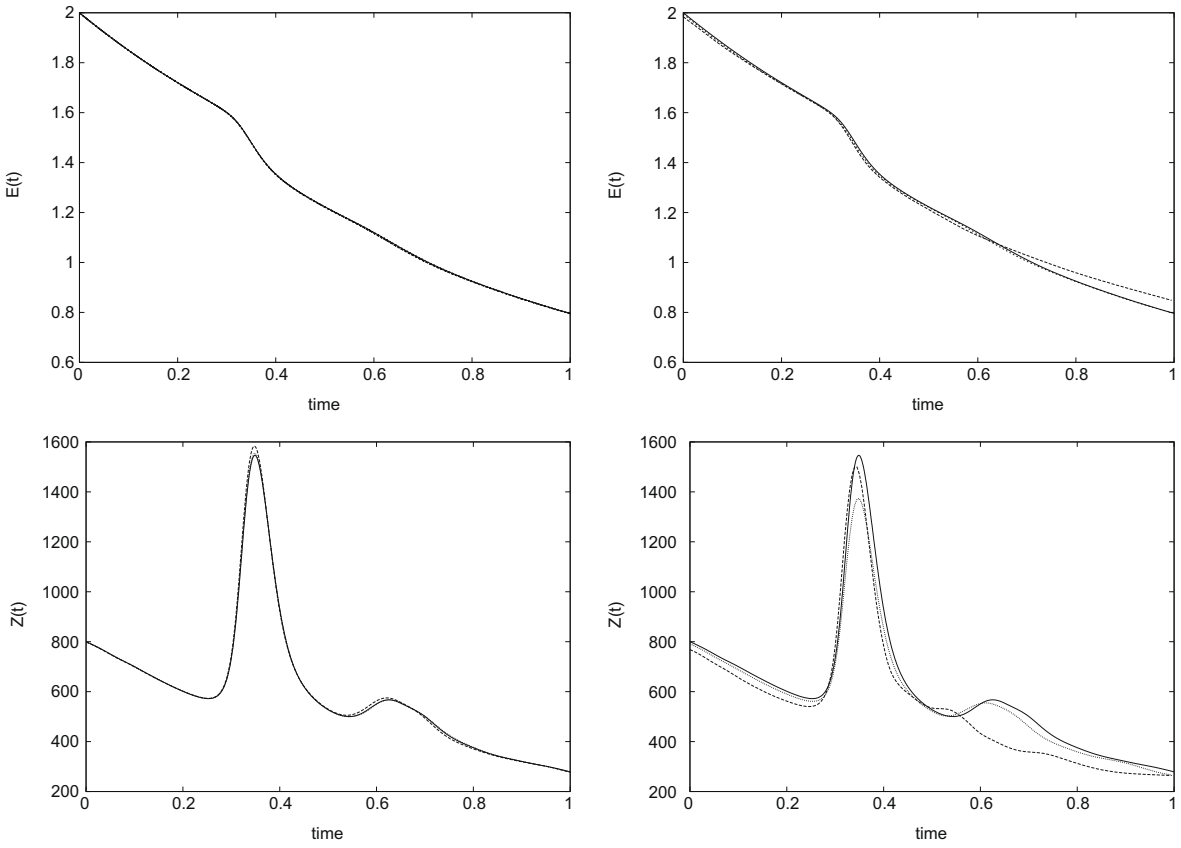

Fig. 9. Time evolution of the total energy $E(t)$ (top) and enstrophy $Z(t)$ (bottom) defined by (74) and (75), respectively during the vortex-dipole rebound. Left: comparison with the present sixth-order code. Right: comparison with the second-order code of [32]. — : reference results of [15]. – – – : simulations using 128×257 mesh nodes. - - - : simulations using 256×513 mesh nodes.

in vorticity visualisations presented in Fig. 8 where the vortex dynamics is found to be unreliable. Even the total kinetic energy is damaged during the simulation (see Fig. 9, bottom right) while the time evolution of the total enstrophy shows an underestimation of the first peak,[10] the second one being completely missed by the fully staggered second-order code when the coarse mesh is used.

Note that the total kinetic energy and enstrophy are found to be significantly underestimated at $t = 0$ for the fully staggered second-order code at low resolution, despite the use of the same initial condition for both codes. This underestimation is just an artefact related with the numerical integration required by definitions (74) and (75). When a fully staggered mesh is used, the kinetic energy has to be first computed on pressure nodes through an interpolation procedure that tends to reduce the overall accuracy. Combined with the numerical integration based on the trapezoidal rule in y-direction, this error is responsible of the slight reduction of the total kinetic energy at $t = 0$. For the total enstrophy, interpolation and integration errors are also introduced, with an additional contribution of the spatial derivation to compute the z-vorticity component, leading to the marked difference of the initial total enstrophy between the sixth- and second-order cases despite the use of exactly the same analytical initial condition given by [7,15]. In conclusion, present validations demonstrate the gain in accuracy offered by the use of compact sixth-order schemes despite the use of a second-order approximation for the pressure, even for the present benchmark flow highly sensitive to the actual accuracy reached at the wall. Hence, in agreement with the tests based on a linearly unstable Tollmien–Schlichting wave in previous section, and contrary to the steady flow configuration of the cavity considered in Section 7.2,[11] the use of ghost conditions for the pressure combined with high-order schemes seems to increase advantageously the numerical accuracy. This property allows the present partially staggered sixth-order code to efficiently simulate very complex vortical motions near a no-slip wall with exactly the same method as for the pressure treatment when free-slip conditions are considered. This ability will be examined in the context of fully developed wall turbulence in the next section.

7.6. Turbulent channel flow

For the last validation test, a turbulent flow in presence of walls is investigated. The non-linear and multi-scale natures of that kind of flows correspond to the main application of the present code, dedicated to DNS/LES of turbulent flows. Note that the flow configuration is the same as in the two previous sections (same boundary conditions in x- and y-directions), with periodicity in z-direction. Two different DNS have been performed using the present partially staggered sixth-order code and using the fully staggered second-order code of [32]. For both calculations, turbulent statistical data have been collected on a time period long enough to reach a statistical convergence. In this section, our analysis focuses on the ability of each DNS to provide relevant profiles of fluctuating velocity, vorticity or pressure root mean square (r.m.s). Corresponding statistics associated with the previous DNS of [29] are presented for comparison. In this reference calculation, pseudo-spectral methods are used using $n_x \times n_y \times n_z = 128 \times 129 \times 128$ de-aliased modes to discretize a computational domain of $L_x \times L_y \times L_z = 4\pi h \times 2h \times 4\pi h/3$ for a nominal value of $h^+ = 180$ where superscript $^+$ indicates a conventional scaling based on the friction velocity u_τ and the kinematic velocity ν. For the reference data, the exact value is $h^+ = 178.1$ while the Reynolds number based on the bulk velocity is $Re = U_m h/\nu = 2793$. Both calculations presented here have been carried out using the same computational domain while a constant longitudinal mean pressure gradient is imposed to also reach the nominal value of $h^+ = 180$. To evaluate the behaviour of each code at marginal resolution, a coarse mesh of $n_x \times n_y \times n_z = 128 \times 129 \times 84$ nodes is used. It is important to note that this resolution corresponds to the one tested by [32] to show the ability of a fully staggered second-order code to provide accurate turbulent statistics in a channel. Note in particular that the reference DNS of [29] based on spectral methods is more computationally expensive, not only because of the 128 Fourier modes in z-direction (instead of 84 mesh nodes in the present DNS), but also because of the de-aliasing procedure that requires to consider 50% more collocation nodes (3/2 rule) in each direction. For the present turbulent channel flow configuration, when the sixth-order code is used, aliasing errors are reduced using the set of coefficients with $c \neq 0$ for the second derivative calculation (14), whereas no aliasing treatment is performed when using the second-order code. As in the previous section, a stretching is applied in y-direction to concentrate mesh nodes near the walls at $y = \pm h$, with the first adjacent node to the wall located at $y^+ = 1$ for both codes.

Compared with reference data of [29], mean velocity profiles are found to be in excellent agreement for the present partially staggered sixth-order code but also for the fully staggered second-order code. Note that the values of friction velocity associated with present statistics leads to $h^+ = 180.2$ and $h^+ = 182.2$ for the sixth- and second-order code, respectively, with the same corresponding Reynolds numbers $Re = 2843$, in good agreement with those obtained by [29]. Profiles of velocity and vorticity r.m.s obtained using the second- or sixth-order code are presented in Figs. 10 and 11. The improvement of statistical predictions offered by the use of sixth-order schemes can be clearly observed. Reminding that results of [29] have the spectral accuracy, the excellent agreement obtained using the present partially staggered sixth-order code can be considered

[10] Note that the prediction of the first peak is unexpectedly better than for the higher resolution. This misleading behaviour can be attributed to an artefact of spurious vorticity creation at the wall that artificially offsets the misrepresentation of the first collision.

[11] Note that in additional tests (not presented in this paper for clarity), the use of second-order schemes only for the pressure (slightly more accurate than their sixth-order counterparts for the cavity flow), has been found clearly less favourable for the vortex-dipole collision problem, a full treatment using sixth-order schemes being by far the best combination for all the unsteady problems considered.

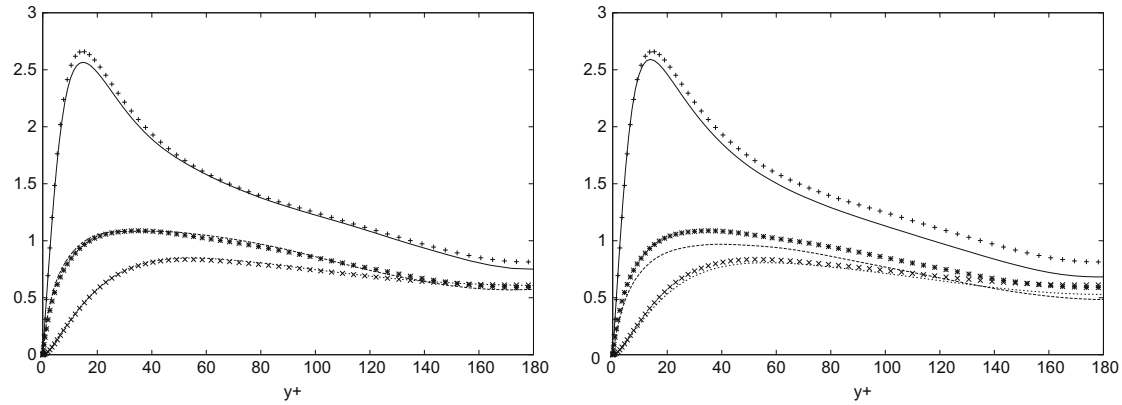

Fig. 10. R.m.s of the fluctuating velocity components in wall units. Lines: present, sixth-order DNS (left), second-order DNS using the code of [32] (right). Symbols: [29]. —, +: $\sqrt{u'u'}$ – – –, ×: $\sqrt{v'v'}$; - - -, *: $\sqrt{w'w'}$.

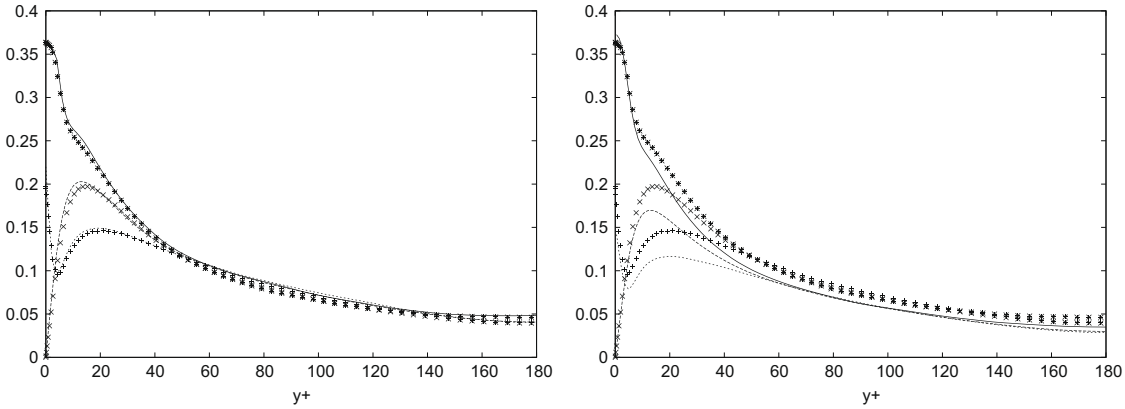

Fig. 11. R.m.s of the fluctuating vorticity components in wall units. Lines: present, sixth-order DNS (left), second-order DNS using the code of [32] (right). Symbols: [29]. —, +: $\sqrt{\omega'_x\omega'_x}$; – – –, ×: $\sqrt{\omega'_y\omega'_y}$; - - -, *: $\sqrt{\omega'_z\omega'_z}$.

as being satisfactory. Indeed, the second-order approximation introduced by the use of ghost conditions (in y-direction) does not seem to deteriorate the quality of statistical predictions. The use of standard second-order schemes based on a fully staggered approach (where no ghost conditions are necessary) does not lead to the same accuracy, especially for the prediction of the spanwise velocity fluctuations that are found significantly underestimated almost everywhere. Concerning the two other Reynolds stress components, a slight under-prediction can also be observed, but mainly in the core of the channel. Note that these behaviours are fully consistent with the observations reported by [32] who emphasized the remarkable ability of a fully staggered second-order code to provide comparable turbulent statistics with those given by a spectral code. Thus, as far as Reynolds stresses are concerned, the improvement offered by sixth-order schemes gives a significant but not decisive advantage to the present code. However, if additionally accurate predictions of vorticity fluctuations are expected, a more marked discrepancy with [29] is found using the fully staggered second-order code whereas a very good agreement is again recovered with the present partially staggered sixth-order code (see Fig. 11). The better prediction of vorticity fluctuations provided by the partially staggered sixth-order code confirms the ability of high-order schemes to maintain accuracy at the smallest scales allowed by the computational mesh.

A rather critical question for incompressible code is related to the physical relevance of the pressure field used to ensure the incompressibility. Schematically, a pressure field computed through a Poisson equation can lead to a satisfactory divergence free condition for the velocity while not being accurate enough to be used for physical analysis. This behaviour can be spectacular when singular modes contaminate the pressure field without having significant effect on pressure gradients in the governing equations. In the present code, singular modes are perfectly cancelled so that the pressure field should be physically relevant. To check the lack of unphysical oscillations on the pressure, several instantaneous fields have been extensively analysed with the help of 2D and 3D visualisations, and no anomaly has been detected. To evaluate quantitatively the truthfulness of the pressure estimation, the r.m.s of fluctuating pressure are presented in Fig. 12. As for the velocity and vorticity statistics, an excellent agreement with reference data of [29] is found with the partially staggered sixth-order

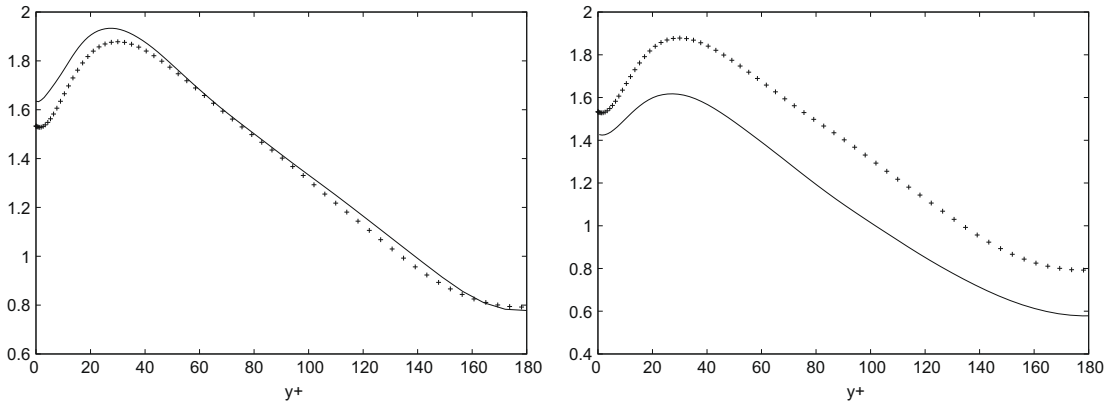

Fig. 12. R.m.s of the fluctuating pressure. Lines: present, sixth-order DNS (left), second-order DNS using the code of [32] (right). Symbols: [29].

code. By contrast, the agreement for the pressure statistics is found to be less favourable when the fully staggered second-order code is used even if the staggered approach eliminates all the singular modes without requiring to remove them in spectral space. This significant change in the quality of statistics depending on the physical quantity under study suggests that the pressure is more demanding in terms of numerical accuracy.

In conclusion, for the present developed turbulent flow, in agreement with the tests presented in the previous sections for the early transitional case and the dipole-vortex rebound, the use of sixth-order schemes is found to reduce significantly the computational cost requirement by allowing the use of a marginal resolution to describe accurately the flow dynamics. Note finally that this advantage is more decisive for the prediction of statistical quantities highly sensitive to small scale motions as the vorticity, as well as the pressure r.m.s that is found to be more receptive to numerical errors than Reynolds stresses at marginal resolution.

8. Discussion and conclusion

A numerical strategy suitable for DNS/LES of incompressible flows is presented in this paper. The approach proposed is based on a mixed method where the spatial differentiation is mainly performed in physical space whereas the treatment of the Poisson equation is carried out in spectral space. The simplicity of the method is presented through basic rules that allow the use of high-order schemes via an equivalent formulation in physical and spectral spaces using the concept of the modified wave number. Compared with a fully spectral code based on Fourier or Chebyshev expansion, the present mixed approach allows us to consider a wider variety of boundary conditions. Moreover, it is shown that an IBM can be favourably combined with the present code in order to consider complex solid body geometries. However, in this specific context, validation results lead us to recommend the use of a partially staggered mesh that eliminates the grid to grid oscillations from the pressure field. Because such a staggering does not introduced a significant computational effort, the fully collocated approach can be left behind whatever the flow configuration considered, with or without IBM.

The computational efficiency of present Poisson solver can be evaluated through its relative cost (with respect to the full solving of Navier–Stokes equations) that is found in practice to be less than 10% or 15% for a regular or stretched mesh respectively. Such a cost, satisfactory regarding the high-order accuracy of present numerical schemes, is made possible by the modified spectral formalism. It allows the direct inversion of the Poisson equation to obtain the incompressibility condition up to the machine accuracy. Furthermore, the non-iterative nature of the solver determines its fixed computational cost whatever the numerical configuration (set of boundary conditions or stretching intensity of the mesh).

In terms of code development, the proposed treatment for the pressure field is light because it only requires the use conventional 3D FFT, available in most of scientific Fortran or C libraries for instance. The partially staggered approach allows the treatment of all types of boundary conditions with only one version of 3D FFT (the more conventional one). Indeed, shifted cosine Fourier transforms are the easiest to deduce from conventional FFT. The other developments for the pressure treatment are based on elementary operations where modified wave numbers and transfer functions just need to be properly defined. The more "advanced" numerical development is related to the use of a stretched mesh (in one direction) that requires to invert a pentadiagonal matrix (basic algebraic operation in the code). Another main advantage of the present pressure treatment is its ability to be easily ported on parallel supercomputers. It is well known that the elliptic nature of the Poisson equation needs important code development to accommodate conventional solver to parallel architecture. However, because parallel version of 3D FFT are already available in several scientific libraries, the portability only concerns the very simple expression of the Poisson equation in spectral space that does not require communications among processors. The full adaptation of present code to parallel supercomputers using Message Passing Interface (MPI)

has been recently carried out. Using a 1-D domain decomposition [36], an excellent scalability of the code has been achieved up to 1024 processors and up to six billion mesh nodes, confirming that present numerical strategy is relevant for parallel computing.

Regarding the accuracy, the present approach allows the use of the same numerical schemes in the three spatial directions whatever the set of boundary conditions. Formally, the use of sixth-order compact schemes leads to a formal sixth-order accuracy when boundary conditions are periodic or free-slip. However, the use of the same schemes for Dirichlet conditions on the velocity (for instance, no-slip or inflow/outflow boundary conditions) needs to perform a specific treatment near the borders that limits the overall algorithm to second-order accuracy in the corresponding direction. However, the technique proposed in this study maintains the spectral nature of the Poisson solver through a relevant use of ghost boundary conditions compatible with cosine Fourier transforms and homogeneous Neumann conditions for the pressure. For the Poisson solver, the use of Neumann condition allows the same treatment of the pressure as for free-slip conditions. In practice, for DNS of unsteady transitional or turbulent flows, the use of sixth-order compact schemes is shown to be advantageous compared with standard second-order schemes. This benefit is preserved even when the formal accuracy is only second order because of the use of no-slip boundary conditions.

In conclusion, present numerical methods seem to be a good compromise in terms of computational cost and accuracy. The various flow configurations considered in this paper lead us to recommend the use of high-order schemes even if the formal accuracy of the overall numerical procedure provides a lower order. DNS/LES of turbulent flows is probably the typical context for which quasi-spectral accuracy seems to be useful, even when it is reduced by the boundary condition treatment. At marginal resolution, our belief is that the improvement of the results offered by the use of high-order schemes is sufficient to justify the moderate numerical developments they need. In comparative tests, it can be shown that the overall increase of the computational cost introduced by the use of sixth-order compact schemes (in the three spatial directions) is about 100% compared with the standard use of second-order schemes. Regarding the gain offered by quasi-spectral accuracy,[12] this increase suggests clearly that the use of high-order schemes based on a partially staggered approach is an efficient strategy to solve the incompressible Navier–Stokes equations. Note finally that the sixth-order version of the present code has already been used for the study of spatially evolving flows (inflow/outflow boundary conditions in the streamwise direction) in presence of immersed boundaries. Some results associated with these studies are reported in [18,19,22,23]. More recently, [20,21] have performed DNS of turbulent flows generated by fractal grids (modelled with IBM) with the new parallel code. The use of a very large number of mesh nodes (up to six billion) has allowed them to consider realistic turbulent states reliable to reference experimental data [14].

Acknowledgments

Calculations were carried out at IDRIS, the computational centre of the CNRS. We are grateful to Jalel Chergui and Jean-Marie Teuler for their very precious help in developing specific and optimized FFT routines. We acknowledge Damien Biau for his helpful remarks during the revision and David Froger for his support for the DNS of turbulent channel flow. We also wish to thank Werner Kramer and Herman Clercx for having kindly provided us their data.

Appendix A. Error introduced by ghost conditions

Near the border $i = 1$, the sixth-order compact scheme (13) can be written

$$\begin{aligned} \alpha f'_0 + f'_1 + \alpha f'_2 &= a\,\frac{f_2 - f_0}{2\Delta x} + b\,\frac{f_3 - f_{-1}}{4\Delta x} \\ \alpha f'_1 + f'_2 + \alpha f'_3 &= a\,\frac{f_3 - f_1}{2\Delta x} + b\,\frac{f_4 - f_0}{4\Delta x} \end{aligned} \tag{A.1}$$

where f_0, f_{-1} and f'_0 are values located outside from the computational domain $x \in [0, L_x]$. For these two expressions, the ghost conditions associated with the assumption that f is symmetric in $x = 0$ can be written

$$f_0 \to f_2,\ f_{-1} \to f_3,\ f'_0 \to -f'_2 \tag{A.2}$$

To evaluate the formal error introduced by with these substitutions, it is easy to show with Taylor series expansion that

$$\begin{aligned} f_2 - f_0 &= 2\Delta x f'_1 + O(\Delta x^3) \\ f_3 - f_{-1} &= 4\Delta x f'_1 + O(\Delta x^3) \\ f'_0 + f'_2 &= 2f'_1 + O(\Delta x^2) \end{aligned} \tag{A.3}$$

[12] In the context of DNS/LES, present tests suggest that the use of sixth-order schemes allows the use of about twice less mesh nodes in each direction for a given accuracy, so that the computational cost of a 3D DNS can be significantly reduced.

For a general function (which does not satisfy the condition $f(x) = f(-x)$) with a zero derivative function f at $x = 0$ ($f'_1 = 0$), the previous relations show that the ghost conditions are third-order accurate for f and second-order accurate for f', so that the two near-border schemes (A.1) can be rewritten as

$$\begin{aligned} f'_1 &= 0 \\ f'_2 + \alpha f'_3 &= a\,\frac{f_3 - f_1}{2\Delta x} + b\,\frac{f_4 - f_2}{4\Delta x} \end{aligned} \tag{A.4}$$

through the introduction of an additional approximation of second-order accuracy. This conclusion can be easily recovered near $i = n_x$ (right boundary), while remaining valid for a staggered scheme of the form (20). The second-order accuracy can also be demonstrated for antisymmetric conditions (assuming in this case $f_1 = 0$) with ghost conditions expressed by the relations (17).

Appendix B. Introduction of aliasing error in the spectral operator for the derivation on a stretched mesh

The purpose of this Appendix is to exhibit the introduction of aliasing when a metric is used to compute a spatial derivative on a stretched mesh in physical space. Furthermore, it will be shown how the spectral operator associated with the same derivative must be modified to lead exactly to the same result (physical/spectral equivalence). Note that only the periodic case is discussed in this section, but similar conclusions can be drawn for symmetric or antisymmetric boundary conditions based on cosine or sine Fourier transforms, respectively.

The product of the metric

$$\frac{1}{h'_i} = a_{-1} e^{\imath k_{-1} s_i} + a_0 + a_1 e^{\imath k_1 s_i} \tag{B.1}$$

by

$$\left.\frac{\partial f}{\partial s}\right|_i = \sum_{m=-n_y/2}^{n_y/2-1} \imath k'_m \hat{f}_m e^{\imath k_m s_i} \tag{B.2}$$

leads to

$$\left.\frac{\partial f}{\partial y}\right|_i = \sum_{m=-n_y/2}^{n_y/2-1} \left(\imath a_{-1} k'_m \hat{f}_m e^{\imath k_{m-1} s_i} + \imath a_0 k'_m \hat{f}_m e^{\imath k_m s_i} + \imath a_1 k'_m \hat{f}_m e^{\imath k_{m+1} s_i} \right) \tag{B.3}$$

To express the convolution product while distinguishing its specific form near the bounds of these truncated Fourier series, two terms have to be rewritten as

$$\begin{aligned} \sum_{m=-n_y/2}^{n_y/2-1} \imath a_{-1} k'_m \hat{f}_m e^{\imath k_{m-1} s_i} &= \sum_{m=-n_y/2-1}^{n_y/2-2} \imath a_{-1} k'_{m+1} \hat{f}_{m+1} e^{\imath k_m s_i} \\ \sum_{m=-n_y/2}^{n_y/2-1} \imath a_1 k'_m \hat{f}_m e^{\imath k_{m+1} s_i} &= \sum_{m=-n_y/2+1}^{n_y/2} \imath a_1 k'_{m-1} \hat{f}_{m-1} e^{\imath k_m s_i} \end{aligned} \tag{B.4}$$

leading to

$$\begin{aligned} \left.\frac{\partial f}{\partial y}\right|_i = & \sum_{m=-n_y/2+1}^{n_y/2-2} \left(\imath a_1 k'_{m-1} \hat{f}_{m-1} + \imath a_0 k'_m \hat{f}_m + \imath a_{-1} k'_{m+1} \hat{f}_{m+1} \right) e^{\imath k_m s_i} + \left(\imath a_0 k'_{-n_y/2} \hat{f}_{-n_y/2} + \imath a_{-1} k'_{-n_y/2+1} \hat{f}_{-n_y/2+1} \right) e^{\imath k_{-n_y/2} s_i} \\ & + \left(\imath a_1 k'_{n_y/2-2} \hat{f}_{n_y/2-2} + \imath a_0 k'_{n_y/2-1} \hat{f}_{n_y/2-1} \right) e^{\imath k_{n_y/2-1} s_i} + \imath a_1 k'_{n_y/2-1} \hat{f}_{n_y/2-1} \underbrace{e^{\imath k_{n_y/2} s_i}}_{= e^{\imath k_{-n_y/2} s_i}} + \imath a_{-1} k'_{-n_y/2} \hat{f}_{-n_y/2} \underbrace{e^{\imath k_{-n_y/2-1} s_i}}_{= e^{\imath k_{n_y/2-1} s_i}} \end{aligned} \tag{B.5}$$

Due to the identity $e^{\imath k_m s_i} = e^{\imath k_{m+n_y} s_i}$, the two last harmonics of this expression are aliased so that the final expression of the (aliased) convolution product is

$$\begin{aligned} \left.\frac{\partial f}{\partial y}\right|_i = & \sum_{m=-n_y/2+1}^{n_y/2-2} \left(\imath a_1 k'_{m-1} \hat{f}_{m-1} + \imath a_0 k'_m \hat{f}_m + \imath a_{-1} k'_{m+1} \hat{f}_{m+1} \right) e^{\imath k_m s_i} \\ & + \left(\imath a_0 k'_{-n_y/2} \hat{f}_{-n_y/2} + \imath a_{-1} k'_{-n_y/2+1} \hat{f}_{-n_y/2+1} + \imath a_1 k'_{n_y/2-1} \hat{f}_{n_y/2-1} \right) e^{\imath k_{-n_y/2} s_i} \\ & + \left(\imath a_{-1} k'_{-n_y/2} \hat{f}_{-n_y/2} + \imath a_1 k'_{n_y/2-2} \hat{f}_{n_y/2-2} + \imath a_0 k'_{n_y/2-1} \hat{f}_{n_y/2-1} \right) e^{\imath k_{n_y/2-1} s_i} \end{aligned}$$

In spectral space through the matricial writing (58), the resulting matrix **A** is

$$\mathbf{A}=\begin{pmatrix} \imath a_0 k'_{-n_y/2} & \imath a_{-1} k'_{-n_y/2+1} & & & & & \imath a_1 k'_{ny/2-1} \\ \imath a_1 k'_{-n_y/2} & \imath a_0 k'_{-n_y/2+1} & \imath a_{-1} k'_{-n_y/2+2} & & & & \\ \ddots & \ddots & \ddots & & & & \\ & \imath a_1 k'_{m-1} & \imath a_0 k'_m & \imath a_{-1} k'_{m+1} & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \imath a_1 k'_{n_y/2-3} & \imath a_0 k'_{n_y/2-2} & \imath a_{-1} k'_{n_y/2-1} & \\ \imath a_{-1} k'_{-n_y/2} & & & & \imath a_1 k'_{n_y/2-2} & \imath a_0 k'_{n_y/2-1} & \end{pmatrix}$$

Then, the aliased matrix $\mathbf{B}=\mathbf{AA}$ associated with the second derivative can be easily deduced. Note that due to the aliasing terms, these two matrices become cyclic. In terms of computational efficiency, the cyclic nature of **B** would increase the cost of the inversion required to solve the Poisson equation under the form (62). However, with the aid of Hermicity ($\hat{f}_m=\hat{f}^*_{-m}$),[13] it is easy to restore the purely pentadiagonal form of **B** in its aliased form.

References

[1] A.S. Almgren, J.B. Bell, W.G. Szymczak, A numerical method for the incompressible Navier–Stokes equations based on an approximate projection, SIAM J. Sci. Comput. 17 (2) (1996) 358–369.
[2] E. Arquis, J.P. Caltagirone, Sur les conditions hydrodynamiques au voisinage d'une interface milieu fluide-milieu poreux: applicationla convection naturelle, C.R. Acad. Sci. 299 (1) (1984) 1–4. Série II.
[3] E.J. Avital, N.D. Sandham, K.H. Luo, Stretched cartesian grids for solution of the incompressible Navier–Stokes equations, Int. J. Numer. Methods Fluids 33 (2000) 897–918.
[4] A.B. Cain, J.H. Ferziger, W.C. Reynolds, Discrete orthogonal function expansions for non-uniform grids using the fast fourier transform, J. Comp. Phys. 56 (1984) 272–286.
[5] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods in Fluid Dynamics, Springer-Verlag, New York, 1988.
[6] H. Choi, P. Moin, Effects of the computational time step on numerical solutions of turbulent flow, J. Comp. Phys. 113 (1994) 1–4.
[7] H.J.H. Clercx, C.H. Bruneau, The normal and oblique collision of a dipole with a no-slip boundary, Comput. Fluids 35 (2006) 245–279.
[8] M.O. Deville, P.F. Fischer, E.H. Mund, High-order Methods for Incompressible Fluid Flow, Cambridge University Press, 2002.
[9] E.A. Fadlun, R. Verzico, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comp. Phys. 161 (2000) 35–60.
[10] M. Fortin, R. Peyret, R. Temam, Résolution numérique des équations de Navier–Stokes pour un fluide incompressible, Journal de Mcanique 10 (3) (1971) 357–390.
[11] A. George, L.C. Huang, W.-P. Tang, Y.D. Wu, Numerical simulation of unsteady incompressible flow ($Re \leqslant 9500$) on the curvilinear half-staggered mesh, SIAM J. Sci. Comput. 21 (6) (2000) 2331–2351.
[12] D. Goldstein, R. Handler, L. Sirovich, Modeling a no-slip boundary condition with an external force field, J. Comp. Phys. 105 (1993) 354–366.
[13] F.H. Harlow, J.E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, Phys. Fluids 8 (12) (1965) 2182–2189.
[14] D. Hurst, J.C. Vassilicos, Scalings and decay of fractal-generated turbulence, Phys. Fluids 19 (035103) (2007).
[15] G.H. Keetels, U. D'Ortona, W. Kramer, H.J.H. Clercx, K. Schneider, G.J.F. van Heijst, Fourier spectral and wavelet solvers for the incompressible Navier–Stokes equations with volume-penalization: Convergence of a dipole-wall collision, J. Comp. Phys. 227 (2007) 919–945.
[16] J. Kim, P. Moin, Application of a fractional-step method to incompressible Navier–Stokes equations, J. Comp. Phys. 59 (1985) 308–323.
[17] A.G. Kravchenko, P. Moin, On the effect of numerical errors in Large Eddy simulation of turbulent flows, J. Comp. Phys. 131 (1997) 310–322.
[18] S. Laizet, E. Lamballais, Direct numerical simulation of a spatially evolving flow from an asymmetric wake to a mixing layer, in: Sixth ERCOFTAC Workshop on Direct and Large-Eddy Simulation, Poitiers, France, 2005.
[19] S. Laizet, E. Lamballais, Simulation numérique directe de l'influence de la forme aval d'une plaque séparatrice sur une couche de mélange – Direct-numerical simulation of the splitting-plate downstream-shape influence upon a mixing layer, C.R. Mecanique 334 (2006) 454–460.
[20] S. Laizet, J.C Vassilicos, Direct numerical simulation of fractal-generated turbulence, in: Proceedings of the DLES-7, Trieste, 2008.
[21] S. Laizet, J.C. Vassilicos, Multiscale generation of turbulence, J. Multiscale Modell. 1 (2009) 177–196.
[22] E. Lamballais, J. Silvestrini, S. Laizet, Direct numerical simulation of a separation bubble on a rounded finite-width leading edge, Int. J. Heat Fluid Flow 29 (3) (2008) 612–625.
[23] E. Laurendeau, P. Jordan, J.-P. Bonnet, J. Delville, P. Parnaudeau, E. Lamballais, Subsonic jet noise reduction by fluidic control: The interaction region and the global effect, Phys. Fluids 20 (101519) (2008).
[24] S.K. Lele, Compact finite difference schemes with spectral-like resolution, J. Comp. Phys. 103 (1992) 16–42.
[25] M. Lesieur, Turbulence in fluids, fourth ed., Springer, 2008.
[26] P. Mercier, M. Deville, A multidimensional compact high-order scheme for 3-D Poisson's equation, J. Comp. Phys. 39 (1981) 443–455.
[27] R. Mittal, G. Iaccarino, Immersed boundary methods, Ann. Rev. Fluid Mech. 37 (2005) 239–261.
[28] P. Moin, Engineering Numerical Analysis, Cambridge University Press, 2001.
[29] R.D. Moser, J. Kim, N.N. Mansour, Direct numerical simulation of turbulent channel flow up to $Re_\tau = 590$, Phys. Fluids 11 (4) (1999) 943–945.
[30] S. Nagarajan, S.K. Lele, J.H. Ferziger, A robust hight-order compact method for Large Eddy simulation, J. Comp. Phys. 191 (2003) 392–419.
[31] P. Orlandi, Vortex dipole rebound from a wall, Phys. Fluids A 2 (8) (1990) 1429–1436.
[32] P. Orlandi, Fluid Flow Phenomena: A Numer. Toolkit, Springer, 2000.
[33] P. Parnaudeau, J. Carlier, D. Heitz, E. Lamballais, Experimental and numerical studies of the flow over a circular cylinder at Reynolds number 3900, Phys. Fluids 20 (085101) (2008).
[34] P. Parnaudeau, E. Lamballais, D. Heitz, J.H. Silvestrini, Combination of the immersed boundary method with compact schemes for DNS of flows in complex geometry, in: Proceedings of the DLES-5, Munich, 2003.
[35] C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comp. Phys. 10 (1972) 252–271.

[13] All the physical quantities are defined real in physical space.

[36] N.D. Sandham, R.J.A. Howard, DNS of turbulence using massively parallel computers, in: D.R. Emerson, A. Ecer, J. Periaux, N. Satofuka, P. Fox (Eds.), Parallel Computational Fluid Dynamic, Elsevier Science B.V, 1998, p. 23–32.
[37] T.M. Shih, C.H. Tan, B.C. Hwang, Effects of grid staggering on numerical schemes, Int. J. Numer. Methods Fluids 9 (1989) 193–212.
[38] A.L.F. Lima E Silva, A. Silveira-Neto, J.J.R. Damasceno, Numerical simulation of two-dimensional flows over a circular cylinder using the immersed boundary method, J. Comp. Phys. 189 (2003) 351–370.
[39] P.N. Swarztrauber, The methods of cyclic reduction, Fourier analysis and the FACR algorithm for the discrete solution of Poisson's equation on a rectangle, SIAM Rev. 19 (1977) 490–501.
[40] F. Tremblay, M. Manhart, R. Friedrich, DNS of flow around a circular cylinder at a subcritical reynolds number with cartesian grid, in: Proceedings of the 8th European Turbulence Conference, EUROMECH, Barcelona, Spain, 2000.
[41] R.B. Wilhelmson, J.H. Ericksen, Direct solutions for Poisson's equation in three dimensions, J. Comp. Phys. 25 (1977) 319–331.
[42] J.H. Williamson, Low-storage Runge–Kutta schemes, J. Comp. Phys. 35 (1980) 48.
[43] T. Ye, R. Mittal, H.S. Udaykumar, W. Shyy, An accurate cartesian grid method for viscous incompressible flow with complex immersed boundaries, J. Comp. Phys. 156 (1999) 209–240.
[44] T.A. Zang, S.E. Krist, Numerical experiments on stability and transition in plane channel flow, Theor. Comput. Fluid Dynam. 1 (1989) 41–64.