Assessment of time implicit discretizations for the computation of turbulent compressible flows

P. Cinnella[1,2] *, and C. Content[1] †

[1] *Laboratoire DynFluid, Arts et Métiers ParisTech, 151 bvd de l'Hôpital, 75013 Paris, France*

[2] *Università del Salento, via per Monteroni, 73100, Lecce, Italy*

Restrictions on the maximum allowable time step of explicit time integration methods can be very severe for direct and large eddy simulations of compressible turbulent flows at high Reynolds numbers, for which extremely small space steps have to be used close to solid walls in order 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. A successful implicit time scheme should provide the best possible compromise between these opposite requirements. In this paper, several implicit schemes assessed against two explicit time integration techniques, namely a standard four-stage and a six-stage optimized Runge–Kutta methods, in terms of computational cost required to achieve a threshold accuracy level for the simulation of compressible turbulent flows. Precisely, a second-order backward scheme solved by means of matrix-free quasi-exact Newton subiterations is compared to time-accurate Runge–Kutta implicit residual smoothing (IRS) schemes. A new IRS scheme of fourth-order accuracy, based on a bilaplacian operator, is developed to improve the accuracy of the classical second-order approach. Numerical results show that the proposed IRS scheme leads to reductions in computational time by about a factor 5 for an accuracy comparable to that of the corresponding explicit Runge-Kutta scheme.

I. 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 point 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 stability 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 adopting 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.

*Professor, DynFluid, Università del Salento, Lecce, Italy. Senior AIAA member.

†Research engineer, Laboratoire DynFluid, Paris. AIAA member.

Then, subiteration techniques are generally used to reduce supplementary errors associated to successive simplifications of the implicit operator. Unfortunately, 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. Also note that 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 Ref. 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. 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.

In this paper, we investigate the suitability for LES and DNS computations of two different implicit techniques. The first one uses a backward linear multistep method of second-order accuracy (BWM2), often used in the litterature[1,5] for the discretization of time derivatives. In this work, the associated nonlinear system of ordinary differential equations introduced by the implicit time discretisation is solved at each physical time step by means of an matrix-free iterative procedure.[6] The BMW2 is A-stable and L-stable, leading to an unconditionally stable fully discrete approximation for any choice of the spatial discretization, but also to increasing damping of high-frequency solution modes as the time step is increased. The second implicit approach under investigation consists in the introduction of a time-accurate implicit residual smoothing (IRS) operator at each stage of a standard explicit Runge–Kutta scheme.[7] The IRS technique, initially proposed by Lerat and Sides[8] for the Lax-Wendroff scheme, and adapted to Runge-Kutta schemes by Jameson,[9] has been widely used in the past to speed up convergence of steady Euler and Navier-Stokes calculations. Both central and upwind smoothing techniques exist.[10,11] The latter lead 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 Ref. 12, 13) have been applied to the simulation of slow unsteady flows. 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 Runge–Kutta stage. The main asset of IRS is that it provides a significant increase of the maximum allowable time step compared to an explicit Runge–Kutta scheme, while keeping a computational cost not much higher, thanks to the efficient inversion of scalar tridiagonal matrices. Nevertheless, care must been 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. In the attempt of improving the accuracy of IRS approaches for scale-resolving time-accurate simulations, a higher-order IRS scheme based on the application of a bilaplacian smoothing operator to the residuals is proposed and investigated. Such a scheme leads to the inversion of a scalar pentadiagonal system in each mesh direction, which can be treated in an efficient way.[14]

All of the numerical methods considered in this study are implemented within the in-house code DynHoLab.[15] 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.

In the following of this draft paper we first briefly describe the computational code in use, than we introduce the time integration techniques under study, and finally present numerical results obtained a linear model problem and for the decay of compressible homogeneous and isotropic turbulence. Preliminary results for a turbulent channel flow are also presented.

II. The DynHoLab code

In this section we briefly describe the DynHoLab code[15], used for the present calculations. The code aims at solving systems of Partial Differential Equations (PDE) of the general form

$$\frac{\partial w}{\partial t} + \nabla \cdot F(w) = S(w) \tag{1}$$

with $w \in \mathbb{R}^n$ the state vector of unknown variables of the problem, n being the number of scalar equations of the system, $F \in \mathbb{R}^{n \times d}$ the linear or non linear fluxes of the problem (d being the number of space dimensions) and $S \in \mathbb{R}^n$ the source terms, all depending on the chosen physical model. DynHoLab can handle different

governing equations both in two and three dimensions of space. To this purpose, it is sufficient to specify the nature of the unknowns w, of the physical fluxes F and source terms S. For instance, by specifying $w \in \mathbb{R}$, $F = aw$ with $a \in \mathbb{R}$ and $S = 0$ one recovers a scalar advection equation. Equation (1) is discretized in space on a set of m structured or unstructured grids by means of some approximation technique, to give

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

where R is a space approximation operator. $\mathcal{R}$ may be obtained by applying any approximation method in use, including finite differences, finite volumes or finite elements. The semi-discrete equation (2) represents a set of N ordinary differential equations, N depending on m, and on the number of degrees of freedom, control volumes or grid points contained in each grid. At the present stage of development, the DynHoLab code uses a finite volume methodology on multi-block structured meshes. Several discretization schemes are available. For the calculations hereafter, we use a flux-extrapolation high-order MUSCL scheme presented in the following. Some results are also presented for a Dispersion Relation Preserving scheme optimized on 11 points (DRP11)[16] to check the effect of the space discretization.

DynHoLab uses a combination of a compiled and type-safe language (FORTRAN) with an interpreted and dynamically typed language (Python[17]): 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,[18] 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.

III. Space discretization schemes

We discuss numerical methods for a system of conservation laws of the form (1).

In a Cartesian reference frame, and if no source terms are present, the last one writes:

$$\frac{\partial w}{\partial t} + \sum_d \frac{\partial f_d(w)}{\partial x_d} = 0 \tag{3}$$

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:

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

The main focus of the present paper is on efficient time integration schemes. For space discretisation, we consider a nineth-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.[19,20] Such a scheme is equivalent to a MUSCL scheme contructed by applying a nineth-order extrapolation to the fluxes. This is why 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 $x_j = j\,h$ the semi-discrete scheme in space writes:

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

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, $\mu(\bullet)_{j+\frac{1}{2}} := \frac{1}{2}(\bullet)_{j+1} + (\bullet)_j$, and Q is a dissipation matrix, here taken equal to the Roe matrix associated to the physical flux f. This scheme is instrisically 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.[7] and Φ is a discrete form of Ducros's vortex sensor,[21] 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.

IV. Time integration schemes

IV.A. 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

$$\left\{\begin{array}{rcl} w^{(0)} & = & w^n \\ \Delta w^{(k)} & = & -a_k \Delta t \mathcal{R}(w^{(k-1)}), k = 1, ...s \\ w^{n+1} & = & w^{(s)} \end{array}\right. \tag{4}$$

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.*,[7] of fourth-order accuracy for linear problems and second-order accurate in general, and the six-stage optimized RK of Bogey and Bailly[16] (RK6), which is second-order accurate.

When coupled with a purely centered approximation for the space derivatives, this 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 13, 22). The RK6 scheme, on the other hand, has a stability limit that is about 40% larger than the RK4 scheme.[16]

Furthermore, the schemes have 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.

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 appearence 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 time step predominates. This is of the order of the second power of the mesh size, which represents a very severe constraint for high-Reynolds turbulent flows, characterized by very fine mesh resolutions in the near wall region.

IV.B. Backward difference schemes

The second implicit time integration approach considered here is a second-order backward multistep scheme (BW2), also called the Gear scheme, which is A-stable and L-stable (see, e.g. 23). 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 $\Delta w^n = w^{n+1} - w^n$. 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) unstead if a direct linear solver, the resulting algorithm is categorized as an "inexact" Newton s method[24]. A simple inexact method results in the following convergence criterion on each linear iteration:

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

where the forcing term γ is a constant smaller than unity. For details on the forcing term see Reference 25.

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 $\rho(G)$. 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. Here, an extension of the method proposed in Ref. 6,26 for the resolution of stationary problems is applied. It consists in an iterative resolution of the implicit stage which is independent of the scheme used for the explicit stage. This implicit scheme is matrix-free and may be interpreted as using in the left hand side of (6) a numerical flux which is not consistent with the one used in the explicit stage but which makes the implicit stage particulary simple to deal with. As shown by in Ref. 6, the use of sub-iterations to correct the implicit stage leads to an amplification factor of the same order as the one obtained by the direct inversion of the system. The number of sub-iterations required to access a resonable convergence rate depends on the space discretization scheme used for the evaluation of the right hand side of (6). Numerical experiments show that the schemes in use in this study require about 15 sub-iterations.

The BW2 scheme has been often used for implicit DNS and LES simulations[4,1,5] and it is considered to offer adequate accuracy. Like all second-order schemes, its leading error term is of dispersive nature. BW schemes of orders higher than 2 exist (see, e.g; 13) but unfortunately they are no longer A-stable and may lead to severe contraints on the time step for advection dominated problems and low-dissipative schemes, which makes them uninteresting for the applications targeted in this study.

IV.C. Implicit residual smoothing

IV.C.1. Second-order implicit residual smoothing

The stability domain of the preceding explicit RK schemes can be enlarged by using an implicit residual smoothing (IRS) technique. The smoothing operator may include the contribution of viscous fluxes 27, 28, 29, 13. 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.

Following Ref. 27, a Lax-Wendroff-like implicit operator is applied at each Runge-Kutta stage as

$$\left\{ \begin{array}{ccc} w^{(0)} & = & w^n \\ \mathcal{J}\Delta w^{(k)} & = & -a_k \Delta t \mathcal{R}(w^{(k-1)}), k = 1, ...s \\ w^{n+1} & = & w^{(s)} \end{array} \right. \tag{9}$$

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

$$\mathcal{J} = 1 - \theta(\tfrac{\Delta t}{\delta x})^2 \delta(\lambda^{e\,2}\delta) \tag{10}$$

where θ is a tuning parameter. For a pure advection problem, the scheme can be made inconditionally stable by choosing:[29,13,22]

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

For multidimensional problems, the implicit operator is obtained by factorization of 1D operators in each mesh direction. This leads to the inversion of a tridiagonal system per mesh direction.

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:

$$-\theta \Delta t^2 \sum_d \lambda_d^{e\,2} f^e_{d\,3x_d} + o(\Delta t^2) \tag{11}$$

Being proportional to a third derivative of the flux f^e_d, this error is recognized to have a dispersive nature. It become 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.

IV.C.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 stabilised 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. With this choice, the IRS operator (referred-to as IRS4 in the following) becomes

$$\mathcal{J} = 1 + \theta(\tfrac{\Delta t}{\delta x})^4 \delta(\lambda^{e\,4}\delta^3) \tag{12}$$

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

$$-\frac{1}{12}\theta \Delta t^4 \sum_d \lambda_d^{e\,4} f^e_{d\,5x_d} + o(\Delta t^4) \tag{13}$$

Like for the IRS2 scheme, it is of dissipative nature, but now it is of the same order or higher of 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, for which an efficient parallel implementation is possible.[14]

IV.C.3. Effect of IRS on stability

Consider the kth RK step:

$$w^{(k)} - w^{(0)} = -a_k \Delta t \mathcal{R}(w^{(k-1)}) \tag{14}$$

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 know that an IRS operator is applied to the left-hand side of (14) . 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 IRS operator. For the preceding IRS schemes:

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

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

where ξ is the reduced wave number. These are real numbers and always greater or equal than 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 has the effect of increasing 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, but qualitatively similar results were found for other schemes, including the DRP11 scheme of Ref. 16).

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 $\kappa_{10} = \frac{1}{1260}$:

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

For IRS schemes, the preceding expression for $\widehat{\mathcal{R}}$ has to be divided by $\widehat{\mathcal{J}}$, as discussed above. The fully discrete scheme is stable if the locus of $\widehat{\mathcal{R}}/\widehat{\mathcal{J}}$ in the complex plane fits within the stability region of the time integration scheme, i.e. the set of complex numbers G such that $|G| \leq 1$. Figure 1 shows the stability regions of the RK4 and RK6 schemes, 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 $\theta \gtrapprox 0.005$. 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. The amplification factors at the last RK step for the RK4 and RK6 schemes coupled the FE9 spatial scheme are represented in figure (2) as a function of the reduced wave number. For the explicit schemes, $|G| > 1$ for some ranges of wavenumbers as soon as $CFL \gtrapprox 2$. For IRS, $|G|$ is always below 1 provided that the smoothing parameter is selected according to the criteria enonciated 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 figure (3) a. 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 figure (3) b. It can be seen that, as expected, the 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 unacceptably 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. In the following Section we investigate numerically the impact of the additional error for some well-documented test cases.

V. Numerical results

V.A. 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:

$$w(x,y,t) = \exp\left(-100\ r(t)\right) \qquad \text{with} \qquad r(t) = (x-(x_0+at)) + (y-(y_0+at)) \tag{15}$$

The hump is initalized at $t=0$ at the position $(x_0,y_0)=(-0.5,-0.5)$ and is advected along the diagonal at $v=\frac{\sqrt{2}}{2}$.

The calculations are carried out with the implicit BW2 scheme and

********* Cedric –¿ Dix ou Quinze??????

10 quasi-exact Newton subiterations,

****************************** and with thr IRS-RK schemes. 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 taken equal to 1, to allow comparisons with the explicit schemes. Results are shown for the FE9 scheme and for the DRP11 scheme, of fourth-order accuracy and optimized in the Fourier space. 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 figure (4) 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 discretizations, even if its convergence order is lower for this linear case. This good behavior is conserved even when introducing IRS. As expected, 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 slight reduction of the error on coarser meshes could be due to the reduced dissipation of the IRS4 scheme compared to the baseline. Figures 5 and 6 show the computed fields of w at the final time 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 as 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 optimised 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 low-dissipative RK6 scheme. Figure 7 shows horizontal sections of the hump at the final time for different schemes, 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 low 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.

Table 1 provides the final computational times for different schemes and CFL numbers. 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 ovecost obtained for IRS2 and IRS4 proves the efficiency of the pentadiagonal solver. For the RK6 scheme, the cost of adding IRS is lower, namely, a bit more than 30%, since the computational cost of the baseline 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 70% lower than the RK4 at $CFL = 1$, while keeping a similar accuracy. Even if the RK4 can be run at CFL greater than 2 (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.

	$CFL = 1$	$CFL = 5$	$CFL = 10$
rk4	1.	x	x
rk6	1.40	x	x
rk4 IRS2	1.56	0.26	0.14
rk6 IRS2	1.83	0.38	0.19
rk4 IRS4	1.54	0.26	0.14
rk6 IRS4	1.88	0.39	0.20
BW2	4.21	1.17	0.84

Table 1. Time Elapsed to reach $t = 1$ with all time integration schemes on a 100x100 mesh. FE 9

V.B. Isotropic homogeneous turbulence

The implicit schemes are then applied to the numerical simulation of decaying compressible homogeneous isotropic turbulence.

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 strong transient, through which the compressible components of the turbulent structures increase and a physical state of fully developed turbulence is reached.

Simulations are carried out at turbulent Mach number $M_t = 0.2$ and 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 Reference 30.

In order to understand the effect of the increasing time step we compare results obtained by Runge–Kutta scheme and those provided by BW2 method with several CFL numbers.

********** CEDRIC: CHECK

Fixed tolerance or fixed subiterations?????

In this study, the convergence criterion (γ from eq(7)) has been taken equal to $\gamma \approx 1.e-2$. The huge computational cost of the method is due to the use of spatial schemes with a large stencil (11 points per direction). Actually, as mentionned before, the inversion of the inner-loop by the matrix-free algorithm requires about 15 subiterations for this kind of space schemes.

****

Figure 8 provides the turbulence kinetic energy spectra computed with the FE9 scheme along with different implicit schemes and a time step corresponding to a CFL equal to 10 at the initial time (CFL_0).

Results obtained with RK6 at $CFL_0 = 1$ are also reported for comparison and represent our reference. The results obtained at simulation times $t = 5\tau$ and $t = 10\tau$ show that the time integration scheme has virtually no impact on the computed spectra.

Inspection of the time evolution of the enstrophy (fig 9) 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 plotting accuracy.

The time evolution of the pressure root mean square p_{rms} (figure 10(a)) 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: even for simulations at a low turbulent Mach number, the characteristic times scales of the thermodynamic fields are at least 10 times 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, especially for simulations at $CFL_0 = 10$. The RK6IRS2 scheme exibits 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 distribution, namely, the skewness and the flatness of the velocity fluctuations, to check the effect of the implicit treatments on fine turbulence features. All of the schemes perform quite well, with some large errors 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. Surprisingly, for this quantity, the BW2 scheme gives a better solution than RK6IRS4.

Table 2 provides the computational times for the different schemes. Due to the large number of inner iterations required, the BW2 leads to an increase in computational time with respect to the reference RK6 simulation when the time step is increased by 5, and to a similar computational time when the time step is doubled. For this reason, this scheme is no longer used in the following.

On the contrary, IRS schemes generate 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 five times smaller.

	$CFL_0 = 1$	$CFL_0 = 5$	$CFL_0 = 10$
RK6	1	x	x
RK6IRS2	x	0.21	0.11
RK6IRS4	x	0.22	0.11
BW2	x	1.81	0.93

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

V.C. Turbulent Channel Flow

Preliminary Implicit large eddy simulations (ILES) of the flow on a periodic channel (13) were carried out using the RK6 and IRS schemes of 2nd-order and 4th-order accuracy. The test case chosen is the one described in Ref. 31 at $Re_\tau = 180$.

The computations were conducted for an average Mach number of 0.2 and an average Reynolds number based on the bulk velocity U_b, the wall viscosity, and the channel half-width h equal to 2800. The flow is driven by a forcing term counteracting the drag force exerted on the channel walls. The forcing term is updated at each time step as:

$$F = F - \frac{1.}{A}\left(Q_0 - 2.Q^n + Q^{n-1}\right) \tag{16}$$

where Q_0 is the imposed mass flow rate, Q^n is the mass flow rate computed at time t^n and A the throughflow area. An isothermal boundary condition is imposed at the channel walls, and periodicity is applied to the spanwise boundaries.

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 calculations were carried out on a structured grid made of $64 \times 150 \times 64$ 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: $\Delta x+ = 35.18$, $\Delta y+_w = 0.19$, $\Delta y+_{CL} = 3.8$ and $\Delta z+ = 11.73$, where the subscript w denotes the first cell close to the wall and CL the channel centerline. 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 only applied on 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 is also useful to reduce computational costs and numerical errors, since the smoothing operator is solved just in one direction.

Table 3 provides the computational times to run the simulations up to $t = 0.15t_0$ for different schemes. The values are normalized with respect to the time required by the RK6 scheme with an initial CFL number $CFL_0 = 1$. The two IRS schemes simulations are about six times cheaper than the RK6.

	$CFL_0 = 1$	$CFL_0 = 5$
rk6	1.	x
rk6 IRS2	x	0.16
rk6 IRS4	x	0.17

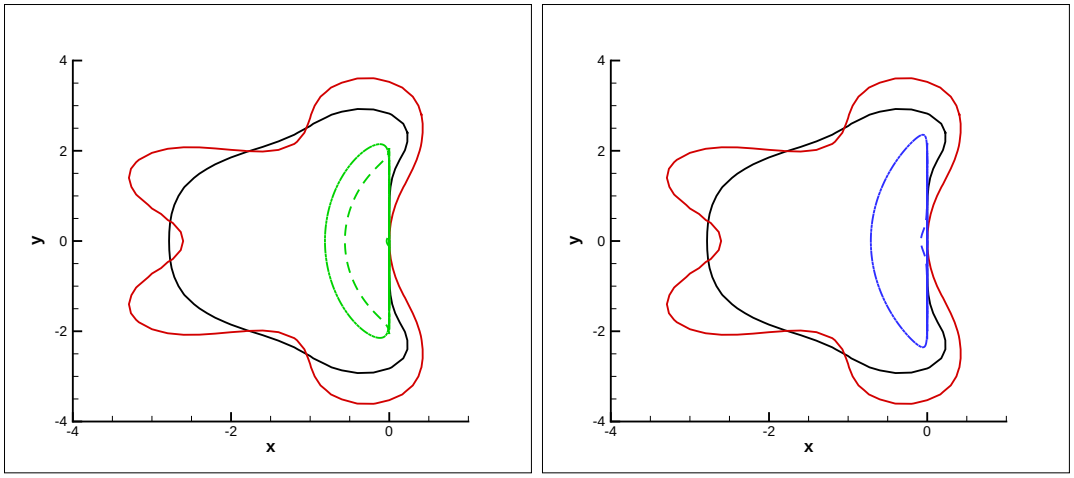
Table 3. Time Elapsed to reach $t = 0.15t_0$ with RK6, RK6IRS2 and RK6IRS4 schemes. FE 9 ******** !!!!! Les cots de calcul sont bizarres: CHECK!!!!! ********** **********

Figures 14 and 15 show the first orders statistics of the different simulations.

**********TO BE CHANGED !!!!!!!

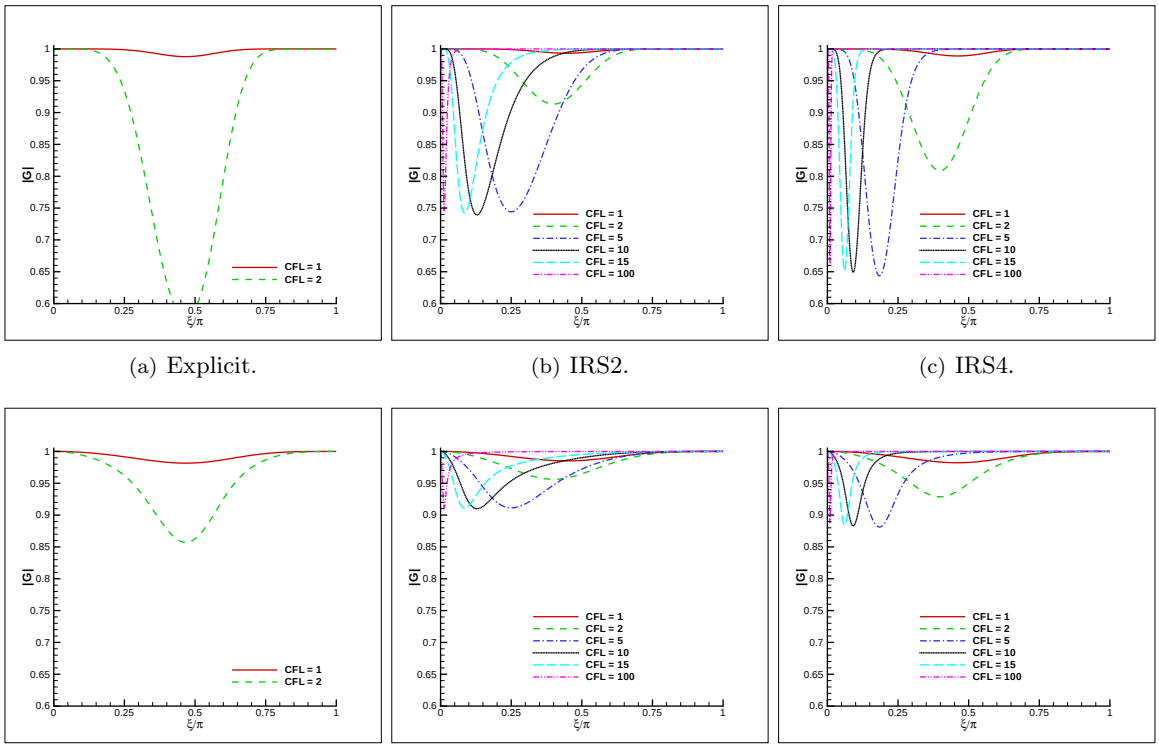
Note that these statistics are not converged yet and will be replaced in the final version of the paper.

All schemes predict similar profiles for $< U >$, u_{rms}, w_{rms}. Nevertheless, the solution provided by IRS4 remains closer to the reference RK6 solution. The prediction of v_{rms} is less accurate. Oscillations are visible on RK6 predictions and seem to be amplified by the use of IRS. Nevertheless, the calculations are not converged yet.



(a) IRS2. (b) IRS4.

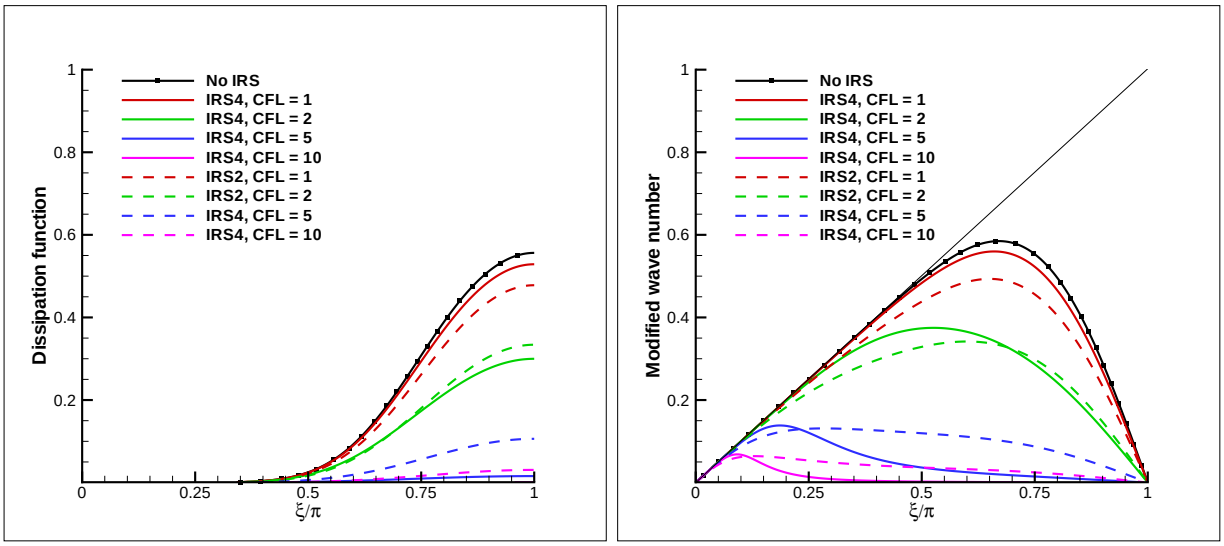
Figure 1. Stability regions of the RK4 (black line) and RK6 (red line) schemes and locus of the FE9 spatial operator with implicit residual smoothing for CFL numbers 1 (solid line), 2 (dashed line) and 10 (dash-dotted line).



(a) Explicit. (b) IRS2. (c) IRS4.

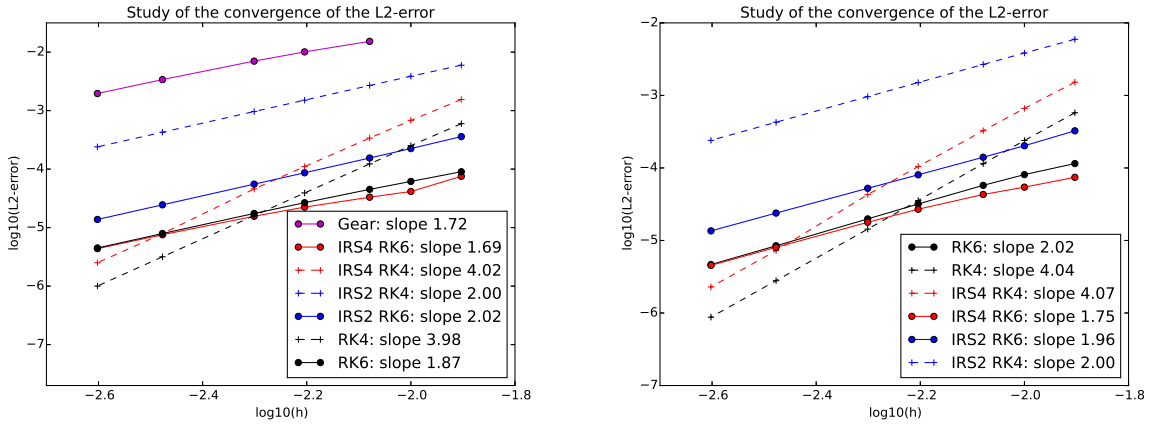
(d) Explicit. (e) IRS2. (f) IRS4.

Figure 2. Amplification factors versus the reduced wave number for different values of CFL. Top: RK4, bottom, RK6.



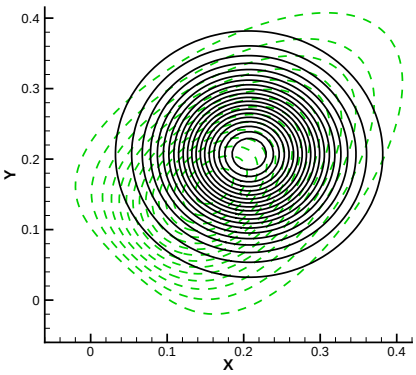
(a) Dissipation function. (b) Modified wave number.

Figure 3. Dissipation function and modified wave number for the modified spatial operator $\mathcal{J}^{-1}\mathcal{R}$ for different values of the CFL number.

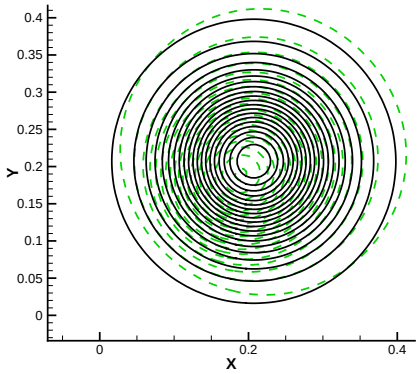


(a) FE 9. (b) DRP 11.

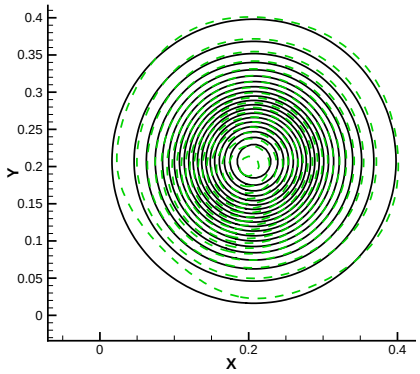
Figure 4. L_2 norm of the error with respect to the exact solution for the FE9 scheme (left) and the DRP11 scheme (right). $CFL = 1$.



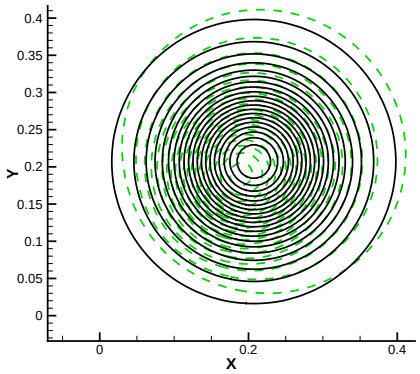
(a) BW2.



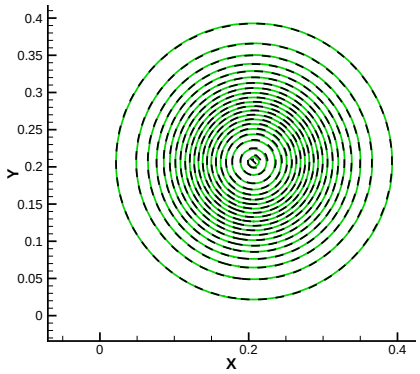
(b) RK4IRS 2.



(c) RK4IRS 4.

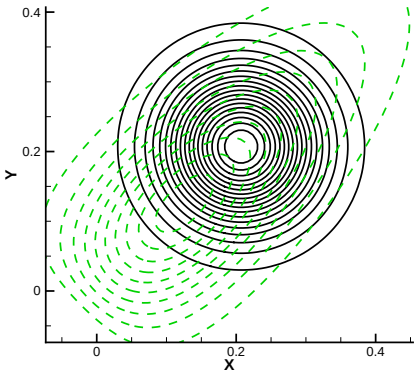


(d) RK6IRS 2.

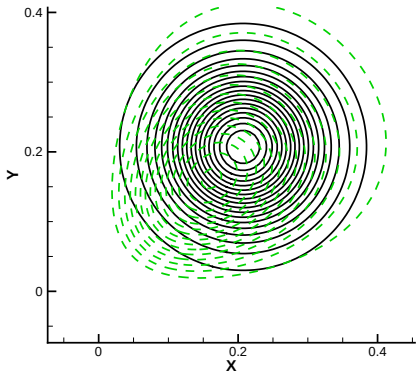


(e) RK6IRS 4.

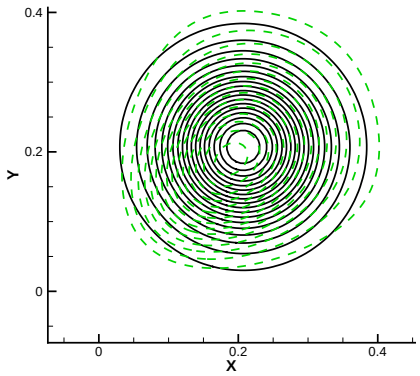
Figure 5. Iso-values of the hump at $t = 1$. $CFL = 5$.



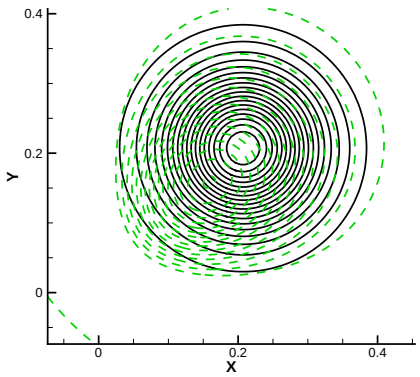
(a) BW2.



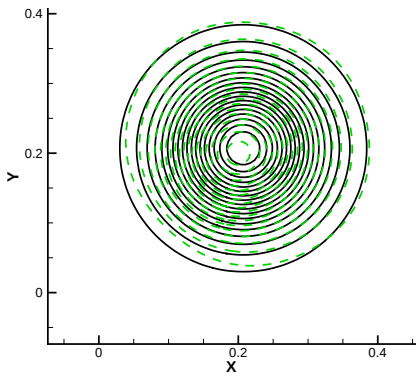
(b) RK4IRS 2.



(c) RK4IRS 4.

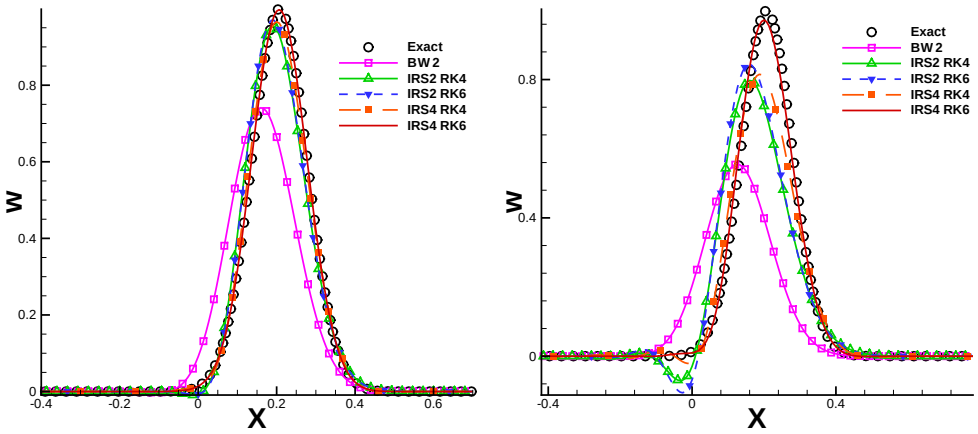


(d) RK6IRS 2.

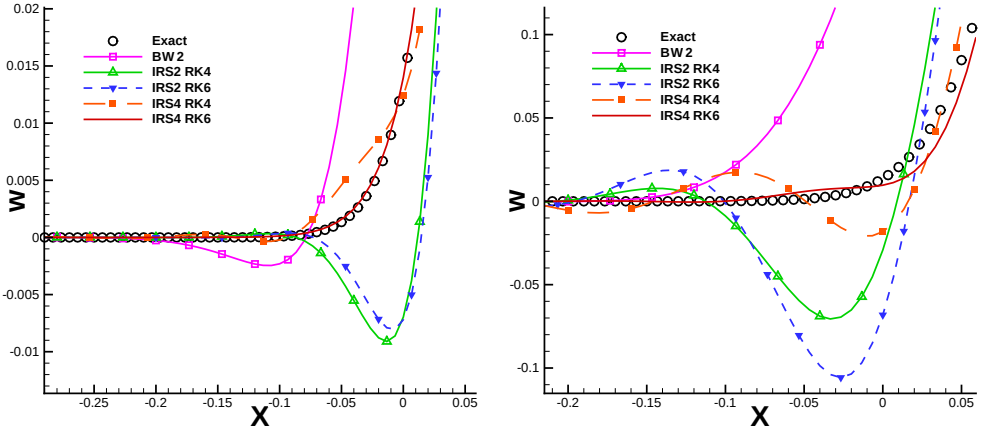


(e) RK6IRS 4.

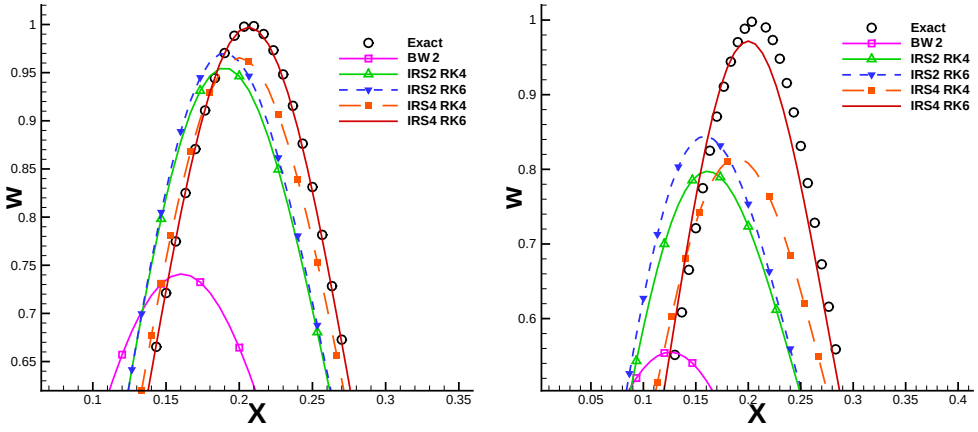
Figure 6. Iso-values of the hump at $t = 1$. $CFL = 10$.



(a) $CFL = 5$. (b) $CFL = 10$.

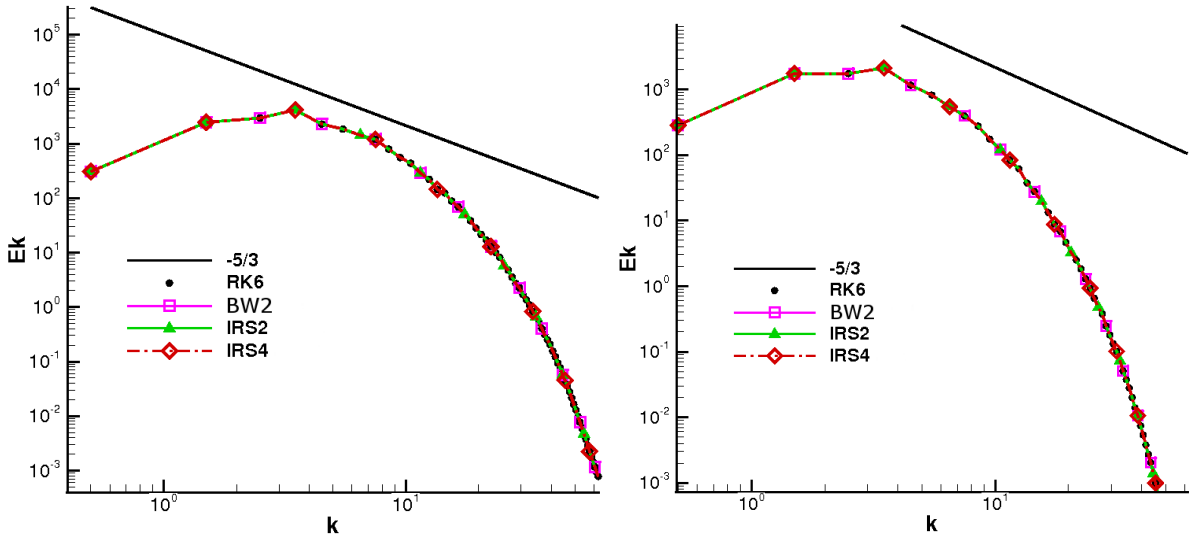


(c) $CFL = 5$. Close-up of the bottom part. (d) $CFL = 10$. Close-up of the bottom part.



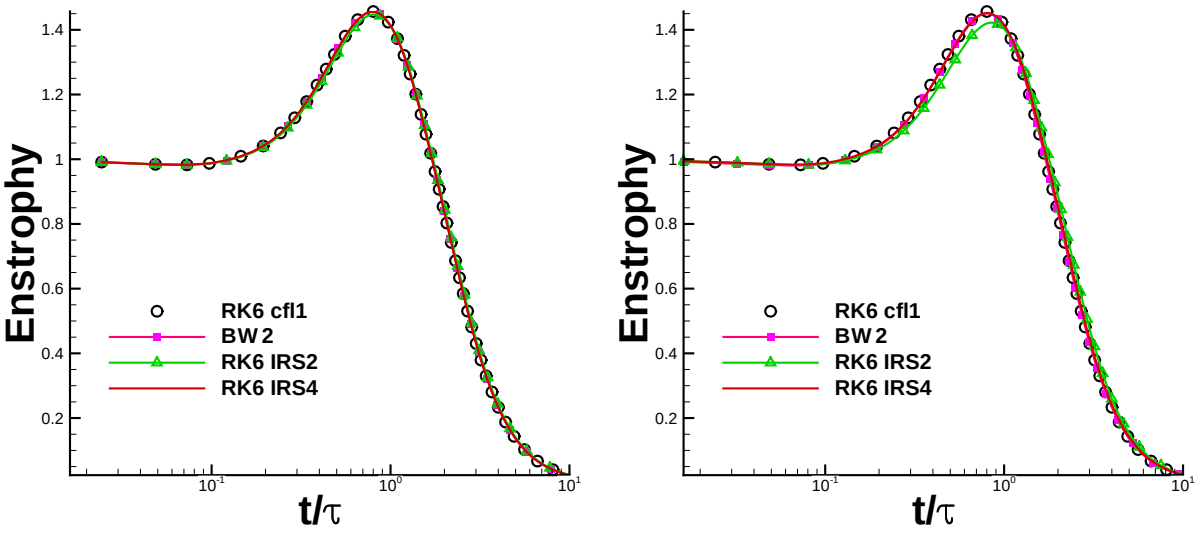
(e) $CFL = 5$. Close-up of the upper part. (f) $CFL = 10$. Close-up of the upper part.

Figure 7. Section of the hump at the pic of w ($y \approx 0.75$), $t = 1$. $CFL = 5$ and $CFL = 10$.



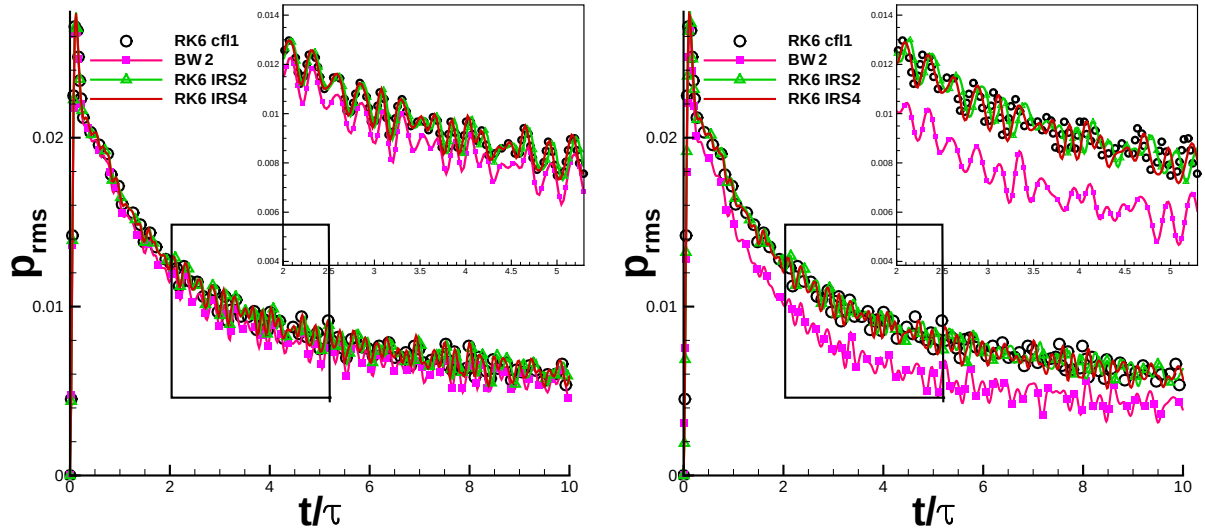
(a) $t = 5\tau$. (b) $t = 10\tau$.

Figure 8. Kinetic energy spectra obtained with RK, RKIRS and BW2 schemes at CFL= 10.



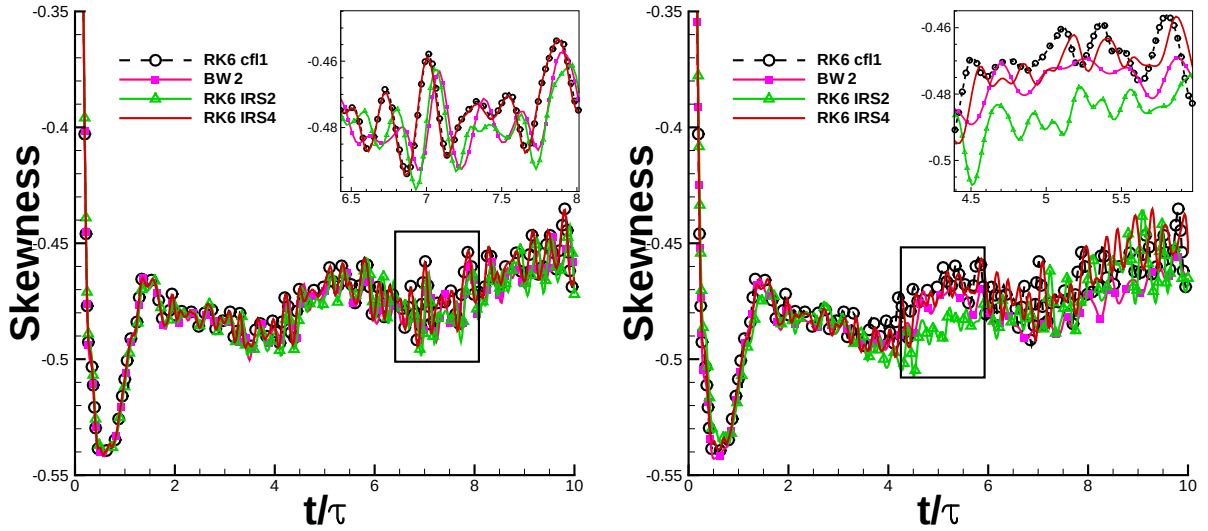
(a) $CFL = 5$. (b) $CFL = 10$.

Figure 9. Comparison of the time evolution of the enstrophy obtained with RKIRS and BW2 schemes with CFL=5 and 10.



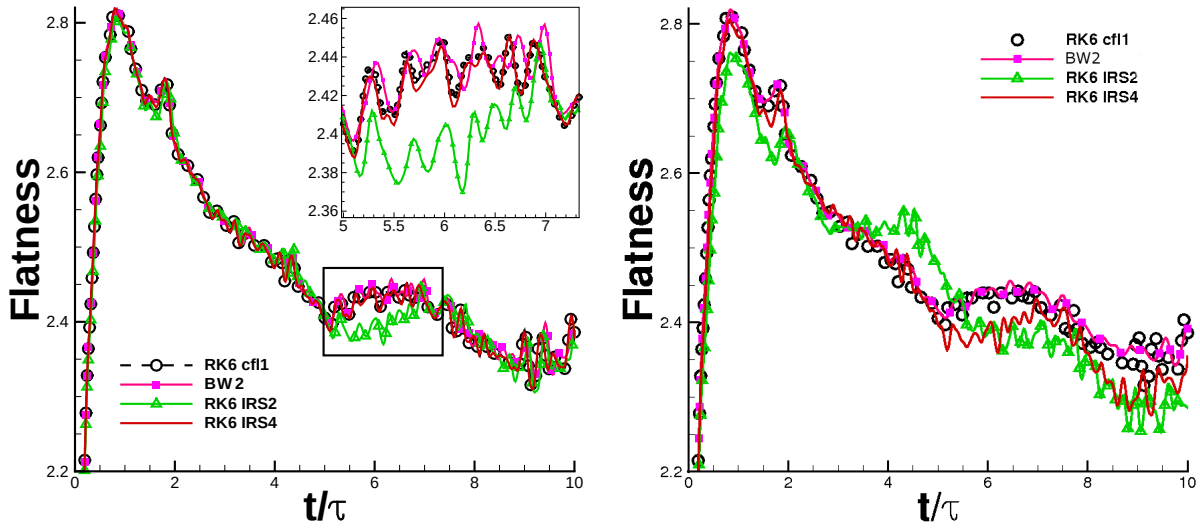
(a) $CFL = 5$. (b) $CFL = 10$.

Figure 10. **Comparison of the time evolution of the p_{rms} obtained with RK6 at $CFL = 1$, RK6IRS2, RK6IRS4, and BW2 schemes with $CFL = 5$ and 10.**



(a) $CFL = 5$. (b) $CFL = 10$.

Figure 11. **Comparison of the time evolution of the skewness of velocity fluctuations obtained with RK6 at $CFL = 1$, RK6IRS2, RK6IRS4 and BW2 schemes at $CFL = 5$ and 10.**



(a) $CFL = 5$. (b) $CFL = 10$.

Figure 12. Comparison of the time evolution of the flatness of velocity fluctuations obtained with RK6 at $CFL = 1$, RK6IRS2, and RK6IRS4 and BW2 schemes at $CFL = 5$ and 10.

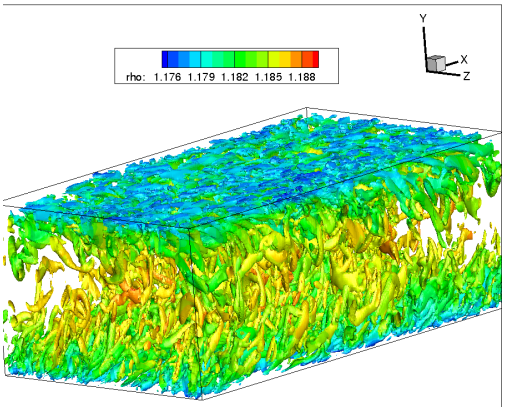
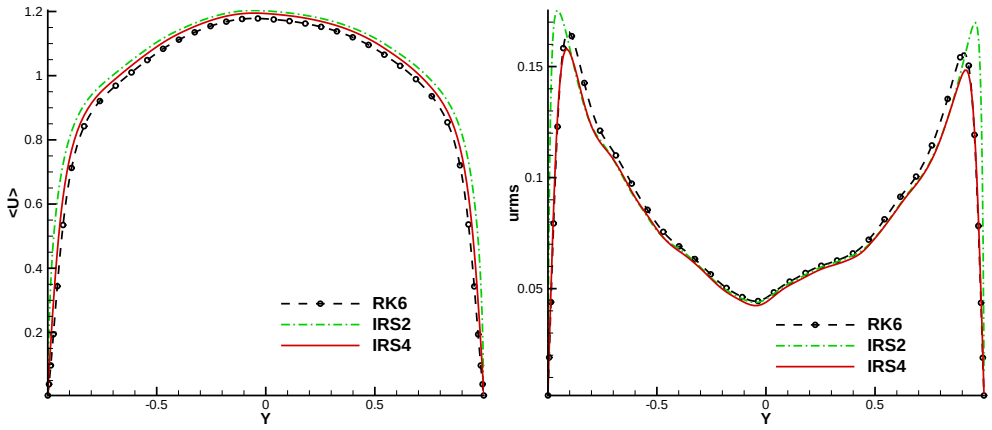
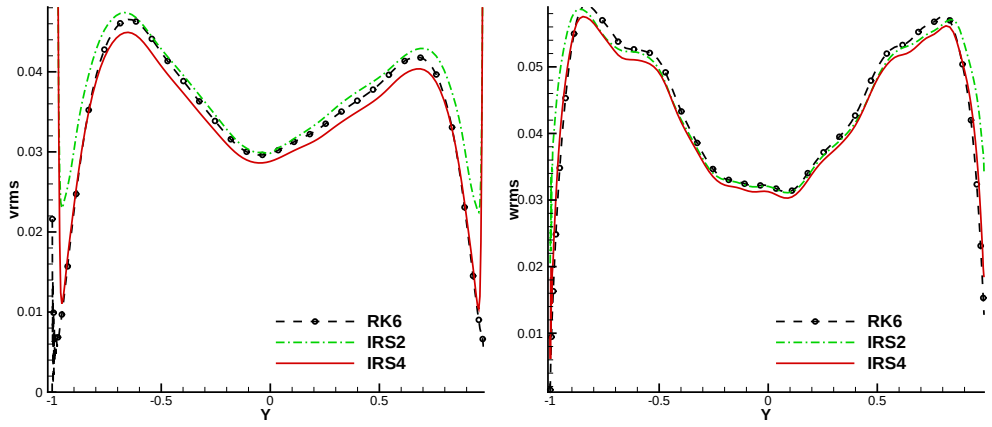


Figure 13. Turbulent channel flow. Iso-surface of λ_2 colored by ρ. Computed with FE9 RK6 on a 64x150x64 mesh



(a) $< U > /U_b$. (b) u_{rms}/U_b.

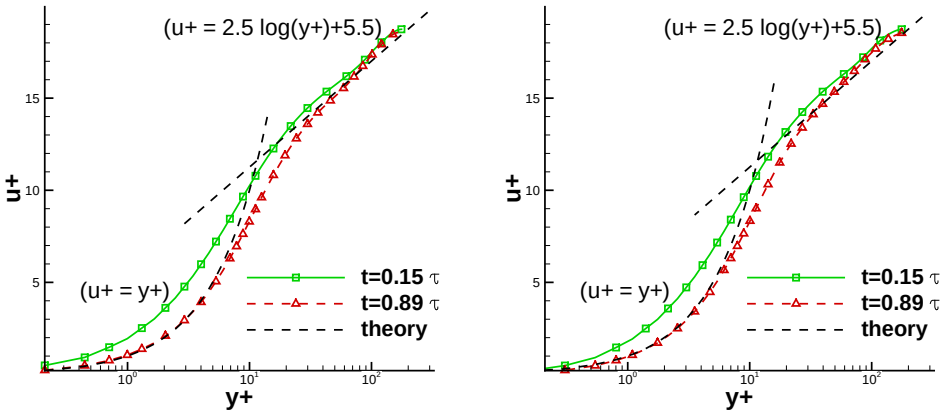
Figure 14. **Comparison of the mean velocity and u_{rms} profiles obtained at $t = 0.15t_0$ with RK6 at CFL=1 and RK6IRS schemes at CFL= 5.**



(a) v_{rms}/U_b. (b) w_{rms}/U_b.

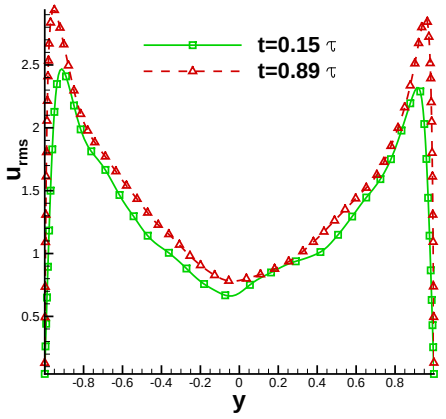
Figure 15. **Comparison of the v_{rms} and w_{rms} profiles obtained at $t = 0.15t_0$ with RK6 at CFL=1 and RK6IRS schemes at CFL= 5.**

Figure (16) shows the statistics of the velocity field at two different simulation times for the RK6IRS4 scheme. If the average streamwise velocity profile seems almost converged, the other quantities are still changing considerably. This will be improved in the final version.

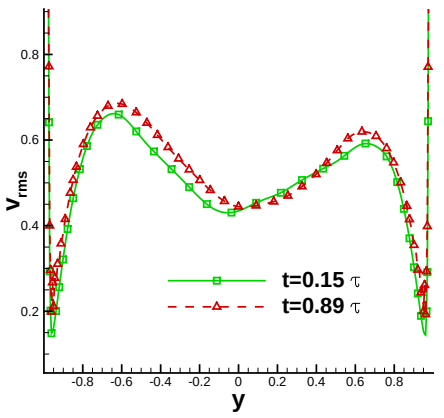


(a) $u+$ at the bottom.

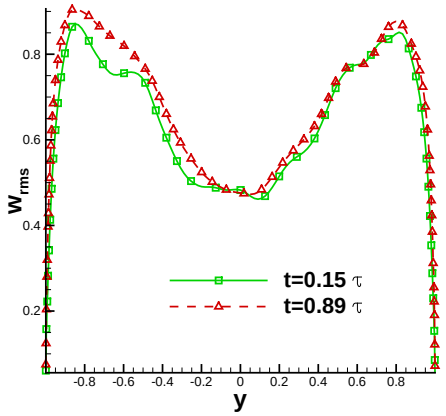
(b) $u+$ at the top.



(c) $\frac{u_{rms}}{u_\tau}$.



(d) $\frac{v_{rms}}{u_\tau}$.



(e) $\frac{w_{rms}}{u_\tau}$.

Figure 16. Comprarison of the computed $\dot{()}+$ profiles over $t = 0.15t_0$ and $t = 0.89t_0$. RK6IRS4 at CFL=5.

VI. Conclusions

In the present work, we assessed different implicit time integration techniques for turbulent compressible flow simulations. Precisely, a second-order backward (BW2) scheme, solved at each time step by means of inexact Newton subiterations was compared to classical Runge-Kutta (RK) schemes with second-order central implicit residual smoothing (IRS) and to a new fourth-order implicit residual smoothing scheme. The last one was introduced to palliate to the large dispersive errors introduced by standard second order implicit residual smoothing. Numerical experiments were carried out for a 2D linear advection problem, and for slightly compressible turbulent flows, namely, decaying homogenous isotropic turbulence and a turbulent channel flow simulation.

The results show that, while the BW2 introduces large phase and amplitude errors already for relatively small CFL numbers (higher than 5), and exhibits a huge computational cost due to the number of inner subiteration required to converge the Newton algorithm when using high-order spatial discretization schemes with large stencils, the IRS schemes imply only a moderate overcorst with respect to explicit RK schemes and exhibit a better accuracy than BW2. In particular, the proposed higher-order IRS scheme, obtained by smoothing the residual with a bilaplacian operator, introduces only very small additional errors up to CFL numbers comprised between 5 and 10, for an overcost that remains similar to that of the standard IRS2 scheme, thanks to the efficiency of the pentadiagonal solver used to invert the operator at each RK stage.

As a consequence, IRS4 is identified as a promising implicit time integration scheme for compressible turbulence computations.

Additional numerical experiments for highly compressible turbulent flows will be carried out in the near future for a more complete assessment of the implicit methods.

Acknowledgement

HPC resources were provided by IDRIS under the allocation 2013-i20132a7085 made by GENCI.

References

[1]Rizzetta, D. P., Visbal, M. R., and Blaisdell, G. E., "A time-implicit high-order compact differencing and filtering scheme for large-eddy simulation," *International Journal for Numerical Methods in Fluids*, Vol. 42, 2003, pp. 665–693.

[2]Grimich, K., Cinnella, P., and Lerat, A., "Spectral properties of high-order residual-based compact schemes for unsteady compressible flows," *Journal of Computational Physics*, Vol. 252, Nov. 2013, pp. 142–162.

[3]Persson, P.-O., "High-Order LES Simulations using Implicit-Explicit Runge–Kutta Schemes," *AIAA Paper 2011-684*, 2011.

[4]Visbal, M. R. and Rizzetta, D. P., "Large-Eddy Simulation on Curvilinear Grids Using Compact Differencing and Filtering Schemes," *Journal of Fluids Engineering*, Vol. 124, 2002, pp. 836–847.

[5]Pino Martin, M. and Candler, G., "A parallel implicit method for the direct numerical simulation of wall-bounded compressible turbulence," *Journal of Computational Physics*, Vol. 215, 2006, pp. 153–171.

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

[7]Jameson, A., Schmidt, W., and Turkel, E., ""Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping"," AIAA Paper 81-1259, 1981.

[8]Lerat, Sidès, and Daru, Lect. Notes in Phys., 170*, pp. 343-349*, 1982.

[9]Jameson, A. and Baker, T., ""Solution of the Euler Equations for Complex Configurations"," AIAA 6th Computational Fluid Dynamics Conference, Denver, Juillet 1983.

[10]Blazek, J., Kroll, N., and Rossow, C., "'A comparison of several implicit residual smoothing methods"," *ICFD Conference on Numerical Methods for Fluid Dynamics*, 1992.

[11]Haeltermana, R., Vierendeels, J., and Van Heule, D., "Optimization of the Runge–Kutta iteration with residual smoothing," *Journal of Computational Physics*, Vol. 234, 2010, pp. 253–271.

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

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

[14]Kim, J. W., "Quasi-disjoint pentadiagonal matrix systems for the parallelization of compact finite-difference schemes and filters," *Journal of Computational Physics*, Vol. 241, 2013, pp. 168–194.

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

[16]Bogey, C. and Bailly, C., "A family of low dispersive and low dissipative explicit schemes for flow and noise computations," *Journal of Computational Physics*, Vol. 194, No. 1, Feb. 2004, pp. 194–214.

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

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

[19]Lerat, A. and Corre, C., *Approximations dordre élevé pour les écoulements compressibles*, Ecole de Printemps de mécanique des fluides, Fréjus, France, 2003.

[20]Lerat, A. and Corre, C., "Higher order residual-based compact schemes on structured grids, chapter CFD-higher order discretization methods," *Comput. Fluid Dyn. Course, von Karman Institute for Fluid Dynamics. VKI LS 2006-1*, 2006, pp. 1–111.

[21]Ducros, F., Ferrand, V., Nicoud, F., Weber, C., Darracq, D., Gacherieu, C., and Poinsot, T., "Large-Eddy Simulation of the Shock/Turbulence Interaction," *Journal of Computational Physics*, Vol. 152, 1999, pp. 517–549.

[22]Saunier, O., *Mthode d'adaptation de maillages cartsiens base sur des schmas d'ordre lev pour les quations d'Euler d'un fluide compressible. Application aux pales de rotor d'hlicoptre.*, Ph.D. thesis, 2008.

[23]Lambert, J. D., "Numerical methods for ordinary differential systems: the initial value problem," *J. Wiley edition*, 1991.

[24]Dembo, R. e., "Inexact Newton Methods," *SIAM J. Num. Anal.*, Vol. 19, 1982, pp. 400–408.

[25]Eisenstat, S. and Walker, H., "Choosing the forcing terms in a inexact Newton method," *SIAM J. Sci. Stat. Comput.*, Vol. 17, 1996, pp. 16–32.

[26]Kloczko, T., Corre, C., and Beccantini, A., "Low-cost implicit schemes for all-speed flows on unstructured meshes," *International Journal for Numerical Methods in Fluids*, Vol. 58, No. 5, 2008, pp. 493–526.

[27]Hollanders, Lerat, and Peyret, *AIAA J., 23, pp 1670-1678*, 1985.

[28]Swanson, R. C. and Turkel, E., ""Multistage Schemes With Multigrid for Euler and Navier–Stokes Equations" ," NASA TP 3631, August 1997.

[29]""Unsteady Euler and Navier-Stokes flows simulations with an implicit Runge-Kutta method"," *Computational Fluid Dynamics'94*, J. Wiley, 1994, 1994, pp. 917–924, 2nd European Computational Fluid Dynamics Conference (ECCOMAS), Stuttgart, Allemagne, Septembre 1994.

[30]Samtaney, R., Pullin, D., and Kosovic, B., "Direct numerical simulation of decaying compressible turbulence and shocklet statistics," *Physics of Fluids*, Vol. 13, No. 5, 2001, pp. 1415–1430.

[31]Kim, J., M. P. and Moser, R., "Turbulence statistics in fully developed channel flow at low Reynolds number," *Journal of Fluid Mechanics*, Vol. 177, 1987, pp. 133–166.