High-order implicit residual smoothing time scheme for direct and large eddy simulations of compressible flows

P. Cinnella *, C. Content

DynFluid, Arts et Metiers ParisTech, 151 Boulevard de l'Hopital, 75013 Paris, France

A B S T R A C T

Restrictions on the maximum allowable time step of explicit time integration methods for direct and large eddy simulations of compressible turbulent flows at high Reynolds numbers can be very severe, because of the extremely small space steps used close to solid walls to capture tiny and elongated boundary layer structures. A way of increasing stability limits is to use implicit time integration schemes. However, the price to pay is a higher computational cost per time step, higher discretization errors and lower parallel scalability. In quest for an implicit time scheme for scale-resolving simulations providing the best possible compromise between these opposite requirements, we develop a Runge–Kutta implicit residual smoothing (IRS) scheme of fourth-order accuracy, based on a bilaplacian operator. The implicit operator involves the inversion of scalar pentadiagonal systems, for which efficient parallel algorithms are available. The proposed method is assessed against two explicit and two implicit time integration techniques in terms of computational cost required to achieve a threshold level of accuracy. Precisely, the proposed time scheme is compared to four-stages and six-stages low-storage Runge–Kutta method, to the second-order IRS and to a second-order backward scheme solved by means of matrix-free quasi-exact Newton subiterations. Numerical results show that the proposed IRS scheme leads to reductions in computational time by a factor 3 to 5 for an accuracy comparable to that of the corresponding explicit Runge–Kutta scheme.

1. Introduction

Efficient DNS and LES simulations of compressible turbulent flows require a smart combination of numerical ingredients. One of them is the use of high-order spatial discretization techniques, allowing to minimize the number of grid points required to resolve a given wavelength of the numerical solution [1,2]. On the other hand, computations must be carried out for extended periods of time in order to converge turbulent statistics, so that the choice of a suitable time integration scheme is of the utmost importance for the overall accuracy and efficiency. While explicit schemes provide accurate temporal resolution for LES and DNS, the time step size is dictated by stability constraints of the algorithm rather than by the frequency content of the large-scale structures. This can be particularly severe for low-Mach number and wall-bounded flows [1]. Sources of stiffness that cause severe time step restrictions for explicit schemes include acoustic waves for low-Mach number flows, viscous effects and large variations of the mesh size [3]. In LES of wall-bounded flows, highly stretched meshes are used to capture the fine-scale structures in the turbulent boundary layer [4]. With such small grid sizes, the sta-

bility constraint of explicit time-marching methods becomes very restrictive in the near-wall regions, whereas much larger time steps could be applied to mesh elements far from the wall [3]. This means that the time step imposed by stability limits is much smaller than that required to achieve a satisfactory accuracy level of the solution [5]. A way of relaxing stability constraints consists in using an implicit time integration method. Unfortunately, this generally involves the calculation of flux Jacobians, which may be a computationally intensive task both in terms of memory and operation count. As a consequence, fully implicit schemes are prohibitively expensive to use and some form of partial implicitation [3] or approximate calculation of the Jacobians [1,5] is required to reduce computational cost to an amenable level. Subiteration techniques are generally used to reduce supplementary errors associated to successive simplifications of the implicit operator. However, this introduces additional tunable parameters, like the convergence criterion or the number of subiterations, with a strong impact on the accuracy and stability of the method. Finally, implicit techniques lead to the inversion of large matrices over the computational domain, which has to be dealt with carefully not to deteriorate the overall parallel performance. A too high computational cost of the implicit scheme can be a particularly penalizing factor in view of long time integrations, even if the overall number of iterations is lower than that of an explicit scheme (see [5]). Thus, the choice of a suitable implicit treatment for LES and DNS simulations derives from a delicate compromise between maximum allowable time step given the accuracy and stability constraints of the simulation and computational cost per iteration.

Implicit LES and DNS computations mostly use the backward linear multistep method of second-order accuracy (BW2) [1,5] for the discretization of time derivatives, along with an iterative technique to solve the resulting nonlinear system of equations at each physical time step. The BW2 is A-stable and L-stable, leading to an unconditionally stable fully discrete approximation for any choice of the spatial discretization, but also to increased damping of high-frequency solution modes as the time step grows. Moreover, the error associated to the incomplete convergence of the subiterations is not easy to control. An implicit technique widely used in the past to speed up steady Euler and Navier–Stokes calculations is the so-called implicit residual smoothing (IRS) in conjunction with Runge–Kutta (RK) time stepping [6]. This approach was initially proposed by Lerat and Sides [7] for the Lax–Wendroff scheme, and adapted to RK schemes by Jameson [8]. The basic idea underlying the IRS approach is to run an explicit RK scheme with a time step beyond its stability limit, then smoothing the residual over a set of neighboring cells. Both central and upwind smoothing techniques exist [9,10]. The latter leads to a more efficient damping of solution modes, which is a suitable property for steady computations but undesirable for time accurate ones. Time accurate extensions of central IRS schemes (see [11,12]) have been applied in the past to the simulation of slow unsteady flows. Central IRS uses an implicit Laplacian smoothing operator to filter out high-frequency modes of the residual, which leads to the solution of tridiagonal systems for each space direction and RK stage. The main asset of IRS is that it provides a significant increase of the maximum allowable time step compared to an explicit RK scheme, while keeping a computational cost not much higher, thanks to the efficient inversion of scalar tridiagonal matrices. Nevertheless, care must be taken in the selection of the smoothing coefficient, to avoid introducing additional errors that are inconsequential for steady computations but may be unacceptably high for the highly unsteady LES and DNS ones.

The goal of this paper is to develop an IRS technique suitable for scale-resolving time-accurate simulations. This is achieved by constructing a higher-order IRS scheme based on the application of a bilaplacian smoothing operator to the residuals. Such a scheme leads to the inversion of a scalar pentadiagonal system in each mesh direction, which can be treated in an efficient parallel way [13]. As a result, the proposed IRS method keeps a similar accuracy to that of the underlying RK scheme while reducing the overall computational cost. Applications to compressible low and high Mach number turbulent flows are presented to assess the accuracy and efficiency of the new high order IRS approach. Improvements over the explicit and implicit time schemes commonly used for DNS and LES computations are shown up.

2. Space discretization schemes

We discuss numerical methods for a system of conservation laws of the form:

$$\frac{\partial w}{\partial t} + \sum_d \frac{\partial f_d(w)}{\partial x_d} = 0 \tag{1}$$

where the summation is carried out over the d space dimensions, f_d is the flux function in the dth direction, and x_d is the corresponding space coordinate. The flux functions contain an inviscid and a viscous part, denoted by superscripts e and v, respectively:

$$f_d = f_d^e - f_d^v.$$

Equation (1) is discretized in space by means of some approximation technique, to give

$$w_t + \mathcal{R}(w) = 0 \tag{2}$$

where $\mathcal{R}$ is a space approximation operator.

The main focus of the present paper is on efficient time integration schemes. For space discretization, we consider a ninth-order accurate upwind approximation of the inviscid fluxes on eleven points per mesh direction, obtained by applying an upwind recursive correction to the leading truncation error term of a centered second-order scheme [14,15]. Such a scheme is equivalent to a MUSCL scheme constructed by applying a ninth-order extrapolation to the fluxes. In the following

we refer to this scheme as FE9. For a 1D problem and a regular Cartesian grid with space step h, so that $x_j = jh$ the semi-discrete scheme in space writes:

$$(w_t)_j + \frac{(\delta \mathcal{F})_j}{h} = 0$$

where δ is the classical difference operator over one cell, $\delta(\bullet)_{j+\frac{1}{2}} := (\bullet)_{j+1} - (\bullet)_j$, and $\mathcal{F}_{j+\frac{1}{2}}$ is the numerical flux at cell interface $j + \frac{1}{2}$:

$$\mathcal{F}_{j+\frac{1}{2}} = \left[\left(I - \frac{1}{6}\delta^2 + \frac{1}{30}\delta^4 - \frac{1}{140}\delta^6 + \frac{1}{630}\delta^8 \right) \mu f - \frac{1}{1260} Q \delta^9 w \right]_{j+\frac{1}{2}}$$

where f is the physical flux, μ is the cell average operator, $\mu(\bullet)_{j+\frac{1}{2}} := \frac{1}{2}\left((\bullet)_{j+1} + (\bullet)_j \right)$, and Q is a dissipation matrix, here taken equal to the Roe matrix associated to the physical flux f. This scheme is intrinsically dissipative, with a high-order dissipation term that damps high-frequency oscillations, but not total variation diminishing. For problems involving flow discontinuities, the scheme is modified by changing the dissipation coefficient, by replacing the dissipation matrix Q by the spectral radius of the inviscid fluxes, and by adding a nonlinear artificial viscosity term.

The numerical flux of the modified scheme finally becomes:

$$\mathcal{F}_{j+\frac{1}{2}} = \left[\left(I - \frac{1}{6}\delta^2 + \frac{1}{30}\delta^4 - \frac{1}{140}\delta^6 + \frac{1}{630}\delta^8 \right) \mu f - \mathcal{D} \right]_{j+\frac{1}{2}}$$

with the numerical dissipation term $\mathcal{D}$:

$$\mathcal{D}_{j+\frac{1}{2}} = \left[\varepsilon_2 \delta w + \varepsilon_{10} \delta^9 w \right]_{j+\frac{1}{2}}$$

with $\varepsilon_{2\,j+\frac{1}{2}} = \kappa_2 \lambda^e_{\,j+\frac{1}{2}} max(\nu_j \Phi_j, \nu_{j+1}\Phi_{j+1})$, $\varepsilon_{10\,j+\frac{1}{2}} = max(0, k_{10} - \varepsilon_{2\,j+\frac{1}{2}})$ and the dissipation coefficients $k_2 \in [0,1]$ and $k_{10} \approx \frac{1}{1260}$. In the above, ν_j is the well-known pressure based shock sensor of Jameson et al. [6] and Φ is a discrete form of Ducros' vortex sensor [16], a measure of the relative weight of the local divergence of the velocity field to the velocity rotational. This sensor is $O(1)$ in high-divergence regions and tends to zero in vortex dominated regions, which allows to capture flow discontinuities sharply while minimizing the effect of numerical dissipation on vortical structures. In the following, some results are also presented for a Dispersion Relation Preserving scheme of fourth order accuracy, optimized on 11 points (DRP11) [17] with the sixth-order selective filter proposed in [18] to check the effect of the space discretization.

The above large-stencil schemes require some special treatment close to the domain boundaries. For most boundaries, like inflow/outflow boundaries and inner boundaries between adjacent subdomains for multi-block grids, the mesh is augmented by introducing the number of ghost cells necessary for applying the boundary scheme (5 layers for FE9 and DRP11). For impermeable boundaries, and specifically wall boundaries, we follow a strategy similar to the one discussed in [19]: the physical flux computed from the boundary information is imposed at the first mesh interface (denoted as $j = 1/2$, see Fig. 1), and non-central schemes are used to express the numerical fluxes up to the 4th inner interface from the wall. For FE9, the boundary schemes are 9th-order accurate on non-central 11 point stencils. Precisely, the modified boundary fluxes write:

$$\mathcal{F}_{11/2} = \frac{1}{1260}f_1 - \frac{23}{2520}f_2 + \frac{127}{2520}f_3 - \frac{473}{2520}f_4 + \frac{1627}{2520}f_5$$
$$+ \frac{1627}{2520}f_6 - \frac{473}{2520}f_7 + \frac{127}{2520}f_8 - \frac{23}{2520}f_9 + \frac{1}{1260}f_{10}$$

$$\mathcal{F}_{9/2} = -\frac{1}{2520}f_1 + \frac{17}{2520}f_2 - \frac{143}{2520}f_3 + \frac{967}{2520}f_4 + \frac{2551}{2520}f_5$$
$$- \frac{1397}{2520}f_6 + \frac{787}{2520}f_7 - \frac{353}{2520}f_8 + \frac{2}{45}f_9 - \frac{11}{1260}f_{10} + \frac{1}{1260}f_{11}$$

$$\mathcal{F}_{7/2} = \frac{1}{420}f_1 - \frac{11}{315}f_2 + \frac{401}{1260}f_3 + \frac{2881}{2520}f_4 - \frac{1859}{2520}f_5$$
$$+ \frac{1249}{2520}f_6 - \frac{683}{2520}f_7 + \frac{277}{2520}f_8 - \frac{11}{360}f_9 + \frac{13}{2520}f_{10} - \frac{1}{2520}f_{11}$$

$$\mathcal{F}_{5/2} = -\frac{11}{1260}f_1 + \frac{59}{315}f_2 + \frac{553}{360}f_3 - \frac{3839}{2520}f_4 + \frac{4021}{2520}f_5$$
$$- \frac{691}{504}f_6 + \frac{2257}{2520}f_7 - \frac{1067}{2520}f_8 + \frac{49}{360}f_9 - \frac{67}{2520}f_{10} + \frac{1}{420}f_{11}$$

$$\mathcal{F}_{3/2} = \frac{23}{252}f_1 + \frac{5081}{2520}f_2 + -\frac{1067}{360}f_3 + \frac{11281}{2520}f_4 - \frac{13619}{2520}f_5$$
$$+ \frac{12421}{2520}f_6 - \frac{8327}{2520}f_7 + \frac{3973}{2520}f_8 - \frac{1277}{2520}f_9 + \frac{31}{315}f_{10} - \frac{11}{1260}f_{11}$$

(3)

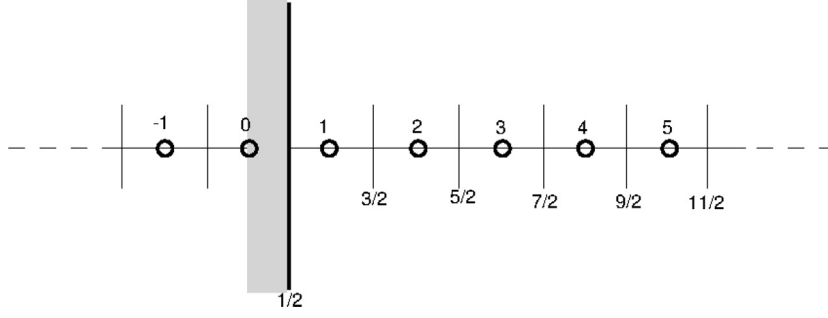For the DRP11, the non-central boundary schemes given in [20] are used.

Fig. 1. 1D sketch of the mesh nodes and interfaces close to a solid boundary.

All of the numerical methods considered in this study are implemented within the in-house code DynHoLab [21]. DynHoLab provides a general framework to easily and fastly develop and test new models and numerical algorithms for CFD: these include discretization schemes, boundary conditions, real-gas models, turbulence models. The main feature of the code is a computational environment independent of specific models or numerical methods used to solve the problem. DynHoLab uses a combination of a compiled and type-safe language (FORTRAN) with an interpreted and dynamically typed language (Python [22]): this way, it benefits from the fast and safe execution of the first one, along with easy and fast development capabilities thanks to the second one. The data-structure is based on a CGNS-tree [23], which provides a fully hierarchical structure to store the data. The code handles parallel computations through an MPI implementation based on its intrinsic multi-block architecture.

3. Time integration schemes

3.1. Explicit Runge–Kutta scheme

The explicit time integration schemes considered in this work are low-storage Runge–Kutta (RK) methods. These may be written as

$$\begin{cases} w^{(0)} = w^n \\ \Delta w^{(k)} = -a_k \Delta t \mathcal{R}(w^{(k-1)}), \quad k = 1, \ldots, s \\ w^{n+1} = w^{(s)} \end{cases} \tag{4}$$

where w^n is the numerical solution at time $n\Delta t$, $\Delta w^n = w^{(k)} - w^{(0)}$ is the solution increment at the kth Runge–Kutta stage, s is the number of stages, a_k are the scheme coefficients, and $\mathcal{R}$ denotes the approximation of the spatial derivatives.

Precisely, in the following we consider the well-known four-stage Runge–Kutta (RK4) proposed by Jameson et al. [6], of fourth-order accuracy for linear problems and second-order accurate in general, and the six-stage optimized RK of Bogey and Bailly [17] (RK6), which is formally only second-order accurate but exhibit very low dispersion and dissipations errors up to a reduced angular frequency of $\frac{\pi}{2}$, *i.e.* the lowest resolved frequency on the time mesh in use.

When coupled with a purely centered approximation for the space derivatives, the RK4 scheme is stable under a CFL condition of the form:

$$CFL \leq \Omega^e 2\sqrt{2}$$

with $CFL = \frac{\Delta t}{h}\lambda^e$ the Courant number, h the local mesh size, λ^e the spectral radius of the inviscid flux jacobians, sum of the velocity magnitude U and of the speed of sound c, and $\Omega^e = 1$ for the second-order centered scheme or $\Omega^e = O(1), \Omega^e < 1$ for higher order centered schemes (see, e.g. Refs. [12,24]). The RK6 scheme, on the other hand, has a stability limit about 40% larger than the RK4 scheme [17].

When solving a viscous flow problem, the schemes have also to satisfy a viscous stability condition of the form:

$$\frac{\Delta t \lambda_v}{h^2} \leq \frac{\Omega^v}{4}$$

with λ^v the spectral radius of the viscous fluxes (proportional to the fluid kinematic viscosity ν) and $\Omega^v = 2.785$ for the specific Runge–Kutta scheme under analysis, if the standard second-order approximation of the viscous fluxes is adopted.

In practice, the simulation time step is selected as the minimum of the time steps allowing to satisfy the inviscid and viscous stability conditions, respectively:

$$\Delta t = min(\Delta t^e, \Delta t^v),$$

where $\Delta t^e = O(h/\lambda^e)$ and $\Delta t^v = O(h^2/\lambda^v)$, so that $\Delta t^v/\Delta t^e = O(h\lambda^e/\lambda^v)$. The latter ratio is easily shown to be $O(Re_h(1 + M^{-1}))$, with $Re_h = hU/\nu$ the grid Reynolds number. For a centered scheme, Re_h has to be of the order of 2 or less to

avoid the appearance of grid-to-grid oscillations. As a consequence, for low Mach number compressible flow computations, $\Delta t^v / \Delta t^e >> 1$ and the main restriction on the time step is due to the inviscid stability constraint (acoustic waves possessing a characteristic time scale much smaller than the flow characteristic time scale), whereas the higher the Mach number is, the more the viscous to inviscid time-step ratio decreases. For DNS and LES of high-Reynolds turbulent flows, characterized by very fine mesh resolutions in the near wall region, Re_h can be lower than 1 and the viscous stability constraint eventually predominates in the determination of the maximum allowable time step.

3.2. Backward difference schemes

As discussed in the introduction, the most used method to relax stability constraints on the time step for DNS and LES simulations consists in approaching the time by a second-order backward multistep scheme (BW2), also called the Gear scheme, which is A-stable and L-stable (see, e.g. [25]). With this choice, the fully discrete scheme writes:

$$F^{n+1} = \frac{Dw^{n+1}}{\Delta t} + \mathcal{R}(w^{n+1}) = 0, \tag{5}$$

with

$$\frac{Dw^{n+1}}{\Delta t} = \frac{3\Delta w^n - \Delta w^{n-1}}{2\Delta t},$$

and $\Delta w^n = w^{n+1} - w^n$. Like all second-order schemes, the leading error term of the BW2 is of dispersive nature. BW schemes of orders higher than 2 exist (see, e.g. [12]) but unfortunately they are no longer A-stable and may lead to severe constraints on the time step for advection dominated problems and low-dissipative schemes, which makes them uninteresting for the applications targeted in this study.

Eq. (5) shows that such a fully implicit scheme leads to the solution of a nonlinear system of equations at each physical time step, which requires some iterative technique. Linearization of the residual leads to the following implicit scheme:

$$\frac{dF^n}{dw}\Delta w^n = -F^n, \tag{6}$$

with

$$\frac{dF^n}{dw} = \frac{3}{2\Delta t}I + \frac{d\mathcal{R}^n}{dw}.$$

Due to the use of an iterative technique to solve equation (6) instead of a direct linear solver, the resulting algorithm is categorized as an "inexact" Newton's method [26]. A simple inexact method results in the following convergence criterion on each linear iteration:

$$\|\frac{dF^n}{dw}\Delta w^n + F^n\| < \epsilon \|F^n\| \tag{7}$$

where the forcing term ϵ is a constant smaller than unity. For details on the forcing term see Reference [27].

In the following, we call implicit stage the inner-loop operator used to solve (6) at each Newton step, and explicit stage the scheme approximating the right hand side of the same equation. The efficiency of an implicit stage resolution depends on two main parameters: the amplification factor (which is defined hereafter) and the cost per-iteration.

Let H and K be the Fourier symbols of space discretization operators for the implicit stage and the explicit stage, respectively. We define the amplification matrix G as:

$$G = I - H^{-1}.K \tag{8}$$

and so the amplification factor f_G is given by its spectral radius $\lambda(G)$. In the scalar case, this reduces to:

$$f_G = \frac{H - K}{H}$$

If a total implicitation of the explicit stage is done, an ideal amplification factor is obtained. It reads: $f_{G*} = \frac{1}{1+K}$.

As we use high-order schemes for the evaluation of the explicit part, the full implicitation could not be performed without a considerable computational cost per iteration. This difficulty is circumvented by applying a defect correction approach, in which a first-order operator is used to approximate the Jacobian matrix. This kind of approach has been applied to DNS/LES [1,4,5]. Here, we apply the method proposed in Refs. [28,29] for the resolution of stationary problems. The resulting implicit stage is matrix-free. The number of sub-iterations required to access a reasonable convergence rate depends on the space discretization scheme used for the evaluation of the right hand side of (6). Numerical experiments show that, for the spatial schemes in use in this study about 15 sub-iterations are required to ensure a good convergence of the inner iteration loop.

3.3. Implicit residual smoothing

3.3.1. Second-order implicit residual smoothing

The stability domain of the explicit RK schemes can be enlarged by using an implicit residual smoothing (IRS) technique. The smoothing operator may include the contribution of viscous fluxes [12,30–32]. However, this would introduce an additional error of the order of Δt, which makes it unsuitable for unsteady computations. For this reason, in the following we restrict our attention to IRS operators involving the contribution of inviscid terms only. This is also justified by the fact that in many situations, the inviscid restriction on the time step is the most constraining one.

Following Ref. [30], a Lax–Wendroff-like implicit operator is applied at each Runge–Kutta stage as

$$
\begin{cases}
w^{(0)} = w^n \\
\mathcal{J}\Delta w^{(k)} = -a_k \Delta t \mathcal{R}(w^{(k-1)}), \quad k = 1, \dots, s \\
w^{n+1} = w^{(s)}
\end{cases}
\tag{9}
$$

For a one-dimensional problem, the implicit operator reads:

$$
\mathcal{J} = 1 - \theta \left(\frac{\Delta t}{\delta x} \right)^2 \delta(\lambda^{e2}\delta)
\tag{10}
$$

where θ is a tuning parameter. For a pure advection problem, the scheme can be made unconditionally stable by choosing [12,24,32]:

$$
\theta \geq \frac{1}{16}
$$

For multidimensional problems, the implicit operator is obtained by factorization of 1D operators in each mesh direction. Precisely, for a d-dimensional problem the IRS operator is of the form

$$
\mathcal{J} = \prod_{l=1}^{d} \left[1 - \theta (\frac{\Delta t}{\delta x_l})^2 \delta_l(\lambda_l^{e2}\delta_l) \right]
\tag{11}
$$

with δx_l, δ_l and λ_l^e are, respectively, the space step, the difference operator and the spectral radius of the flux jacobian in the lth direction. The multidimensional IRS operator leads to the inversion of a tridiagonal system per mesh direction at each Runge–Kutta stage. If required, the IRS operator can be applied only in space directions leading to a stiff behavior (e.g. the direction normal to a solid wall), thus avoiding the inversion of systems in the other directions. This strategy is applied for the DNS of turbulent channel flows presented in the following. The implicit smoothing terms are easily seen to be consistent with a Laplacian operator applied to the residual or, alternatively, to the time derivative. For the governing equation (1) the additional error introduced by the IRS operator with respect to the explicit scheme is of the form:

$$
-\theta \Delta t^2 \sum_d (\lambda_d^e)^2 \frac{\partial f_d^e}{\partial x_d^3} + O(\Delta t^2)
\tag{12}
$$

Being proportional to a third derivative of the flux f_d^e, this error is recognized to have a dispersive nature. It becomes larger and larger as the smoothing coefficient is increased to stabilize the baseline scheme. In the following, the preceding second-order IRS scheme is referred to as IRS2.

3.3.2. Fourth-order implicit residual smoothing

The main idea of IRS techniques is to run the explicit scheme with a time step greater than the stability limit of the scheme. The scheme is then stabilized by smoothing the residual by means of a Laplacian filter after each RK stage. To increase the accuracy of the IRS scheme, we investigate in the following the possibility of replacing the Laplacian filter by a bilaplacian operator, approximated by fourth order central differences. With this choice, the 1D IRS operator (referred to as IRS4 in the following) written in conservative form becomes:

$$
\mathcal{J} = 1 + \theta \left(\frac{\Delta t}{\delta x} \right)^4 \delta(\lambda^{e4}\delta^3)
\tag{13}
$$

The numerical error introduced by the IRS4 operator is now of the form:

$$
-\frac{1}{12}\theta \Delta t^4 \sum_d (\lambda_d^e)^4 \frac{\partial f_d^e}{\partial x_d^5} + O(\Delta t^4)
\tag{14}
$$

Like for the IRS2 scheme, it is of dissipative nature, but now it is of the same order or higher than the baseline explicit time integration scheme. For small enough values of the smoothing coefficient θ it is then expected that this additional error has a negligible impact on the accuracy of the baseline scheme.

The IRS4 scheme involves fourth difference operators. As a consequence, it leads to the inversion of a scalar pentadiagonal system per RK stage for 1D problems, or to the inversion of a pentadiagonal system per mesh direction for multi-D problems.

3.3.3. Matrix systems for IRS schemes

The IRS schemes at kth Runge–Kutta step (9) in matrix-vector form write:

$$\mathcal{J}\Delta w^{(k)} = -a_k \Delta t \mathcal{R}(w^{(k-1)}) \tag{15}$$

When Dirichlet boundaries are applied at the physical boundaries of the computational domain, the solution increment inside the boundary is set to 0. In this case, the IRS operator leads to the inversion of the following $n \times n$ matrix:

$$
\mathbf{J}_{n\times n} =
\begin{pmatrix}
\gamma & \alpha & \beta & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
\alpha & \gamma & \alpha & \beta & 0 & 0 & 0 & 0 & \cdots & 0 \\
\beta & \alpha & \gamma & \alpha & \beta & 0 & 0 & 0 & \cdots & 0 \\
0 & \beta & \alpha & \gamma & \alpha & \beta & 0 & 0 & \cdots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & \beta & \alpha & \gamma & \alpha & \beta & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & 0 & 0 & \beta & \alpha & \gamma & \alpha & \beta \\
0 & \cdots & 0 & 0 & 0 & 0 & \beta & \alpha & \gamma & \alpha \\
0 & \cdots & 0 & 0 & 0 & 0 & 0 & \beta & \alpha & \gamma
\end{pmatrix}
$$

with $\beta = 0$, $\alpha = -\theta CFL^2$ and $\gamma = 1 - 2\alpha$ for the IRS2, and $\beta = \theta CFL^4$, $\alpha = -4\beta$ and $\gamma = 1 + 6\beta$ for the IRS4.

For other kind of physical (*i.e.* Neumann, mixed or periodic) and at inter-subdomain boundaries, one or two layers of ghost cells (depending on the IRS order) are used. This results in the inversion of a $(n + 2g) \times (n + 2g)$ matrix on each subdomain ($g = 1, 2$ being the number of ghost cells used for expressing the IRS operator):

$$
\mathbf{J}_{(n+2g)\times(n+2g)} =
\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
\beta & \alpha & \gamma & \alpha & \beta & 0 & 0 & 0 & \cdots & 0 \\
0 & \beta & \alpha & \gamma & \alpha & \beta & 0 & 0 & \cdots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & \beta & \alpha & \gamma & \alpha & \beta & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & 0 & 0 & \beta & \alpha & \gamma & \alpha & \beta \\
0 & \cdots & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & \cdots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}
$$

Information in the layers of ghost cells is also used to compute the right-hand side of Eq. (15). For Neumann or mixed boundaries where $\frac{\partial(\Delta w^{(k)})}{\partial x_i}$ must be set to 0, the right-hand side term is extrapolated from the first inner cell, e.g. (for a left boundary like the one in Fig. 1):

$$\mathcal{R}(w^{(k-1)})_{1-j} = \mathcal{R}(w^{(k-1)})_1 \quad \text{for } j = \{1, g\} \tag{16}$$

For inter-subdomain boundaries, ghost-cell information is transferred from the adjacent subdomain. This strategy is used for the parallel implementation of the method, leading to satisfactory results for a variety of DNS simulations, like the ones presented in sections 4.2 and 4.3. Nevertheless it involves MPI exchanges between adjacent subdomains, which implies a communication overcost. This remains however rather small in our computations, for which the minimal cell number is of the order of 50^3 in each block. The above-mentioned strategy may also deteriorate the overall stability at subdomain boundaries if very high CFL numbers are used. In the present simulations, based on maximal CFL numbers below 10, no stability problems were observed due to the boundary treatment. Finally, the efficiency and accuracy of the method at subdomain-boundaries could be further improved in the future, e.g. by using the approach described in [13].

3.3.4. Effect of IRS on stability

Consider the kth RK step:

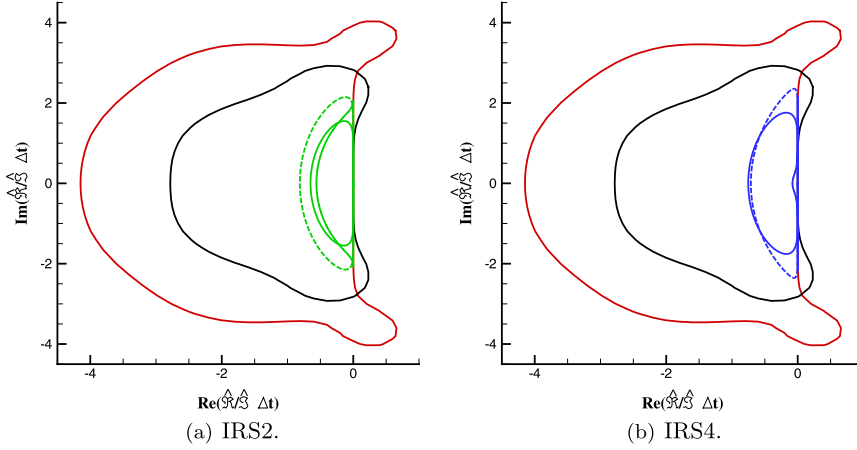$$w^{(k)} - w^{(0)} = -a_k \Delta t \mathcal{R}(w^{(k-1)}) \tag{17}$$

Fig. 2. Stability regions of the RK4 (black line) and RK6 (red line) schemes and locus of the FE9 spatial operator with implicit residual smoothing for CFL numbers 1 (solid line), 2 (dashed line) and 5 (dash-dotted line). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Considering a linear problem, taking the Fourier transform of the preceding expression, and introducing the amplification factor of the RK scheme at the kth step $G^{(k)}$, we get:

$$G^{(k)} = 1 - \Delta t \widehat{\mathcal{R}} G^{(k-1)}$$

where $G^{(0)} = 1$ and the complex $\widehat{\mathcal{R}}$ is the Fourier symbol of the spatial operator.

Suppose now that an IRS operator is applied to the left-hand side of (17). In this case, the amplification factor becomes:

$$G^{(k)} = 1 - \Delta t \frac{\widehat{\mathcal{R}}}{\widehat{\mathcal{J}}} G^{(k-1)}$$

where $\widehat{\mathcal{J}}$ is the Fourier symbol of the scalar IRS operator. For the IRS schemes introduced above:

$$\widehat{\mathcal{J}}_{IRS2} = 1 + 4\theta CFL^2 sin^2 \frac{\xi}{2}$$

$$\widehat{\mathcal{J}}_{IRS4} = 1 + 16\theta CFL^4 sin^4 \frac{\xi}{2}$$

where ξ is the reduced wave number. These are real numbers and always greater than or equal to 1. Then, the effect of the implicit operator is to reduce the Fourier symbol of the spatial operator by a quantity that is greater than one, which increases the maximum allowable time step.

To fix ideas, we study hereafter the stability of the preceding RK schemes, with and without the application of IRS, in conjunction with the FE9 spatial discretization scheme. Similar results are found for other schemes, including the DRP11 scheme [17].

Assuming a linear PDE, we take the Fourier transform of the semi-discrete equation (2):

$$\hat{w}_t = \widehat{\mathcal{R}} \hat{w}$$

where, for the FE9 scheme with $\kappa_2 = 0$ and $\kappa_{10} = \frac{1}{1260}$:

$$\Delta t \widehat{\mathcal{R}} = CFL \left\{ i \sin\frac{\xi}{2} \cos\frac{\xi}{2} \left[1 + \frac{4}{3} sin^3 \frac{\xi}{2} + \frac{16}{15} sin^5 \frac{\xi}{2} + \frac{32}{35} sin^7 \frac{\xi}{2} + \frac{256}{315} sin^9 \frac{\xi}{2} \right] - \frac{256}{315} sin^{10} \frac{\xi}{2} \right\}$$

For IRS schemes, the preceding expression for $\Delta\widehat{\mathcal{R}}$ has to be divided by $\widehat{\mathcal{J}}$, as discussed above. The fully discrete scheme is stable if the locus of $\Delta\widehat{\mathcal{R}}/\widehat{\mathcal{J}}$ in the complex plane fits within the stability region of the time integration scheme, i.e. the set of complex numbers G such that $|G| \leq 1$. Fig. 2 shows the stability regions of the RK4 and RK6 schemes in the complex plane ($\Re e(\Delta t\widehat{\mathcal{R}}/\widehat{\mathcal{J}})$, $\Im m(\Delta t\widehat{\mathcal{R}}/\widehat{\mathcal{J}})$), along with the locus of the spatial operator FE9 supplemented by IRS2 or IRS4, for CFL equal to 1, 2 and 10.

For IRS2, the smoothing coefficient is taken equal to the minimum value for unconditional stability, $\theta_{IRS2} = 1/16$. For IRS4, an analytical study of the optimum value of θ for unconditional stability is difficult, but a numerical search shows that unconditional stability is obtained for $\theta \gtrsim 0.005$. In the following, we choose $\theta_{IRS4} = 0.005$. As the CFL number is increased, the locus of the spatial operator is shrinked toward the origin, so that it always fits within the stability region of the time discretization scheme.
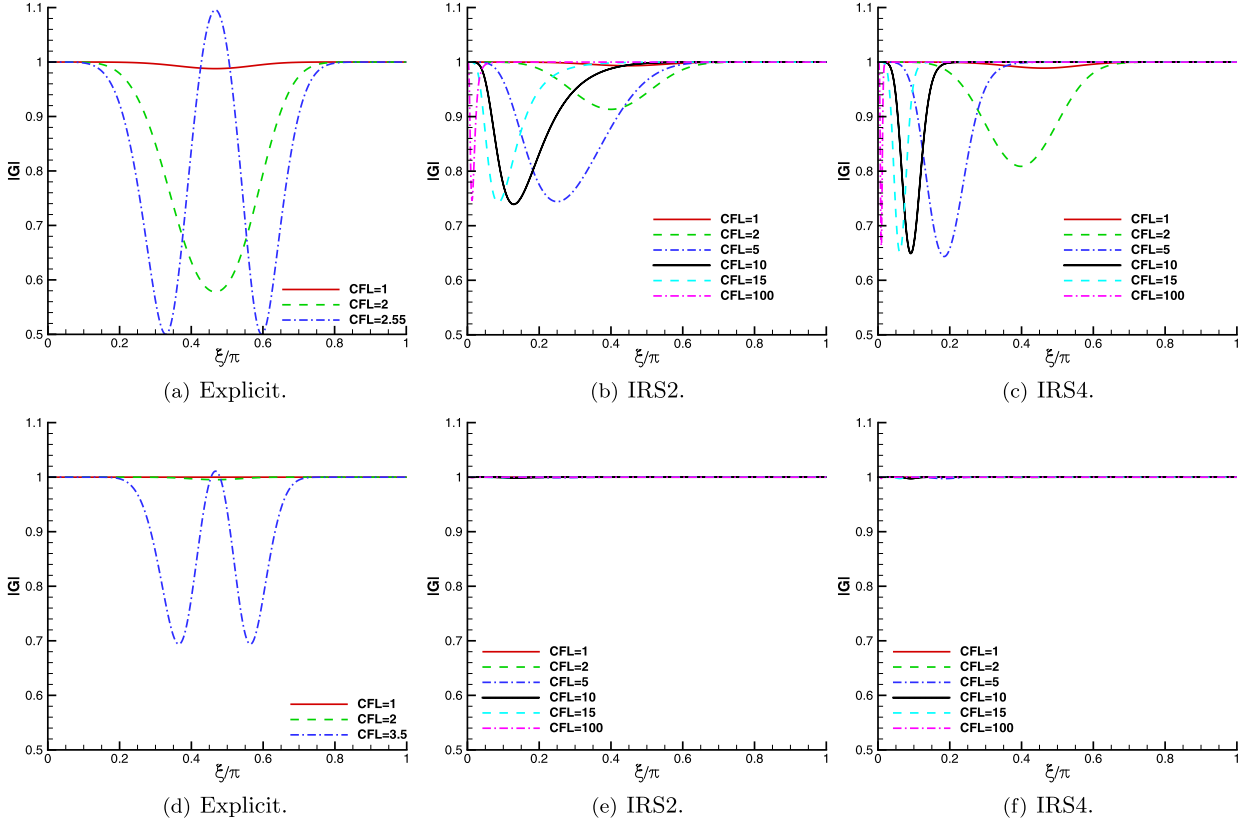
Fig. 3. Amplification factors versus the reduced wave number for different values of CFL. Top: RK4, bottom, RK6.

The amplification factors at the last RK step for the RK4 and RK6 schemes coupled the FE9 spatial scheme are represented in Fig. 3 as a function of the reduced wave number. For the explicit schemes, $|G| > 1$ for some ranges of wavenumbers as soon as CFL becomes greater than about 2–2.5. For IRS, $|G|$ is always below 1 provided that the smoothing parameter is selected according to the criteria enunciated above. It can be noticed that $|G|$ becomes closer and closer to 1 when applying IRS with high values of the CFL number, which means that the overall dissipation of the scheme is reduced with respect to the explicit scheme. However, the maximum dissipation is moved toward lower wave numbers. This is confirmed by inspection of the dissipation function of the modified spatial operator $\mathcal{J}^{-1}\mathcal{R}$, represented in Fig. 4a. The IRS operator reduces the numerical dissipation, and this effect increases with the CFL number and with the IRS order. For high CFL numbers, this could lead to undamped grid to grid oscillations, which should be avoided. The phase error associated to the same operator is represented in Fig. 4b. It can be seen that, as expected, that IRS increases the dispersion error, and its effect is more significant when higher values of the CFL are used. IRS4 reduces this effect with respect to IRS2 and, for a given CFL, it moves the cutoff toward higher wavenumbers. In any case, dispersion errors become high for CFL numbers above about 10. Nevertheless, for DNS or LES simulations, the CFL will typically not exceed such values to avoid temporal aliasing of turbulent structures. In the following numerical experiments, we restrict our analyses for IRS to CFL numbers up to 10.

The preceding analysis is valid for an infinite computational domain or a domain with periodic boundary conditions. Modifications of both the spatial and IRS operators close to solid boundaries affect both stability and numerical errors. For this reason, we carried out a stability analysis on a finite computational domain, using the boundary schemes described in sections 2 and 3.3.3, for the spatial and IRS operators, respectively. For this purpose we consider a one-dimensional linear scalar wave equation:

$$w_t + a w_x = 0 \tag{18}$$

Numerical approximation of the spatial derivatives as in section 2 supplemented by the introduction of an IRS matrix operator with Dirichlet boundary conditions (see section 3.3.3) leads to the following system of linear differential equations:

$$w_t = -\left(\mathbf{J}_{n \times n}^{-1} \mathbf{R}_{n \times n} \right) w \tag{19}$$

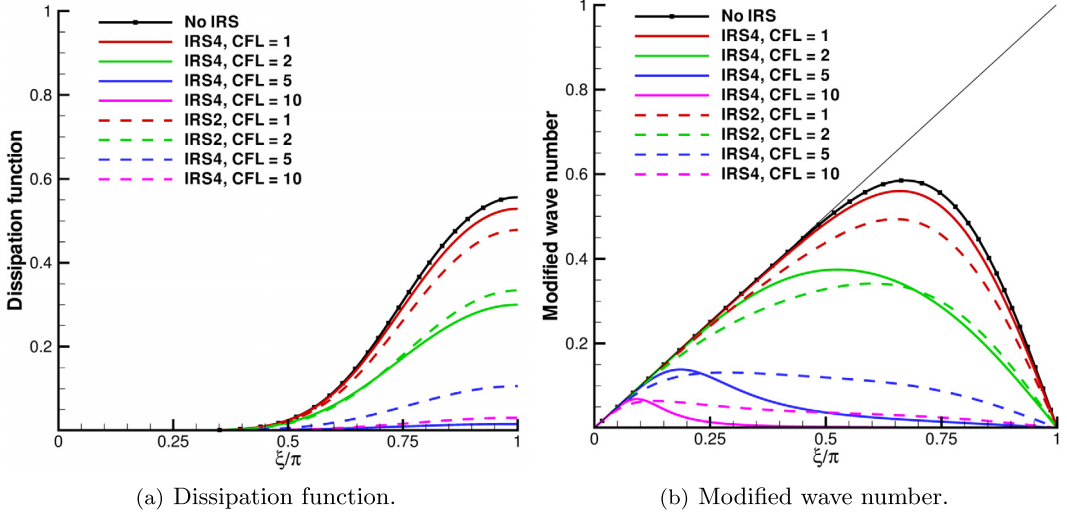(a) Dissipation function. (b) Modified wave number.

Fig. 4. Dissipation function and modified wave number for the modified spatial operator $\mathcal{J}^{-1}\mathcal{R}$ for different values of the CFL number.
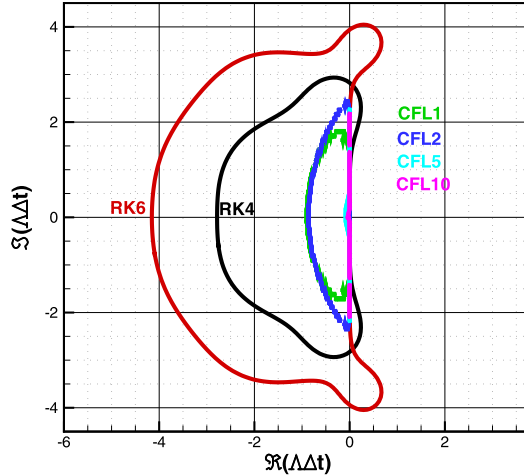

Fig. 5. Eigenvalues of the discrete spatial operator FE9–IRS4 including boundary schemes for $CFL = \{1, 2, 5, 10\}$.

where $\mathbf{R}_{n \times n}$ is the matrix associated to the operator $\mathcal{R}$. The solution of (19) consists of n normal modes $w = \widehat{w}e^{\Lambda_k t}$, with $\Lambda_k \in \mathbb{C}$ the amplification rates. Inserting the normal modes in the ODE leads to the following eigenvalue problem:

$$\left(\mathbf{J}_{n \times n}^{-1} \mathbf{R}_{n \times n} \right) \widehat{w} = -\Lambda \widehat{w} \tag{20}$$

The real part of the eigenvalues must be equal or less than zero to guarantee numerical stability of the spatial scheme. As it has been made previously, both sides of equation (20) are multiplied by Δt to make appear the CFL number in the expression of the discretization matrix. The eigenvalues are calculated numerically by using the numpy linear algebra package of the Python environment.

Fig. 5 shows the eigenvalues computed on a 200×200 grid in the complex plane $(\mathfrak{Re}(\Lambda \Delta t), \mathfrak{Im}(\Lambda \Delta t))$, for various CFL numbers for the FE9 spatial scheme with IRS4.

The real part of the eigenvalues is found to be negative for $CFL = \{1, 2, 5, 10\}$, thus satisfying the necessary condition for stability. We also observe that the eigenvalue distribution matches closely the locus of the spatial operator reported in Fig. 2b, corresponding to the fully periodic case. This proves that modifications of the scheme at the boundary have a small influence on the spectral properties of the spatial operator, especially along the imaginary axis (which is a measure of the dispersion errors). For coarser grids the boundary scheme has a greater influence on the eigenvalue distribution, and in particular it leads to more negative eigenvalues (see Fig. 6), which suggests a slightly more dissipative behavior.

In the following section we investigate numerically the impact of the additional error for some well-documented test cases.
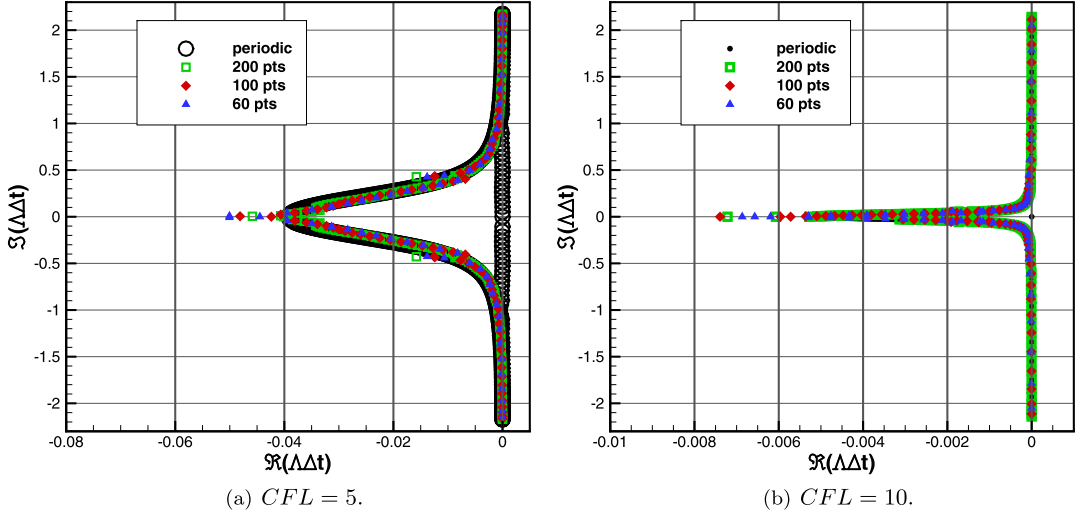
(a) $CFL = 5$.   (b) $CFL = 10$.

Fig. 6. Effect of the grid size on the eigenvalues of the FE9–IRS4 discretization matrix, for CFL 5 (a) and 10 (b).
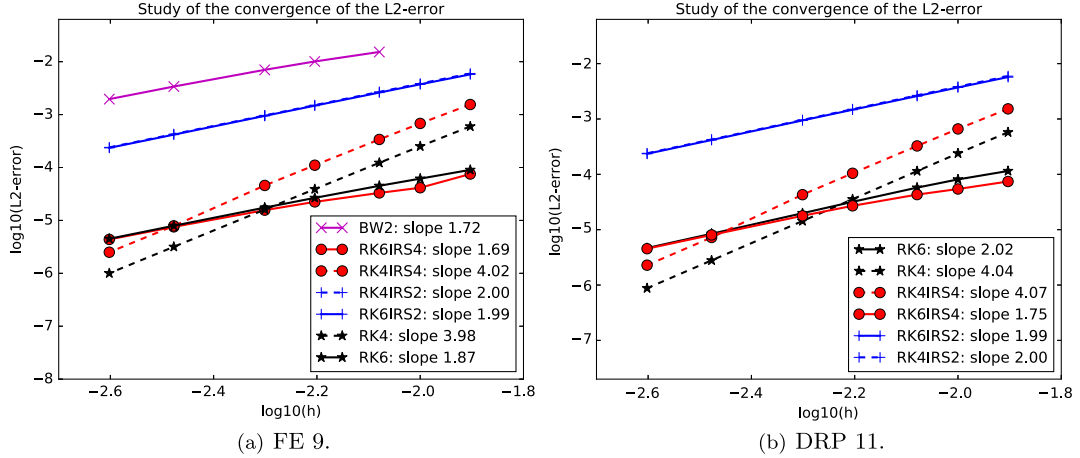

(a) FE 9.   (b) DRP 11.

Fig. 7. L_2 norm of the error with respect to the exact solution for the FE9 scheme (a) and the DRP11 scheme (b). $CFL = 1$.

4. Numerical results

4.1. Linear advection of a Gaussian hump

We consider the linear advection of a Gaussian hump along the diagonal of the periodic-square domain $[-1, 1]^2$. The exact solution is of the form:

$$w(x, y, t) = \exp\left(-100\, r(t)\right) \quad \text{with} \quad r(t) = \left(x - (x_0 + \frac{a}{\sqrt{2}}t)\right) + \left(y - (y_0 + \frac{a}{\sqrt{2}}t)\right) \tag{21}$$

The hump is initialized at $t = 0$ at the position $(x_0, y_0) = (-0.5, -0.5)$ and is advected along the diagonal at a velocity $a = \sqrt{2}$. The calculations are carried out using the implicit new high-order IRS and the IRS2 schemes along with RK time stepping, as well as the BW2 scheme with 10 quasi-exact Newton subiterations. For the lower values of the CFL number, the results are compared to those of the explicit RK schemes. First of all, we check the convergence order of the schemes under investigation by computing the errors with respect to the exact solution. The CFL is chosen equal to 1, to allow comparisons with the explicit schemes. Results are shown both for the FE9 and for the DRP11 schemes which use the same discretization stencil, to check the sensitivity of the spatial approximation. For this linear problem, RK4 is expected to be fourth-order accurate in time, whereas the optimized RK6 is second-order accurate. The variation of the L_2 norm of the error with the mesh size is represented in Fig. 7 for the two spatial schemes and the different time integration schemes under investigation. The expected formal orders of convergence are reasonably well recovered. The BW2 scheme provides by far the highest error levels. As expected, RK6 gives lower error levels than RK4 when using coarse time discretizations, even if its convergence order is lower for this linear case. This good behavior is conserved even when introducing IRS. As expected,

Table 1

Elapsed CPU time to reach $t = 1$ with all time integration schemes on a 100×100 mesh. FE9.

	$CFL = 1$	$CFL = 5$	$CFL = 10$
RK4	1.	x	x
RK6	1.40	x	x
RK4 IRS2	1.56	0.32	0.16
RK6 IRS2	1.83	0.38	0.19
RK4 IRS4	1.54	0.32	0.16
RK6 IRS4	1.88	0.39	0.20
BW2	4.21	1.17	0.84

the error levels of the RK schemes are increased when using IRS compared to the purely explicit scheme, mainly because of the additional dispersion errors. Compared to IRS2, the IRS4 reduces error levels greatly and is then identified as a good candidate for accurate implicit simulations. In particular, when combined to the second-order RK6 scheme, the fourth-order IRS leaves the error of the baseline scheme almost unchanged. The slightly lower errors of the RK6IRS scheme on coarser meshes could be due to the reduced dissipation of the IRS4 scheme compared to the baseline. Figs. 8 and 9 show the computed fields of w at the final time $t = 1$ for all time integration schemes in used in this study, at $CFL = 5$ and 10. The grid in use is made of 100×100 cells. The FE9 scheme is used for the spatial discretization. For $CFL = 5$, the BW2 scheme has a much too large dispersion error, which results in highly deformed iso-contours with respect to the exact solution. The IRS schemes give much more accurate results and their solutions remain close to the exact one. A small deformation of the contours is observed for the RK4 scheme, both with IRS2 and IRS4, and an even smaller one for the RK6IRS2 scheme. RK6IRS4 provides the most satisfactory solution, almost superposed to the exact one, thanks to the optimized dispersion error of the RK6 scheme and the accurate IRS treatment. Similar trends are observed also for $CFL = 10$, even if the errors are greater. Specifically, for this high CFL number, IRS2 leads to large errors. Errors are much more acceptable when using IRS4, especially in conjunction with the optimized RK6 scheme. Fig. 10 shows horizontal sections of the hump at the final time for different time schemes and FE9, along with close-ups of the solution at the top and at the bottom of the hump. The BW2 solution is considerably shifted to left and exhibits an undershoot at the hump foot, as well as a much lower peak, due to the large dissipation and dispersion errors. IRS2 behaves better, yet it introduces a phase shift with respect to the exact solution; it also leads to more pronounced undershoots than BW2, due to the significant dispersion error combined to a lower numerical dissipation. These undesirable effects are strongly reduced for the IRS4 schemes. The best results are obtained once again when using the RK6IRS4 scheme. Note that, in agreement with the theoretical results of the preceding section, using CFL numbers below 10 appears to be an essential condition to keep the numerical errors low in this kind of fast unsteady simulations. Table 1 provides the final computational times for different schemes and CFL numbers. These were obtained with FE9 but the observed behavior for DRP11 is similar. The values are normalized with respect to the time required by the RK4 scheme with $CFL = 1$. For $CFL = 1$ the RK4 is the cheapest scheme, whereas BW2 is by far the more expensive one, due to the inner subiteration of the quasi-exact Newton procedure. The RK6 scheme is about 40% more expensive than RK4 but it is also more accurate. Adding the IRS treatment increases the cost of the RK4 scheme by a bit more than 50%, both with IRS2 and IRS4. The similar overcost obtained for IRS2 which requires the inversion of tridiagonal systems, and for the pentadiagonal IRS4 operator proves the efficiency of the pentadiagonal solver. For the RK6 scheme, the overcost due to the IRS is lower (a bit more than 30%) since the computational cost of the baseline RK solver is higher. These comparisons are made using the same time step for all of the schemes. For higher values of the CFL number, the overall cost of the implicit schemes becomes lower than that of the explicit ones, due to the reduced number of time steps required to complete the simulation. The BW2 scheme is more than twice more expensive than the IRS schemes, and is less accurate. The RK6IRS4 scheme, which provides the more accurate results for the hump advection problem at $CFL = 5$, leads to an overall cost that is about 70% lower than the RK4 at $CFL = 1$, while keeping a similar accuracy. Even if the RK can be run at CFL greater than 1 (albeit with lower accuracy), the overall cost of RK6IRS4 remains lower. In the following, we restrict our attention to the BW2 scheme and to the RK6 scheme with different kinds of IRS.

4.1.1. Effect of the boundary treatment

To assess the effect of the boundary treatment of the IRS operators at periodic boundaries (see section 3.3.3), simulations with larger time integration at $CFL = 5$ are carried out on the 250×250 grid. Fig. 11 shows the numerical solutions at $t = 6$ (*i.e.* after three revolutions around the computational box) obtained with the BW2, RK6IRS2 and RK6IRS4 time integration schemes. The RK6IRS4 scheme gives results very close to the exact solution whereas the RK6IRS2 and BW2 solutions exhibit large errors: the former introduces significant oscillations due to dispersion errors, whereas the latter tends to dissipate the solution significantly.

The effect of the boundary treatment used for the IRS operator is shown in Fig. 12. In this figure, the periodic calculation is carried out by using the matrix system augmented with ghost cells at the periodic boundaries or by just truncating the matrix as for Dirichlet boundary conditions (denoted as noGH, no ghost cells, in the figure). Truncating the matrix at periodic boundaries increases the numerical errors significantly, and specifically introduces a delay in the solution, which is avoided by using the ghost-cell treatment.
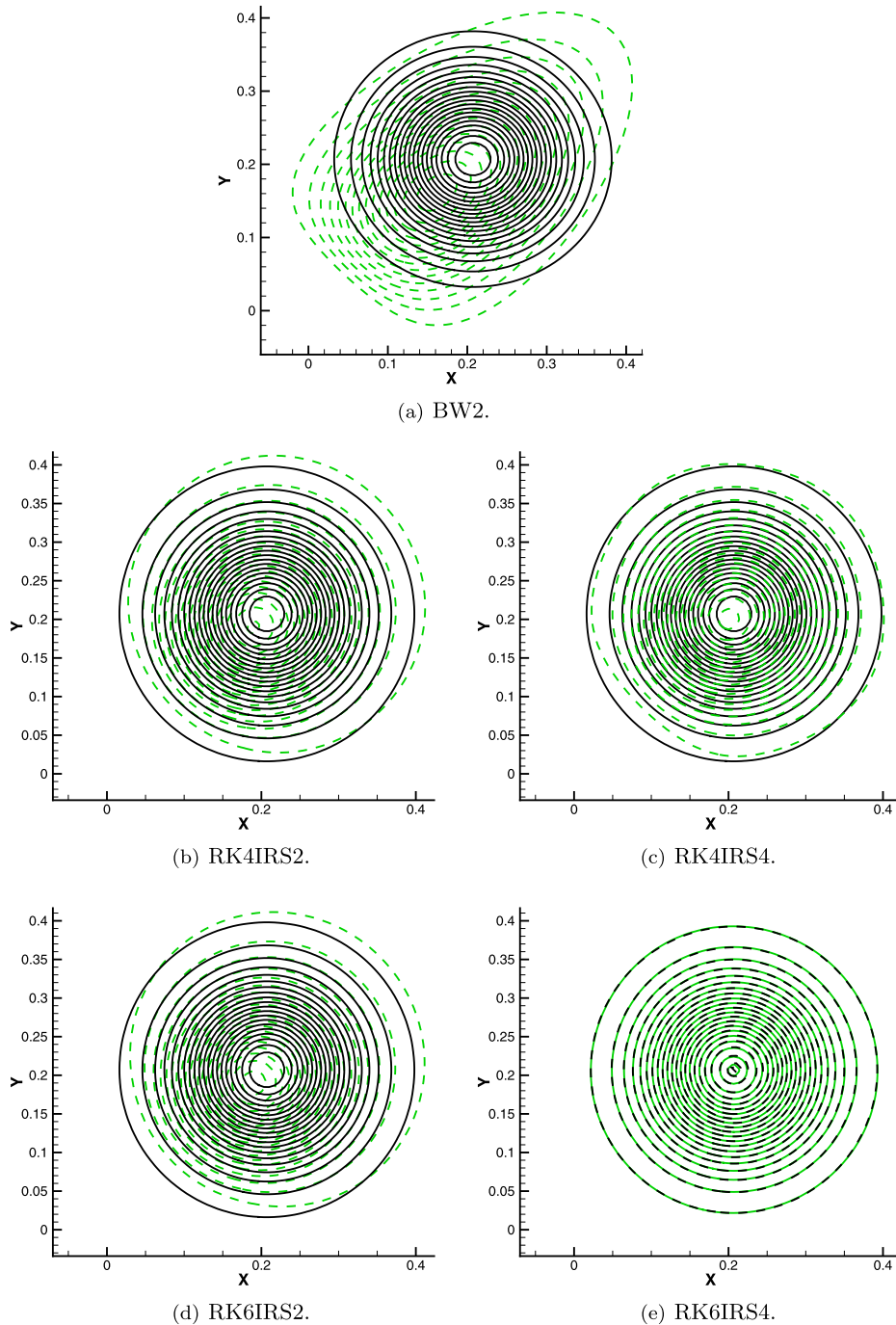
(a) BW2.

(b) RK4IRS2.

(c) RK4IRS4.

(d) RK6IRS2.

(e) RK6IRS4.

Fig. 8. Iso-values of the hump at $t = 1$. $CFL = 5$.

To check the effect of spatial boundary schemes in conjunction with IRS, numerical solutions are also computed with non-periodic boundary conditions. Specifically, Dirichlet boundary conditions are applied at the left and bottom boundaries of the domain, were the exact solution is imposed, and an outflow (von Neumann) condition based on fourth-order extrapolation from inner cells is applied at the right and top boundaries, as sketched in Fig. 13. Simulations of this test case are run up to $t = 1.8$. Iso-contours of the solution obtained with RK6IRS4 at $CFL = 1$ at uniformly distributed times in the range $0 < t < 1.5$ are also reported on this figure. The hump exits the computational domain without creation of spurious waves. The same conclusions hold for calculations at $CFL = 5$ presented in Fig. 13b.

(a) BW2.

(b) RK4IRS2.

(c) RK4IRS4.

(d) RK6IRS2.

(e) RK6IRS4.
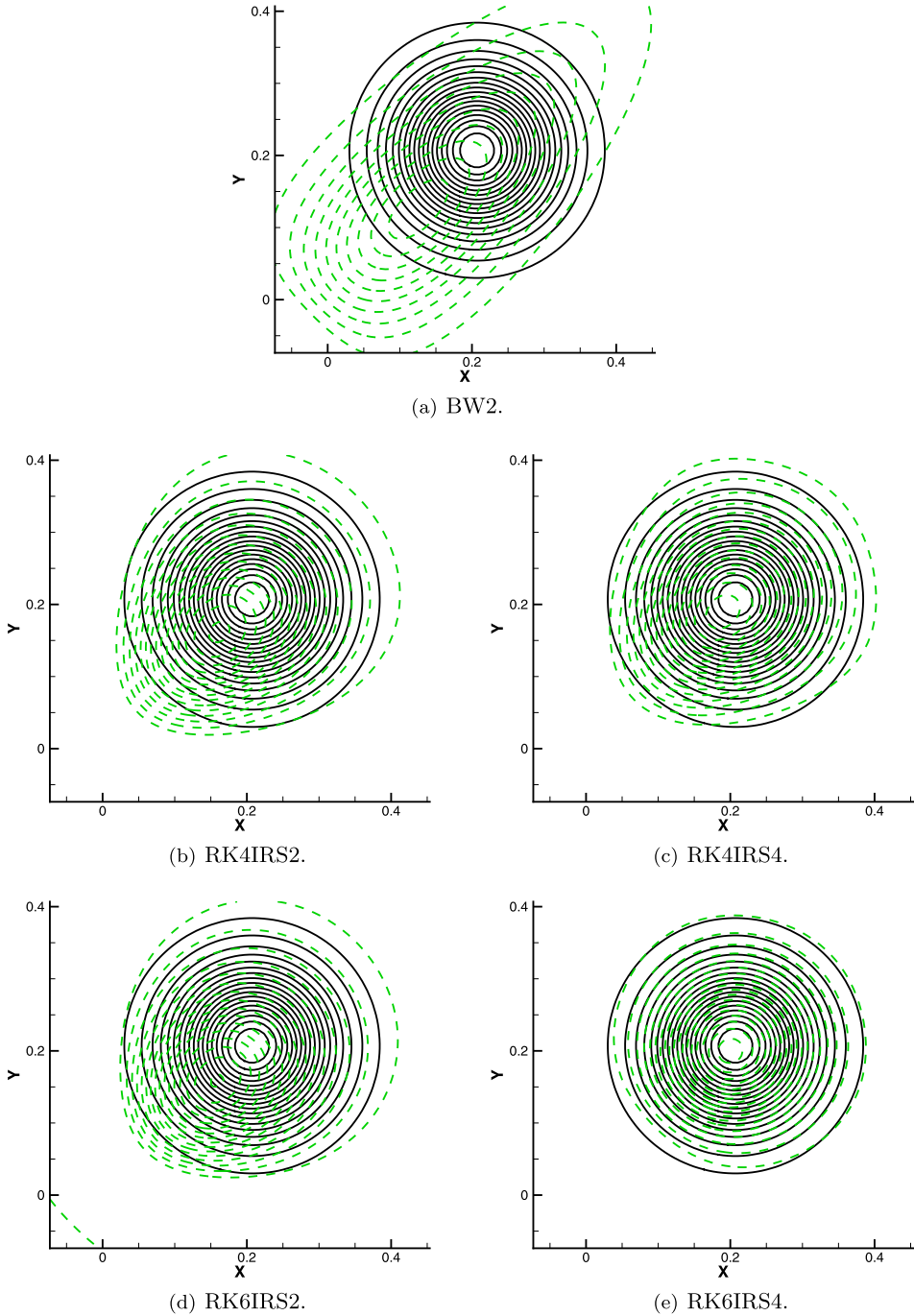
**Fig. 9.** Iso-values of the hump at $t = 1$. $CFL = 10$.

The L_2 norms of the errors for this configurations are reported in Fig. 14 for various time integration schemes and $CFL = 1$. Introduction of the boundary schemes does lead to slightly higher error levels with respect to the periodic case, but does not affect the convergence order of the solutions significantly.
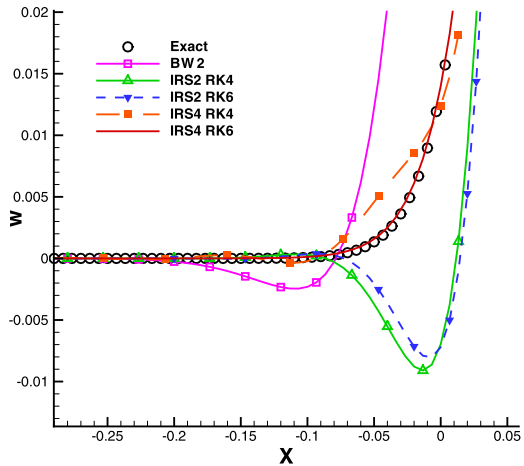
4.2. Homogeneous isotropic turbulence

The implicit schemes under investigation are applied to the numerical simulation of decaying homogeneous isotropic turbulence (HIT). Two regimes of HIT are considered in the following: a quasi-incompressible flow at initial turbulent Mach number $M_{t0} = 0.2$ and the compressible one at $M_{t0} = 1$.
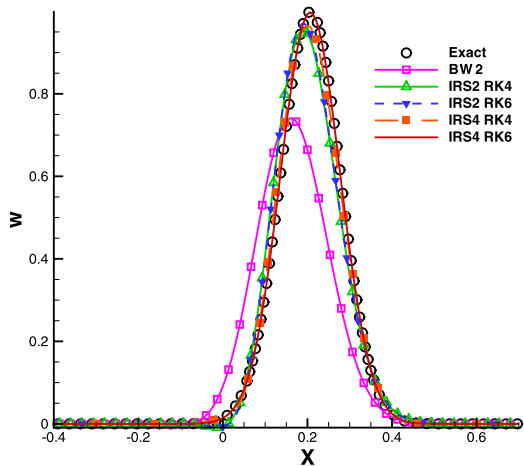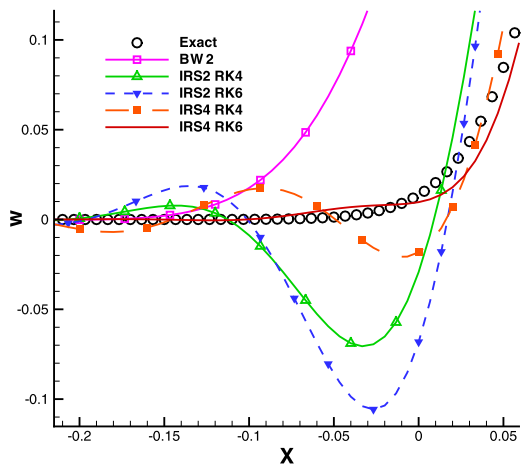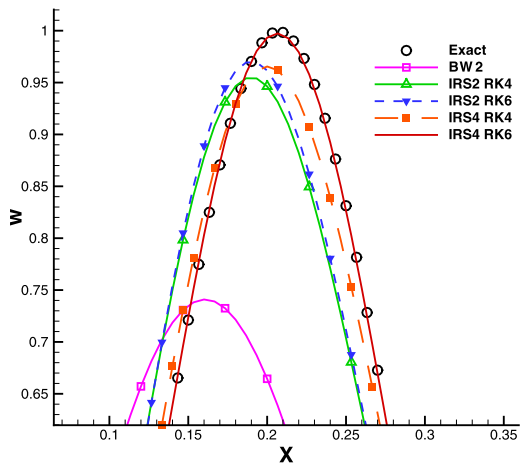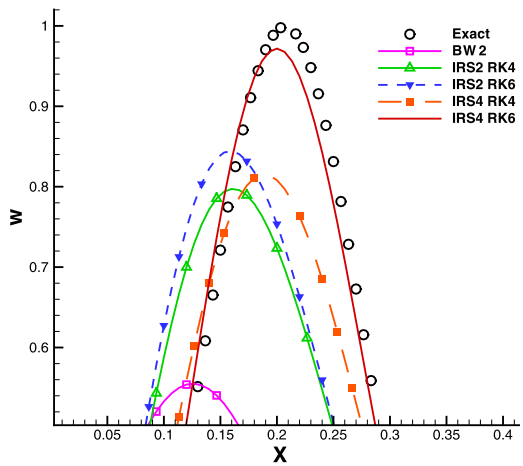
(a) $CFL = 5.$
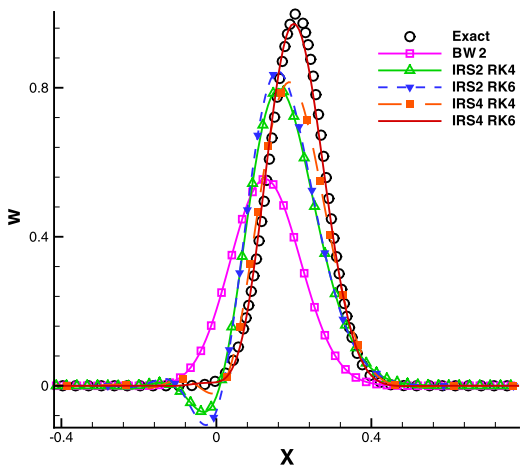
(b) $CFL = 10.$

(c) $CFL = 5.$ Close-up of the bottom part.

(d) $CFL = 10.$ Close-up of the bottom part.

(e) $CFL = 5.$ Close-up of the upper part.

(f) $CFL = 10.$ Close-up of the upper part.

Fig. 10. Plots of numerical solutions for w along the line $y(\max w)$ at time $t = 1$. $CFL = 5$ (left) and $CFL = 10$ (right).
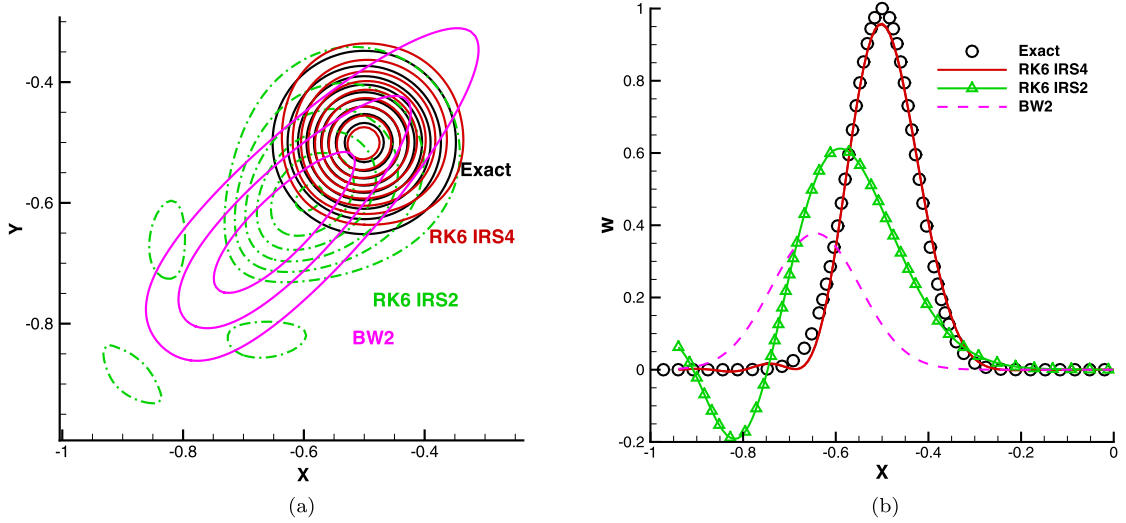
Fig. 11. (a) Iso-contours of the numerical solution at $t = 6$ for the advection of a hump using different time integration schemes ($CFL = 5$); (b) distribution of the solution along the line $y = y(\max w)$.
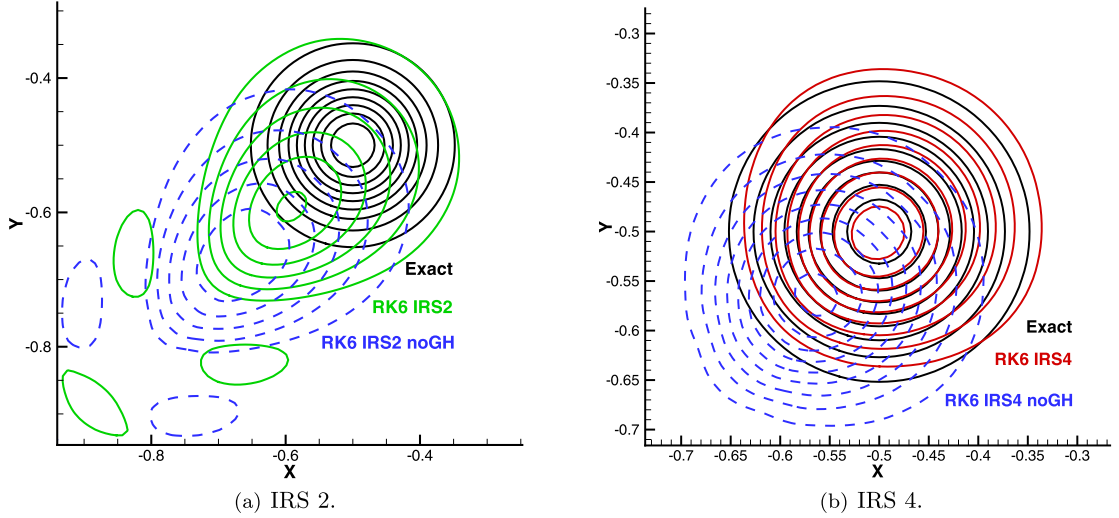

Fig. 12. Iso-contours of the numerical solution at $t = 6$ for the advection of a hump using different boundary treatments for the IRS operators (with and without ghost cells). $CFL = 5$.

In both cases, the turbulent field is initialized by using divergence-free initial velocity fluctuations and zero density fluctuations. The initial velocity spectrum is of the Passot–Pouquet-type, with peak wave number fixed to $k_0 = 4$. Random phases are used for the Fourier coefficients of the initial velocity field. Since the initialization is almost incompressible, for relatively high-Mach numbers it can be observed at the beginning of the simulation a transient phase, through which the compressible components of the turbulent structures increase and a physical state of fully developed turbulence is reached [33].

Simulations are carried out at Reynolds number $Re_\lambda = 50$ over a tri-periodic domain of side $L = 2\pi$, discretized by 128^3 regularly spaced Cartesian cells. Computations are run up to the final time is $t = 10\tau$, where τ is the large-eddy turnover time. The code has been validated against the test cases with divergence-free initial conditions discussed in [34] (results not shown for brevity).

The explicit RK6 scheme is used to generate a reference solution. As mentioned in [17], this scheme requires 4 points per time scale to ensure a phase error less than 4×10^{-4} whereas the classical RK4 scheme requires 9 points to obtain the same accuracy. A priori estimates of the maximal time step allowing to resolve Kolmogorov's time scale throughout the computation lead to a maximal initial CFL number $CFL_0 \approx 1.8$ for the HIT at $M_{t0} = 1$ and $CFL_0 \approx 5$ for the HIT at $M_{t0} = 0.2$ with the present spatial grid. For the high Mach case, the time step required to resolve all relevant time scales is then found to be of the same order of the maximum CFL_0 allowed by the explicit scheme, whereas for the quasi-incompressible case
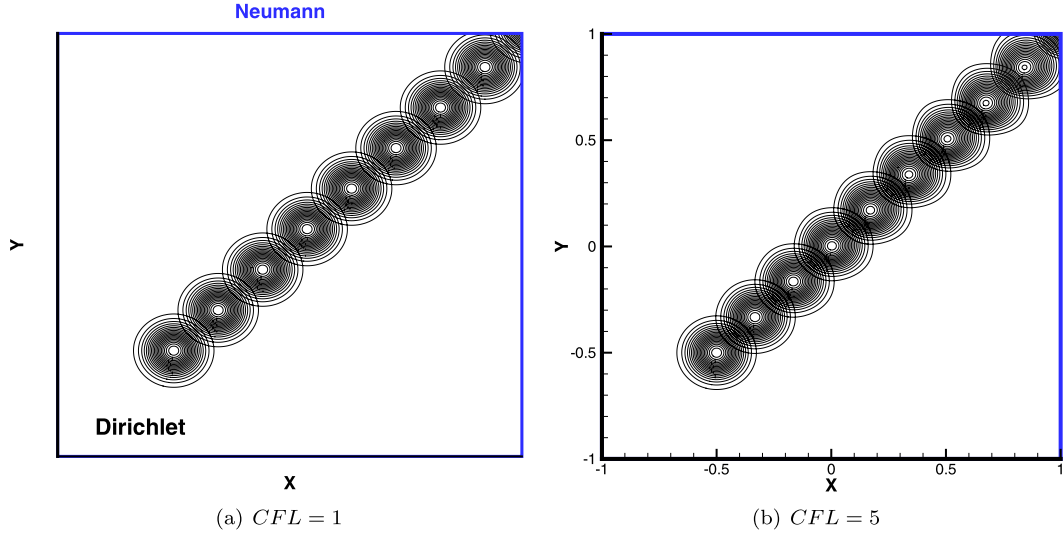
Fig. 13. Iso-contours of the numerical solution at various times up to $t = 1.8$ for the advection of a Gaussian hump with non-periodic boundary conditions. Dirichlet conditions are imposed on the left and bottom boundary (highlighted in black), von Neumann conditions at the top and right boundary (highlighted in blue). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
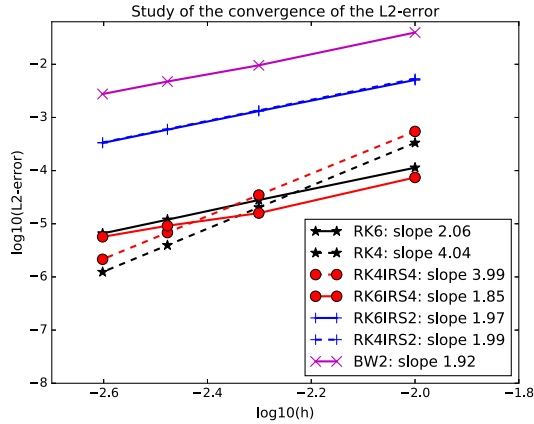

Fig. 14. Advection of a Gaussian hump with non-periodic boundary conditions and non-central boundary schemes. L_2 norm of the error with respect to the exact solution at $t = 1$ ($CFL = 1$).

it is bigger. Thus, for the high-Mach case, running the solution at large CFL number cannot be properly considered as DNS. Instead, it can be seen as a well-resolved implicit LES.

Afterwards, we compare the results obtained with several initial CFL numbers by using RK6IRS with those provided by the BW2 method. In this study, the convergence criterion for the inner iteration (γ from eq. (7)) of the BW2 scheme has been taken equal to $\gamma \approx 1.e{-}2$. With the spatial schemes used, the inversion of the inner-loop by the matrix-free algorithm requires about 15 subiterations.

4.2.1. Quasi-incompressible HIT at $M_{t0} = 0.2$

Fig. 15 provides the turbulent kinetic energy spectra computed with the FE9 scheme along with different implicit schemes and a time step corresponding to a CFL_0 equal to 10. The results obtained at simulation time $t = 10\tau$ show that the time integration scheme has virtually no impact on the computed spectra. Inspection of the time evolution of the enstrophy (Fig. 16), defined from the vorticity ω as $\Omega = \frac{1}{(2\pi)^3} \int \int \int_{[0,2\pi]^3} \frac{1}{2}\rho\|\omega\|^2 \, dxdydz$, shows a little under-prediction of the peak when the initial CFL is set to 10, whereas all of the results obtained with $CFL_0 = 5$ are superimposed to within plotting accuracy. The time evolution of the spatial root mean square of pressure fluctuations $p_{rms} = (\frac{1}{(2\pi)^3} \int \int \int_{[0,2\pi]^3} p'^2 \, dxdydz)^{\frac{1}{2}}$ (Fig. 17) shows that thermodynamic fields are more sensitive to the time step and numerical scheme in use. Specifically, when the time step is doubled, some information on the thermodynamic quantities is lost, the characteristic times scales of the thermodynamic fields being about one order of magnitude smaller than those of velocity fields. When the different implicit schemes are compared, it appears that the BW2 dramatically underestimates the solution with respect to RK6, es-
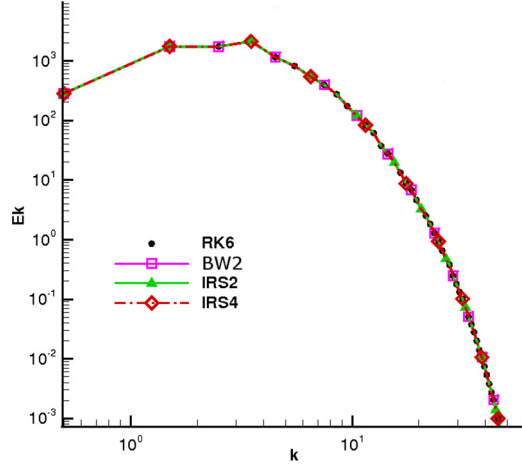
Fig. 15. HIT at $M_{t0} = 2$. Kinetic energy spectra at $t = 10\tau$ obtained with RK6 at $CFL_0 = 1.7$ and the RK6IRS2, RK6IRS4 and BW2 schemes at $CFL_0 = 10$.
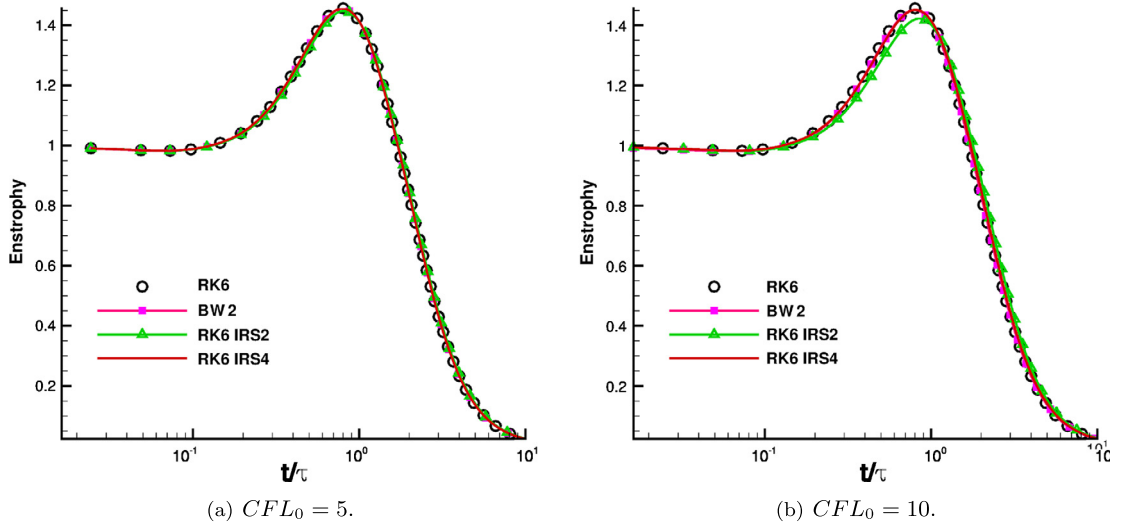

(a) $CFL_0 = 5$.

(b) $CFL_0 = 10$.

Fig. 16. HIT at $M_{t0} = 2$. Comparison of the time evolution of the enstrophy obtained with RK6 at $CFL_0 = 1.7$ and the RK6IRS2, RK6IRS4 and BW2 schemes with $CFL_0 = 5$ and 10.

pecially for simulations at $CFL_0 = 10$. The RK6IRS2 scheme exhibits a phase lag compared to RK6: this is very small for the lower time step and becomes more evident but still acceptable for $CFL_0 = 10$. The IRS4 solution is superposed to within plotting accuracy to the RK6 one for simulations with $CFL_0 = 5$ and exhibits a slight phase lag for computations with the higher value of CFL. Finally, we consider the time evolutions of higher order moments of the velocity gradient distribution, namely, the skewness and the flatness of the fluctuating velocity gradient (see Figs. 18 and 19), to check the effect of the implicit treatments on fine turbulence features. Let recall that the skewness factor S_a and the flatness factor F_a of the random field a are defined as:

$$S_a = \frac{\langle a^3 \rangle}{\langle a^2 \rangle^{3/2}}, F_a = \frac{\langle a^4 \rangle}{\langle a^2 \rangle^2} \tag{22}$$

All of the schemes perform quite well, but some large errors are observed for RK6IRS2 and $CFL_0 = 10$. A close up of the solution shows that, once again, the RK6IRS4 solution is almost identical to the RK6 one for $CFL_0 = 5$, and it exhibits some phase lag for higher CFL. Similar results are obtained for the flatness.

4.2.2. Compressible HIT at $M_{t0} = 1$

As for the quasi-incompressible case, the time integration scheme does not modify the prediction of the evolution of the enstrophy at $CFL_0 = 5$ (Fig. 20). At $CFL_0 = 10$, in the growth part of the enstrophy and in the vicinity of the peak (before $t/\tau = 0.8$), the BW2 simulation slightly under-predicts the enstrophy. The evolution of the pressure root mean square (Fig. 21) is less sensitive to the time integration scheme than in the quasi-incompressible case. On the contrary,
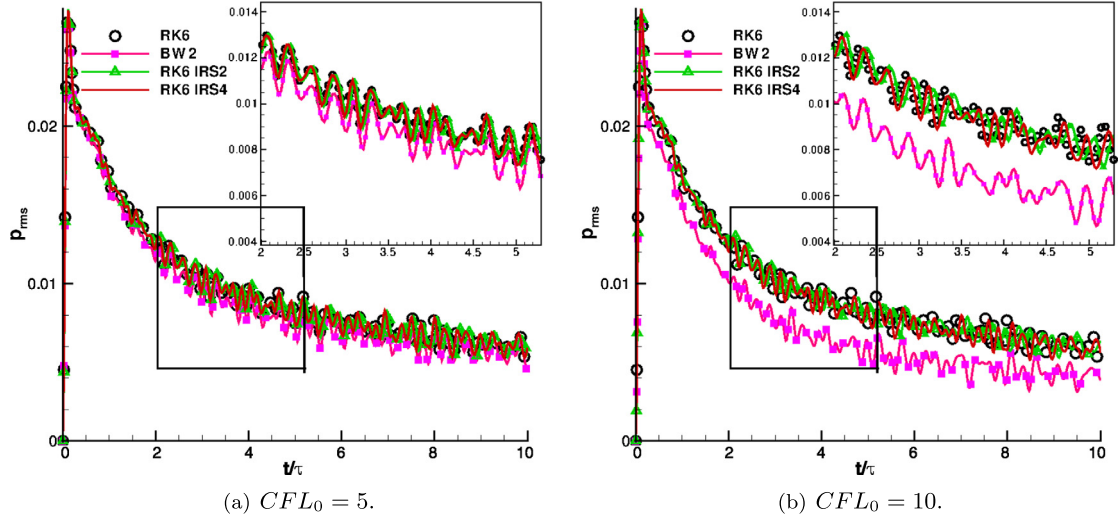
Fig. 17. HIT at $M_{t0} = 2$. Comparison of the time evolution of the p_{rms} obtained with RK6 at $CFL_0 = 1.7$ and the RK6IRS2, RK6IRS4, and BW2 schemes with $CFL_0 = 5$ and 10.
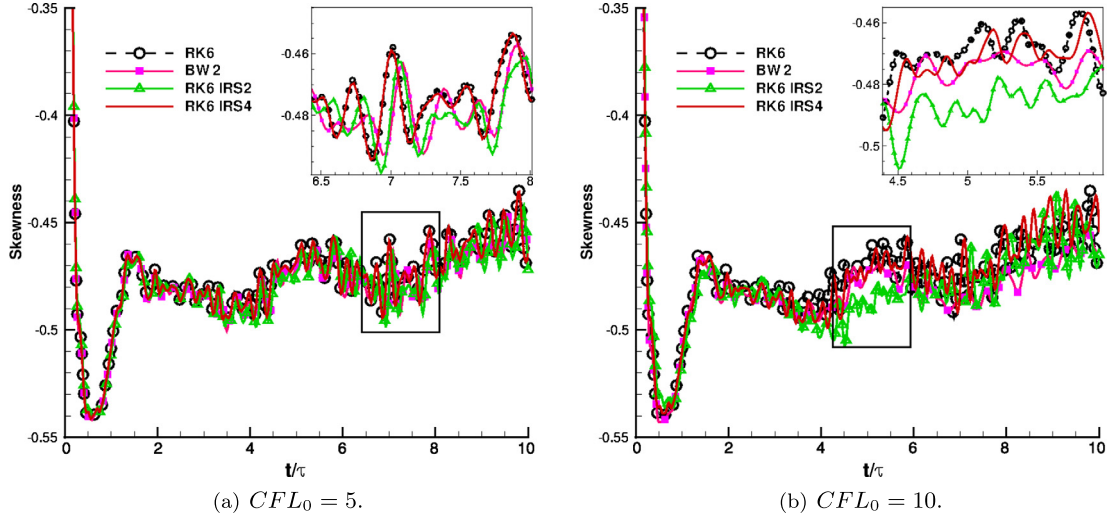

Fig. 18. HIT at $M_{t0} = 2$. Comparison of the time evolution of the skewness factor of the fluctuating velocity gradient $S_{\nabla u'}$ obtained with RK6 at $CFL_0 = 1.7$ and the RK6IRS2, RK6IRS4 and BW2 schemes at $CFL_0 = 5$ and 10.

the prediction of the evolution of the skewness (Fig. 22) and the flatness (Fig. 23) of the velocity fluctuations are more affected, namely for $t/\tau < 4$. At $CFL_0 = 5$, the BW2 solution and, in a minor way, the RK6IRS2 differ from the reference solution which indicates that fine turbulence features are not completely resolved. Deviations between the implicit and explicit solutions become more evident at $CFL_0 = 10$. Nevertheless, the error remains small, specially for RK6IRS4 scheme.

4.2.3. Computational cost

Tables 2 and 3 provide the computational times of the different simulations, for the quasi-incompressible and compressible HIT, respectively. Due to the large number of inner iterations required, the BW2 considerably increase the computational cost with respect to the reference RK6 simulation even when the time step is increased by 5.9. Moreover, the solution is less accurate. For this reason, this scheme is no longer used in the following. On the contrary, IRS schemes introduce only a moderate overcost with respect to the explicit scheme, while allowing greater time steps. In particular, the RK6IRS4 scheme at $CFL_0 = 5$ provides a solution very close to the reference for an overall computational time almost three times smaller for both HIT cases.
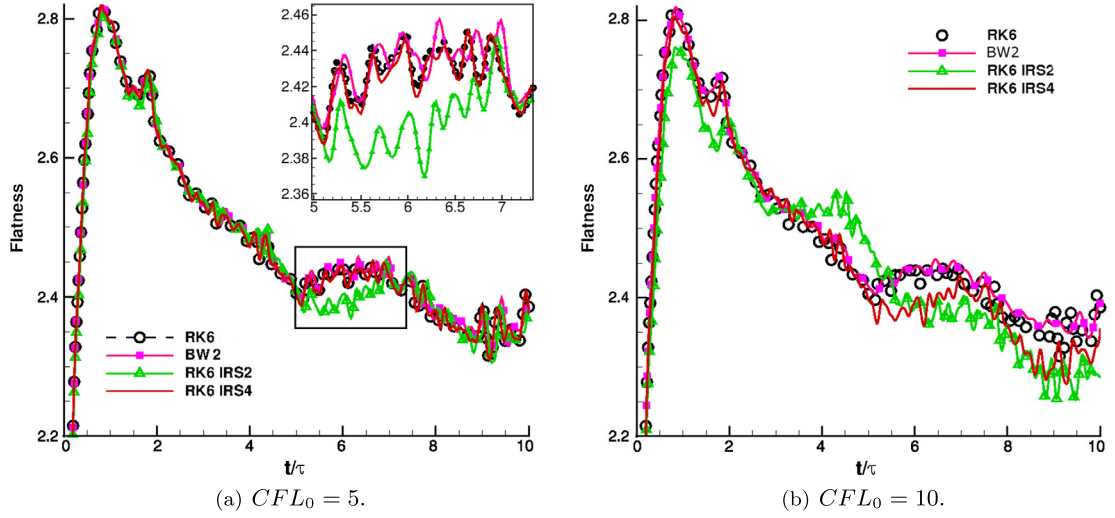
Fig. 19. HIT at $M_{t0} = 2$. Comparison of the time evolution of the flatness factor of the fluctuating velocity gradient $F_{\nabla u'}$ obtained with RK6 at $CFL_0 = 1.7$ and the RK6IRS2, and RK6IRS4 and BW2 schemes at $CFL_0 = 5$ and 10.
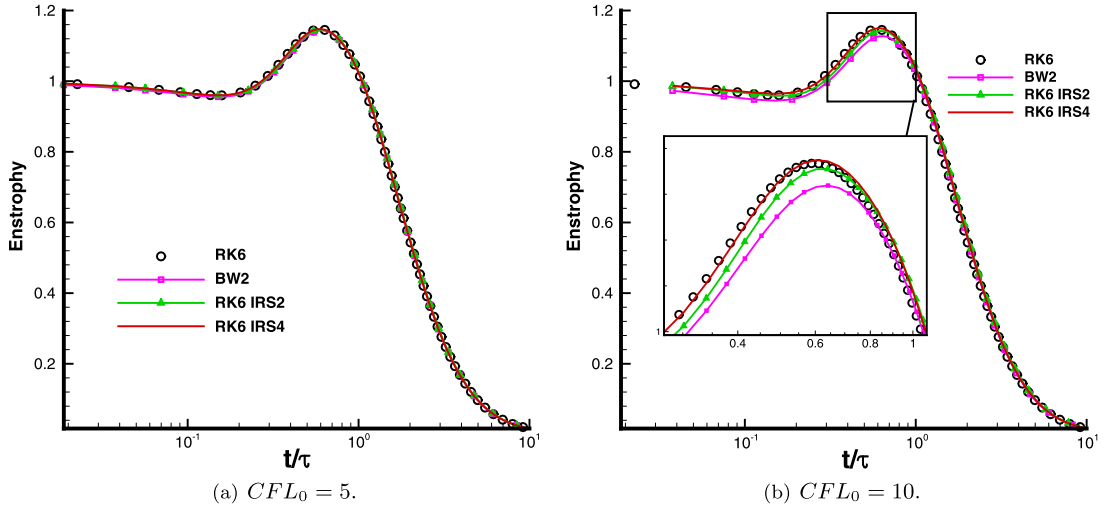

Fig. 20. HIT at $M_{t0} = 1$. Comparison of the time evolution of the enstrophy obtained with RKIRS and BW2 schemes with $CFL_0 = 5$ and 10.

Table 2
HIT at $M_{t0} = 0.2$. Time elapsed to reach $t = 10\tau$ with different time integration schemes and initial CFL numbers, on a 128^3 mesh.

	$CFL_0 = 1.7$	$CFL_0 = 5$	$CFL_0 = 10$
RK6	1	x	x
RK6IRS2	x	0.35	0.19
RK6IRS4	x	0.36	0.19
BW2	x	3.06	1.58

Table 3
HIT at $M_{t0} = 1$. Time elapsed to reach $t = 10\tau$ with different time integration schemes and initial CFL numbers, on a 128^3 mesh.

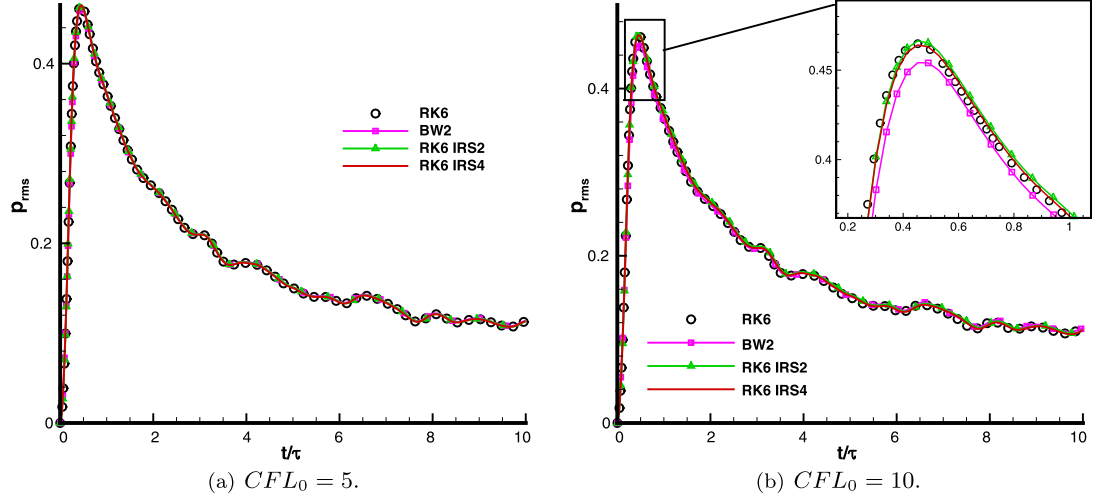|	$CFL_0 = 1.7$	$CFL_0 = 5$	$CFL_0 = 10$
RK6	1	x	x
RK6IRS2	x	0.38	0.19
RK6IRS4	x	0.38	0.19
BW2	x	2.87	1.44

Fig. 21. HIT at $M_{t0} = 1$. Comparison of the time evolution of the p_{rms} obtained with RK6 at $CFL_0 = 1.7$, RK6IRS2, RK6IRS4, and BW2 schemes with $CFL_0 = 5$ and 10.
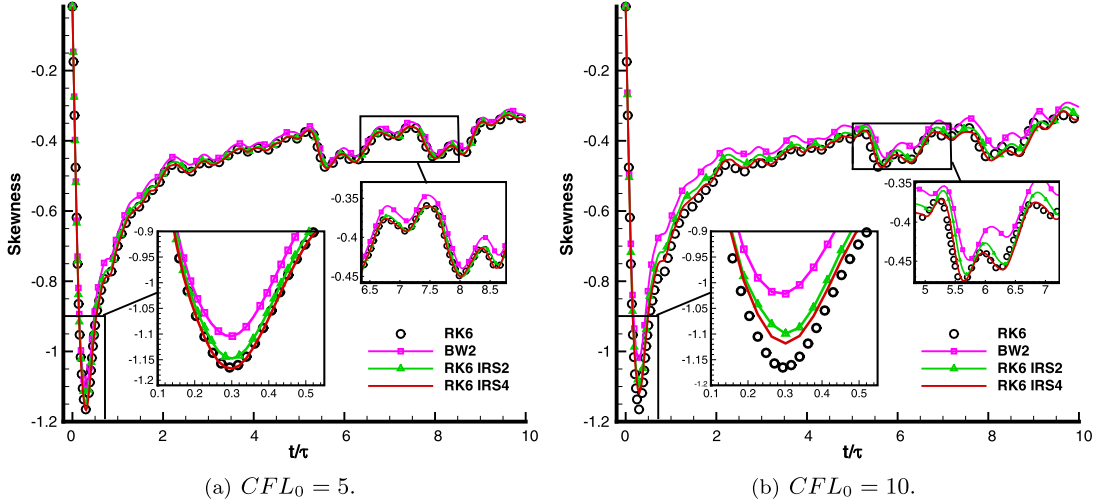

Fig. 22. HIT at $M_{t0} = 1$. Comparison of the time evolution of the skewness of velocity fluctuations obtained with RK6 at $CFL_0 = 1.7$, RK6IRS2, RK6IRS4 and BW2 schemes at $CFL_0 = 5$ and 10.

4.3. Turbulent channel flow

Implicit Large Eddy Simulations (ILES) of flows on a periodic channel (Fig. 24) are carried out using the RK6 at $CFL_0 = 1$ and IRS schemes of 2nd-order and 4th-order accuracy at $CFL_0 = 5$. Note that the choice of $CFL_0 = 1$ for the RK6 scheme is based on an empirical evaluation of the maximum CFL_0 suitable on the compressible channel flow test case (a slight higher CFL conduct to spurious oscillations in the solution with RK6 scheme).

In the following we note h the channel half-width, we use the w subscript to denote wall quantities, b for bulk quantities and CL for centerline quantities; furthermore $+$ superscript denotes wall units based on friction velocity u_τ and the kinematic viscosity at wall ν_w, t_0 is the throughflow time and $t_\tau = \frac{h}{u_\tau}$ the eddy turnover time. Let us also recall the non-dimensionalization proposed by Choi [35] for the computational time step: $\Delta t^+ = \Delta t \frac{u_\tau^2}{\nu}$, or in terms of Δy_w^+, $\Delta t^+ = \Delta y_w^+ u_\tau CFL \frac{M_b}{1+M_b}$ with $M_b = \frac{u_b}{a_w}$ the bulk Mach number.

As for the HIT test cases, quasi-incompressible and compressible flow regimes are considered. For both cases, the physical dimensions of the computational domain are $L_x = 4\pi h$, $L_y = 2h$ and $L_z = \frac{4}{3}\pi h$ in the streamwise, crossflow, and spanwise directions, respectively.

The computational grids have been chosen in order to ensure a good spatial resolution in all directions: in wall units $\Delta x^+ = 10 \div 12$, $\Delta y_w^+ = 0.5 \div 0.8$, $\Delta z^+ = 4 \div 5$. The grid spacing was kept constant in the streamwise and spanwise directions, whereas a constant stretching ratio of about 0.5–1% is used in the wall-normal direction, resulting in $\Delta y_{CL}^+ \approx 4$ for
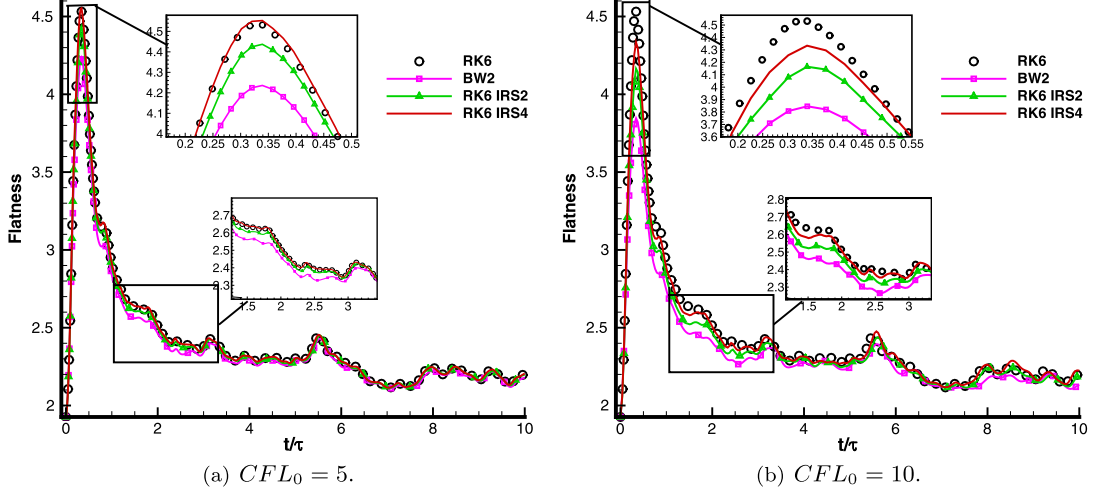
Fig. 23. HIT at $M_{t0} = 1$. Comparison of the time evolution of the flatness of velocity fluctuations obtained with RK6 at $CFL_0 = 1.7$, RK6IRS2, and RK6IRS4 and BW2 schemes at $CFL_0 = 5$ and 10.
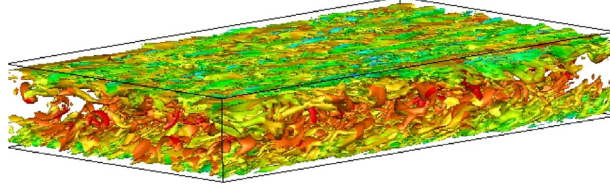

Fig. 24. Quasi-incompressible turbulent channel flow. Iso-surface of λ_2 colored by ρ computed with the DRP11 space discretization and the RK6 on a grid of $192 \times 192 \times 192$ cells.

the centerline cells for both cases. The initial flow field consists of random velocity perturbations superposed to a laminar Poiseuille profile. Transition to turbulence is obtained on very coarse grids (of the order of 64^3); then the turbulent field is successively interpolated on a series of finer grids, up to the final resolution. For each channel configuration, the computation of the flow statistics is started from the same fully developed instantaneous field in order to fairly compare the different time integration schemes.

The flow is driven by a forcing term counteracting the drag force exerted on the channel walls. As proposed in [36], the forcing term is updated at each time step as:

$$F^n = F^{n-1} \times ((\rho u)_b^{\text{target}} / (\rho u)_b^{\text{sch}})$$

(23)

with $F^0 = 1$. An isothermal boundary condition is imposed at the channel walls, and periodicity is applied to the spanwise boundaries.

Since the characteristic grid size is smaller in the direction normal to the wall, the main constraint on the maximum allowable time step for an explicit scheme comes from the wall-normal derivatives. In the other directions, the time step is basically limited by the eddy turnover time and can be much larger. For this reason, IRS schemes are applied only along the wall-normal direction, so to relax the stability constraint and allow a time step comparable to that of the streamwise and spanwise directions. This strategy allows to reduce both the computational cost and numerical errors, since the smoothing operator is applied and solved just in one direction.

4.3.1. Quasi-incompressible turbulent channel flow

The test case chosen is the one described in Ref. [37] at $Re_\tau = \frac{\rho_w u_\tau h}{\mu_w} = 180$. The computations are conducted for a bulk Mach number of $M_b = 0.2$ and a bulk Reynolds number based on the channel half-width h equal to $Re_b = \frac{\rho_b u_b h}{\mu_w} = 2800$. The calculations are carried out on a structured grid made of $192 \times 192 \times 192$ cells, uniformly distributed in x and z, and stretched in the direction normal to the wall. This corresponds to the following mesh size in wall units: $\Delta x^+ = 11.78$, $\Delta y_w^+ = 0.74$, $\Delta y_{CL}^+ = 3.78$, $\Delta z^+ = 3.92$ and $\Delta t^+ = 7.99 \times 10^{-3} CFL_0$, where CFL_0 is the CFL number prescribed at the beginning of the computation. The throughflow time t_0 corresponds to $t^+ = 143.43$ and $0.796 t_\tau$. Table 4 provides the computational times to run the simulations up to $t = 0.15 t_0$ for different schemes. The values are normalized with respect to the time required by the RK6 scheme with $CFL_0 = 1$. The two IRS simulations are nearly five times cheaper than the RK6 at $CFL_0 = 1$ due to the almost negligible overcost introduced by the IRS.

Table 4

Quasi-incompressible turbulent channel flow. Time elapsed to reach $t = 0.15t_0$ with RK6, RK6IRS2 and RK6IRS4 schemes.

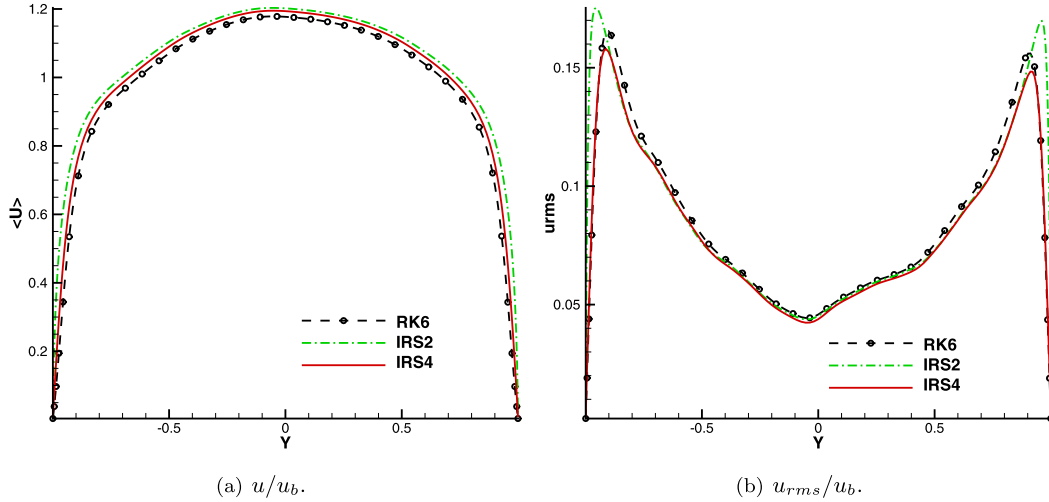|	$CFL_0 = 1$	$CFL_0 = 5$
RK6	1.	x
RK6 IRS2	x	0.21
RK6 IRS4	x	0.22

(a) u/u_b. (b) u_{rms}/u_b.

Fig. 25. Quasi-incompressible turbulent channel flow. Comparison of the mean velocity and u_{rms} profiles obtained at $t = 0.15t_0$ with RK6 at $CFL_0 = 1$ and RK6IRS schemes at $CFL_0 = 5$.

Fig. 25 shows examples of first orders statistics, and namely the mean streamwise velocity and velocity fluctuations accumulated up to $t = 0.15t_0$, for the different schemes. Note that at this integration time, these statistics are not converged yet. Both schemes predict similar profiles. Nevertheless, the IRS4 solution remains closer to that provided by the explicit RK6 scheme.

In the following, we keep just only the IRS4 scheme and we carry out comparisons with recent reference DNS data available in the literature, namely those of Ref. [38].

The mean velocity and RMS of the velocity fluctuations predicted with RK6IRS4 are presented in Fig. 26. Note that these statistics averaged over approximately twenty five throughflow times ($\approx 20t_\tau$). The present ILES appears to be in very good agreement with the reference DNS, despite the coarser grid and less accurate spatial discretization scheme in use.

4.3.2. Compressible turbulent channel flow

The test case chosen is the one proposed by Coleman ([39]) at $Re_{\tau*} = \frac{\sqrt{\rho_{CL}\tau_w}}{\mu_{CL}} = 150$. Precisely, the computations are conducted for a bulk Mach number of $M_b = 1.5$ and a bulk Reynolds number $Re_b = 3100$ as mentioned in [40]. The calculations are carried out on a structured grid made of $256 \times 256 \times 200$ cells, uniformly distributed in x and z, and stretched in the direction normal to the wall. This corresponds to the following mesh size in wall units: $\Delta x^+ = 10.94$, $\Delta y_w^+ = 0.5$, $\Delta y_{CL}^+ = 3.12$, $\Delta z^+ = 4.8$ and $\Delta t^+ = 16.26 \times 10^{-3} CFL_0$. And the throughflow time t_0 corresponds to $t^+ = 120.19$ and $0.53t_\tau$.

Comparison of the mean velocity and RMS fluctuations profiles obtained with RK6 ($CFL_0 = 1$), RK6IRS2 ($CFL_0 = 5$) and RK6IRS4 ($CFL_0 = 5$) at the same integration time for flow statistics ($\approx 13t_0$ or $7t_\tau$) are shown in Figs. 27, 28 and 29. For all time integration schemes, the mean longitudinal velocity profile (Fig. 27a) is well recovered. This is not the case for the rms fluctuations. The RK6IRS2 simulation at $CFL_0 = 5$ is not able to compute as accurately as the reference scheme rms velocity fluctuation profiles (Figs. 27b, c, d) and fails to predict the rms pressure fluctuations. It also clearly overpredicts the rms of velocity gradient fluctuations (Figs. 28 and 29). All these statistics are well recovered with RK6IRS4 scheme at $CFL_0 = 5$ for an overall CPU cost nearly 5 times cheaper than the explicit simulation.

Simulations at iso-CPU cost have also been made with the explicit RK6 and the RK6IRS4 time integration schemes at $CFL_0 = 1$ and $CFL_0 = 5$, respectively. As the simulation with RK6IRS4 has been performed with $CFL_0 = 5$ and because this implicitation technique introduces only a small overcost in terms of CPU time, statistics could be accumulated on an integration time about 5 times larger than the one allowed by the explicit scheme (i.e. $\approx 65t_0$ or $35t_\tau$). Thus, higher order statistics (up to the fourth order) could be converged with RK6IRS4 at the same CPU cost as the one required to converge 2nd order statistics with an explicit simulation.
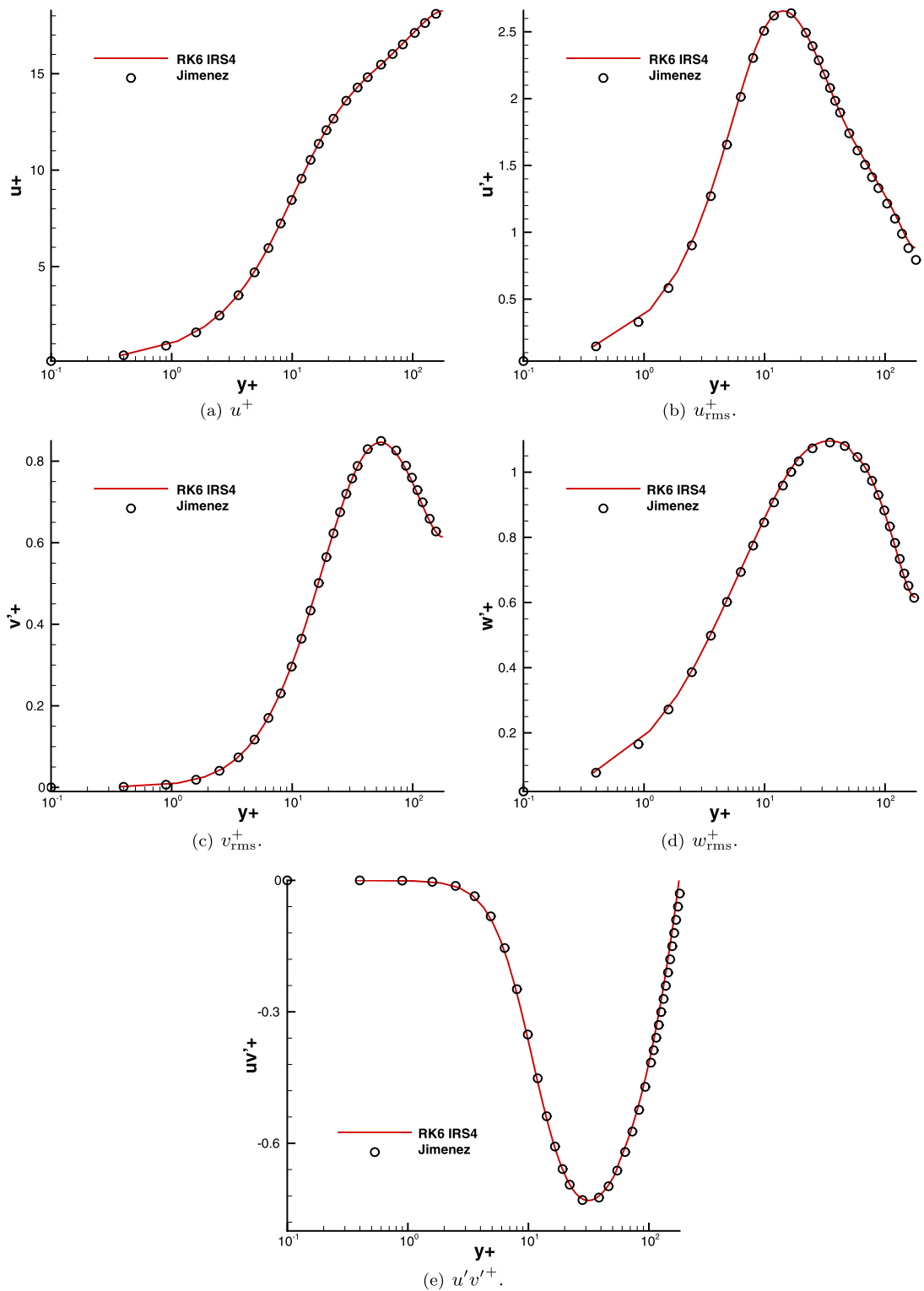
(a) u^+

(b) u_{rms}^+.

(c) v_{rms}^+.

(d) w_{rms}^+.

(e) $u'v'^+$.

Fig. 26. Quasi-incompressible turbulent channel flow. Computed $(.)^+$ profiles for RK6IRS4 at $CFL_0 = 5$.

(a) u^+

(b) u_{rms}^+

(c) v_{rms}^+

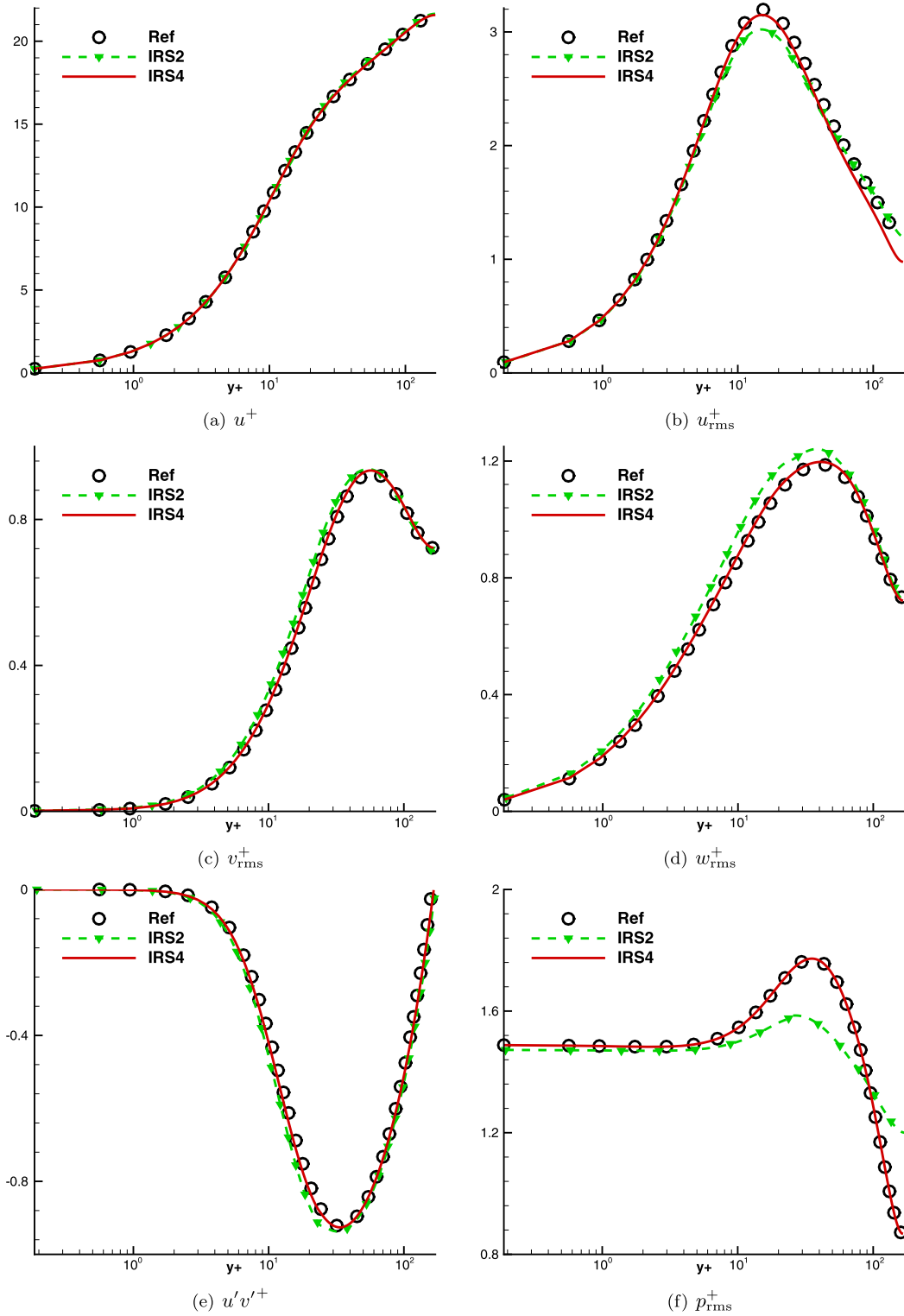(d) w_{rms}^+

(e) $u'v'^+$

(f) p_{rms}^+

Fig. 27. Compressible turbulent channel flow. Computed $(.)^+$ velocity profiles and RMS pressure fluctuations for RK6IRS2 and RK6IRS4 at $CFL_0 = 5$ compared to RK6 predictions at $CFL_0 = 1$ (reference).
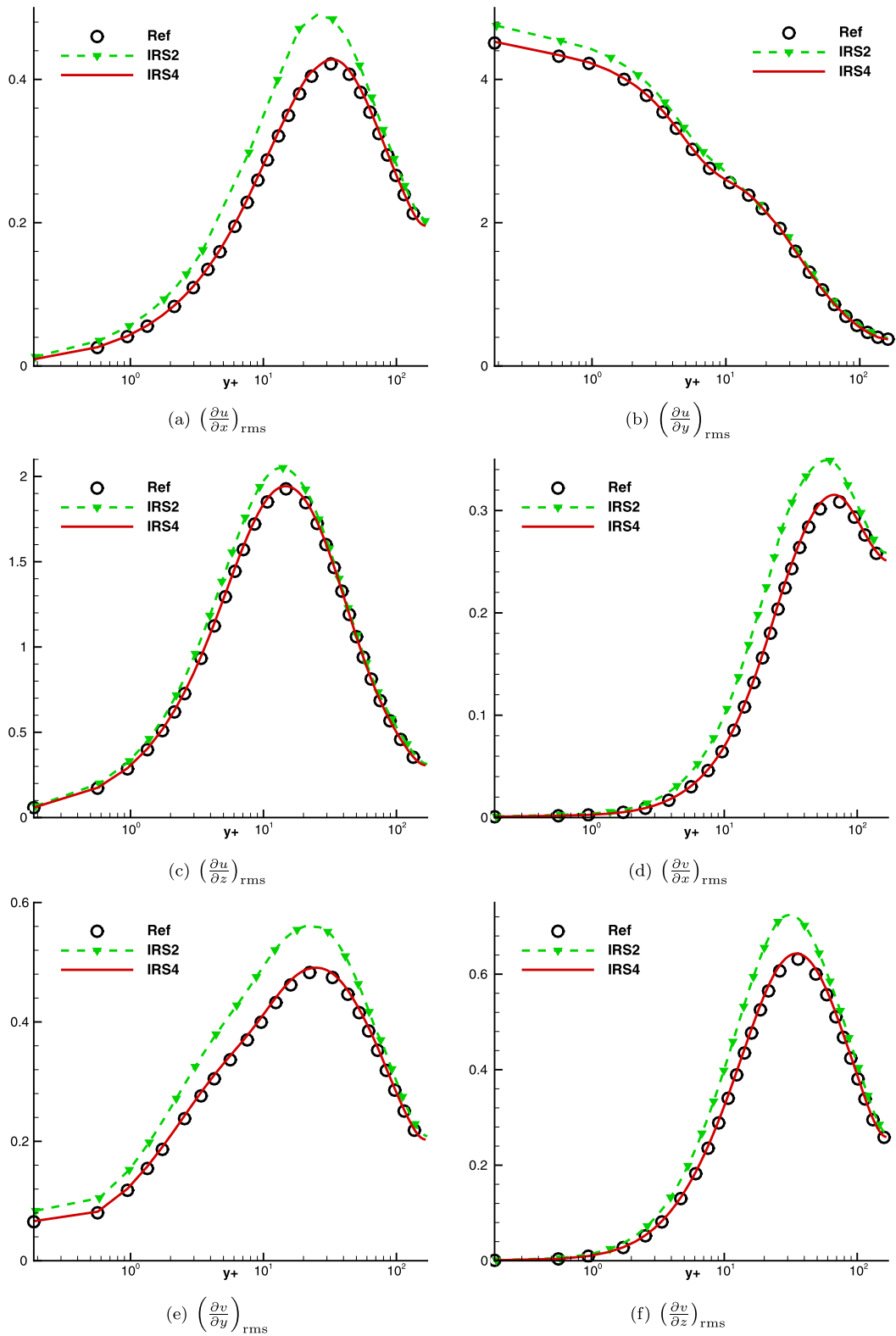
(a) $\left(\dfrac{\partial u}{\partial x}\right)_{\mathrm{rms}}$

(b) $\left(\dfrac{\partial u}{\partial y}\right)_{\mathrm{rms}}$

(c) $\left(\dfrac{\partial u}{\partial z}\right)_{\mathrm{rms}}$

(d) $\left(\dfrac{\partial v}{\partial x}\right)_{\mathrm{rms}}$

(e) $\left(\dfrac{\partial v}{\partial y}\right)_{\mathrm{rms}}$

(f) $\left(\dfrac{\partial v}{\partial z}\right)_{\mathrm{rms}}$

Fig. 28. Compressible turbulent channel flow. Computed $(.)_{\mathrm{rms}}$ profiles of the velocity gradients for RK6IRS2 and RK6IRS4 at $CFL_0 = 5$ compared to RK6 predictions at $CFL_0 = 1$ (reference) [first part].

Fig. 29. Compressible turbulent channel flow. Computed $(.)_{\mathrm{rms}}$ profiles of the velocity gradients for RK6IRS2 and RK6IRS4 at $CFL_0 = 5$ compared to RK6 predictions at $CFL_0 = 1$ (reference) [second part].
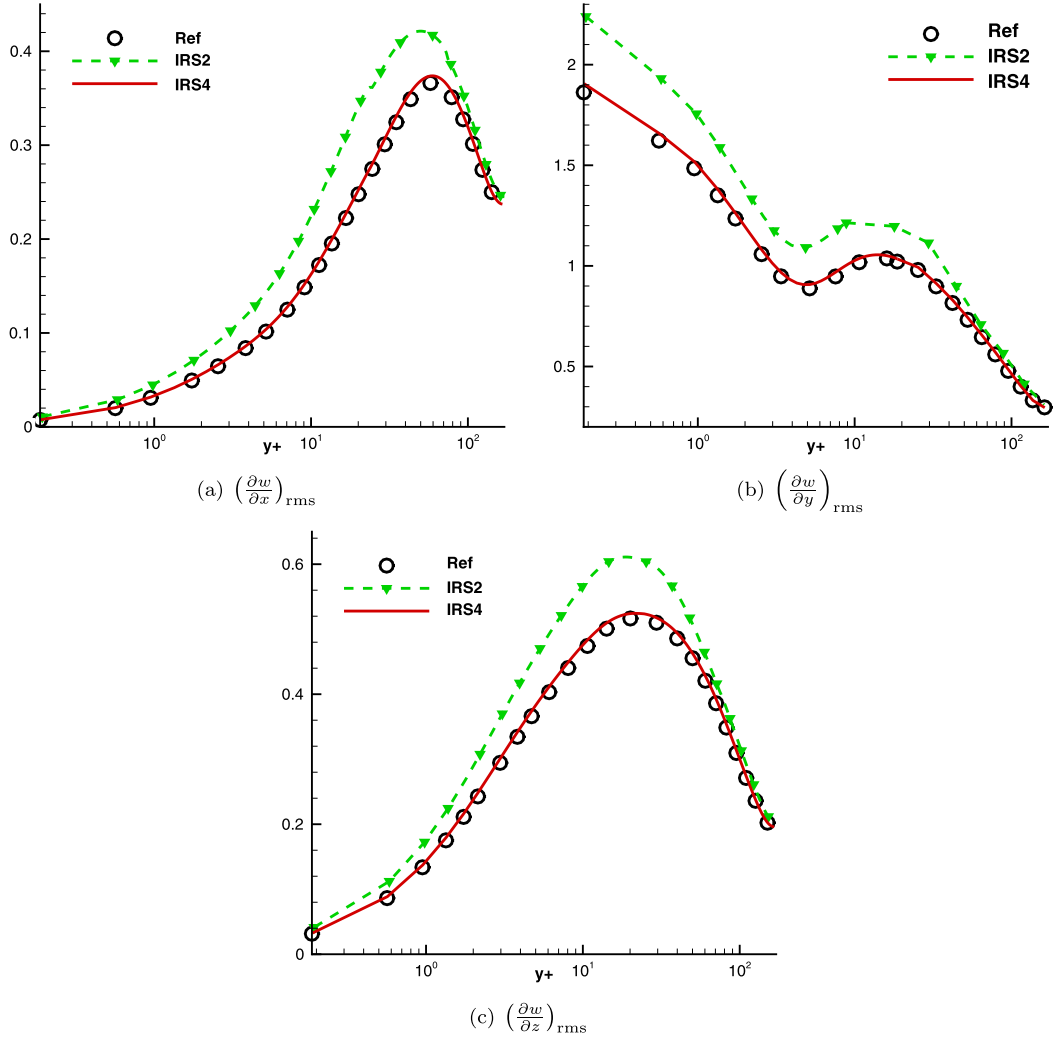
Evolutions of skewness and flatness of velocity and rms pressure fluctuations are reported in Fig. 30. The near zero value of the skewness of w (which should be null in a perfectly converged solution because of the symmetry) indicates the good convergence of these statistics. Present results are similar to those obtained in [37] and [41] for incompressible channel flows at slightly higher Reynolds numbers (respectively $Re_\tau = 180$ and $Re_\tau = 392$).

5. Conclusions

In the present work we introduced a fourth-order accurate central implicit residual smoothing (IRS) operator, well suited for direct and large eddy simulations of compressible turbulent flows. The IRS scheme can be applied in conjunction with existing explicit RK schemes to enlarge the maximum allowable time step without introducing significant additional errors. It is based on the application of a bi-Laplacian smoothing operator to the residual computed at each Runge–Kutta stage, which leads to the inversion of a scalar pentadiagonal system per space direction and per stage. This can be done in an efficient and parallel way.

The theoretical and numerical studies presented in the paper show that the high-order IRS operator introduces a fourth-order error of dispersive nature, which however remains small compared to that of the baseline time scheme, at least for the second and fourth-order RK schemes considered in this study. Numerical applications to turbulent compressible flows show that the IRS allows to reduce the overall computational cost by a factor 3 to 5 with respect to the corresponding explicit RK. The solution accuracy is practically unaffected, at least for CFL numbers comprised between 5 and 10. The method is also very flexible, thanks to the possibility of applying the IRS operator only along space directions involving stiff
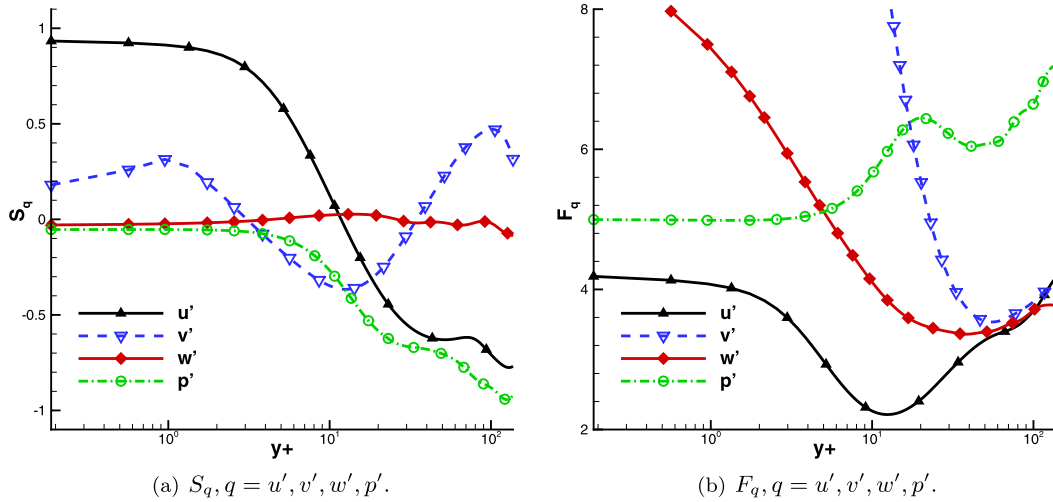
(a) $S_q, q = u', v', w', p'$.

(b) $F_q, q = u', v', w', p'$.

Fig. 30. Compressible turbulent channel flow. RK6IRS4 at $CFL_0 = 5$ results for skewness ($S_q = \frac{\langle q^3 \rangle}{\langle q^2 \rangle^{3/2}}$) and flatness ($F_q = \frac{\langle q^4 \rangle}{\langle q^2 \rangle^2}$) where $q = u', v', w', p'$.

derivatives. The new IRS scheme can also be easily implemented in existing DNS and LES codes using RK time stepping (or other explicit schemes), by introducing the required pentadiagonal inversions at the end of each stage.

The IRS appears to be particularly useful for accelerating the convergence of flow statistics of wall bounded turbulent flows. The scheme efficiency has been demonstrated for both quasi-incompressible and compressible flow cases. The effectiveness of such an approach has been shown in particular for turbulent channel flow computations. For this kind of flow, using the IRS allows computing accurately the higher order statistics (skewness and flatness) of the fluctuating quantities, which are too costly for explicit time stepping schemes.

The IRS schemes presented in this paper were developed for structured grids. Extensions of low-order IRS schemes to unstructured grids have been proposed in the past [42,43] by using an unstructured finite volume approximation of the Laplacian operator. A similar extension could be foreseen for the IRS4, by introducing a suitable approximation of the bi-Laplacian operator. This leads however to sparse matrices, often solved by means of point-Jacobi iterations, which is quite efficient for steady problems but is not suitable for the highly unsteady problems of interest here. A possible alternative is to use line-detection techniques in conjunction with a structured formulation of the IRS operators (see, e.g. [44,45]). This approach, that has the advantage of leading to inversion of banded matrices, will be explored in the future.

As a final remark, the approach presented here has been applied to flow problems with Mach numbers as low as 0.2. For lower Mach numbers, preconditioning of the matrix system may be necessary. Preconditioned IRS techniques are discussed in [46]. Higher-order preconditioned IRS schemes require additional research effort and may be considered for future developments.

Acknowledgements

References

[1] D.P. Rizzetta, M.R. Visbal, G.E. Blaisdell, A time-implicit high-order compact differencing and filtering scheme for large-eddy simulation, Int. J. Numer. Methods Fluids 42 (2003) 665–693.

[2] K. Grimich, P. Cinnella, A. Lerat, Spectral properties of high-order residual-based compact schemes for unsteady compressible flows, J. Comput. Phys. 252 (Nov. 2013) 142–162.

[3] P.-O. Persson, High-order LES simulations using implicit–explicit Runge–Kutta schemes, AIAA Paper 2011-684, 2011.

[4] M.R. Visbal, D.P. Rizzetta, Large-eddy simulation on curvilinear grids using compact differencing and filtering schemes, J. Fluids Eng. 124 (2002) 836–847.

[5] M. Pino Martin, G. Candler, A parallel implicit method for the direct numerical simulation of wall-bounded compressible turbulence, J. Comput. Phys. 215 (2006) 153–171.

[6] A. Jameson, W. Schmidt, E. Turkel, Numerical solutions of the Euler equations by finite volume methods using Runge–Kutta time stepping, AIAA Paper 81-1259, 1981.

[7] A. Lerat, J. Sidès, V. Daru, Lect. Notes Phys. 170 (1982) 343–349.

[8] A. Jameson, T. Baker, Solution of the Euler equations for complex configurations, in: AIAA 6th Computational Fluid Dynamics Conference, Denver, July 1983.

[9] J. Blazek, N. Kroll, C. Rossow, A comparison of several implicit residual smoothing methods, in: ICFD Conference on Numerical Methods for Fluid Dynamics, 1992.

[10] R. Haelterman, J. Vierendeels, D. Van Heule, Optimization of the Runge–Kutta iteration with residual smoothing, J. Comput. Phys. 234 (2010) 253–271.

[11] P. Jorgenson, R. Chima, An unconditionally stable Runge–Kutta method for unsteady flows, Tech. rep., NASA, 1989.

[12] P. Cinnella, A. Lerat, A fully implicit third-order scheme in time and space for compressible turbulent unsteady simulations, in: Proceedings of ECCOMAS 2000, European Computational Fluid Dynamics Conference (ECCOMAS), Barcelona, Spain, September 2000, 2000.

[13] J.W. Kim, Quasi-disjoint pentadiagonal matrix systems for the parallelization of compact finite-difference schemes and filters, J. Comput. Phys. 241 (2013) 168–194.

[14] A. Lerat, C. Corre, Approximations d'ordre élevé pour les écoulements compressibles, in: Ecole de Printemps de mécanique des fluides, Fréjus, France, 2003.

[15] A. Lerat, C. Corre, Higher order residual-based compact schemes on structured grids, in: CFD-Higher Order Discretization Methods, in: Comput. Fluid Dyn. Course, VKI LS 2006-1, von Karman Institute for Fluid Dynamics, 2006, pp. 1–111.

[16] F. Ducros, V. Ferrand, F. Nicoud, C. Weber, D. Darracq, C. Gacherieu, T. Poinsot, Large-eddy simulation of the shock/turbulence interaction, J. Comput. Phys. 152 (1999) 517–549.

[17] C. Bogey, C. Bailly, A family of low dispersive and low dissipative explicit schemes for flow and noise computations, J. Comput. Phys. 194 (1) (Feb. 2004) 194–214.

[18] C. Bogey, N. de Cacqueray, C. Bailly, A shock-capturing methodology based on adaptive spatial filtering for high-order non-linear computations, J. Comput. Phys. 228 (2016) 1447–1465.

[19] T.J. Poinsot, S.K. Lele, Boundary conditions for direct simulations of compressible viscous flows, J. Comput. Phys. 101 (1992) 104–129.

[20] J. Berland, C. Bogey, O. Marsden, C. Bailly, High-order, low dispersive and low dissipative explicit schemes for multiple-scale and boundary problems, J. Comput. Phys. 224 (2007) 637–662.

[21] P.-Y. Outtier, C. Content, P. Cinnella, B. Michel, The high-order dynamic computational laboratory for CFD research and applications, in: 21st AIAA Computational Fluid Dynamics Conference, Fluid Dynamics and Co-located Conferences, American Institute of Aeronautics and Astronautics, June 2013.

[22] Python web site, http://www.python.org/.

[23] CGNS website, http://cgns.sourceforge.net/.

[24] O. Saunier, Méthode d'adaptation de maillages cartésiens basée sur des schémas d'ordre élevé pour les équations d'Euler d'un fluide compressible. Application aux pales de rotor d'hélicoptère, Ph.D. thesis, 2008.

[25] J.D. Lambert, Numerical Methods for Ordinary Differential Systems: The Initial Value Problem, J. Wiley, 1991.

[26] R.e. Dembo, Inexact Newton methods, SIAM J. Numer. Anal. 19 (1982) 400–408.

[27] S. Eisenstat, H. Walker, Choosing the forcing terms in a inexact Newton method, SIAM J. Sci. Stat. Comput. 17 (1996) 16–32.

[28] C. Corre, Contribution à la simulation et à l'analyse des écoulements compressibles, Ph.D. thesis, ENSAM Paris, December 2004.

[29] T. Kloczko, C. Corre, A. Beccantini, Low-cost implicit schemes for all-speed flows on unstructured meshes, Int. J. Numer. Methods Fluids 58 (5) (2008) 493–526.

[30] H. Hollanders, A. Lerat, R. Peyret, AIAA J. 23 (1985) 1670–1678.

[31] R.C. Swanson, E. Turkel, Multistage schemes with multigrid for Euler and Navier–Stokes equations, NASA TP 3631, August 1997.

[32] N. Liamis, V. Couaillier, Unsteady Euler and Navier–Stokes flows simulations with an implicit Runge–Kutta method, in: Computational Fluid Dynamics '94, 2nd European Computational Fluid Dynamics Conference (ECCOMAS), Stuttgart, Germany, September 1994, J. Wiley, 1994, pp. 917–924.

[33] S. Pirozzoli, F. Grasso, Direct numerical simulations of isotropic compressible turbulence: influence of compressibility on dynamics and structures, Phys. Fluids 16 (12) (2004) 4386–4407.

[34] R. Samtaney, D. Pullin, B. Kosovic, Direct numerical simulation of decaying compressible turbulence and shocklet statistics, Phys. Fluids 13 (5) (2001) 1415–1430.

[35] H. Choi, P. Moin, Effects of the computational time step on numerical solutions of turbulent flows, J. Comput. Phys. 113 (1994) 1–4.

[36] G. Gerolymos, D. Senechal, I. Vallet, Performance of very-high-order upwind schemes for DNS of compressible wall-turbulence, Int. J. Numer. Methods Fluids 63 (2010) 769–810.

[37] J. Kim, P. Moin, R. Moser, Turbulence statistics in fully developed channel flow at low Reynolds number, J. Fluid Mech. 177 (1987) 133–166.

[38] S. Hoyas, J. Jimenez, Reynolds number effects on the Reynolds-stress budgets in turbulent channels, Phys. Fluids 20 (2008) 101511.

[39] G. Coleman, J. Kim, R. Moser, A numerical study of turbulent supersonic isothermal-wall channel flow, J. Fluid Mech. 305 (1995) 159–183.

[40] G. Gerolymos, I. Vallet, Pressure, density, temperature and entropy fluctuations in compressible turbulent plane channel flow, J. Fluid Mech. 757 (2014) 701–746.

[41] E. Jeyapaul, G. Coleman, C. Rumsey, High-order and length-scale statistics from DNS of decelerated planar wall-bounded turbulent flow, Int. J. Heat Fluid Flow 54 (2015) 14–27.

[42] C. Rossow, New Results in Numerical and Experimental Fluid Mechanics III, Notes in Numerical Fluid Mechanics.

[43] J. Azevedo, H. Korzenowski, An assessment of unstructured grid finite volume schemes for cold gas hypersonic flow calculations, J. Aerosp. Technol. Manag. 1 (2009) 135–152.

[44] J. Lassaline, D. Zingg, An investigation of directional-coarsening and line-implicit smoothing applied to agglomeration multigrid, AIAA Paper 2003-3435, 2003.

[45] L. Catalano, V. Daloiso, Upwinding and implicit residual smoothing on cell-vertex unstructured grids, Int. J. Numer. Methods Fluids 47 (2005) 895–902.

[46] E. Turkel, Preconditioning techniques in computational fluid dynamics, Annu. Rev. Fluid Mech. 31 (1999) 385–416.