INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2007; **55**:815–865
Published online 17 April 2007 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1494

Fifth-order Hermitian schemes for computational linear aeroacoustics

G. Capdeville*, †, ‡

Laboratoire de Mecanique des fluides, Ecole Centrale de Nantes, 1, rue de la Noë, B.P. 92101, 44321 Nantes Cedex 3, France

SUMMARY

We develop a class of fifth-order methods to solve linear acoustics and/or aeroacoustics. Based on local Hermite polynomials, we investigate three competing strategies for solving hyperbolic linear problems with a fifth-order accuracy. A one-dimensional (1D) analysis in the Fourier series makes it possible to classify these possibilities. Then, numerical computations based on the 1D scalar advection equation support two possibilities in order to update the discrete variable and its first and second derivatives: the first one uses a procedure similar to that of Cauchy–Kovaleskaya (the 'Δ-P5 scheme'); the second one relies on a semi-discrete form and evolves in time the discrete unknowns by using a five-stage Runge–Kutta method (the 'RGK-P5 scheme'). Although the RGK-P5 scheme shares the same local spatial interpolator with the Δ-P5 scheme, it is algebraically simpler. However, it is shown numerically that its loss of compactness reduces its domain of stability.

Both schemes are then extended to bi-dimensional acoustics and aeroacoustics.

Following the methodology validated in (*J. Comput. Phys.* 2005; **210**:133–170; *J. Comput. Phys.* 2006; **217**:530–562), we build an algorithm in three stages in order to optimize the procedure of discretization.

In the 'reconstruction stage', we define a fifth-order local spatial interpolator based on an upwind stencil. In the 'decomposition stage', we decompose the time derivatives into simple wave contributions.

In the 'evolution stage', we use these fluctuations to update either by a Cauchy–Kovaleskaya procedure or by a five-stage Runge–Kutta algorithm, the discrete variable and its derivatives.

In this way, depending on the configuration of the 'evolution stage', two fifth-order upwind Hermitian schemes are constructed.

The effectiveness and the exactitude of both schemes are checked by their applications to several 2D problems in acoustics and aeroacoustics. In this aim, we compare the computational cost and the computation memory requirement for each solution.

The RGK-P5 appears as the best compromise between simplicity and accuracy, while the Δ-P5 scheme is more accurate and less CPU time consuming, despite a greater algebraic complexity. Copyright © 2007 John Wiley & Sons, Ltd.

Received 11 September 2006; Revised 15 February 2007; Accepted 2 March 2007

*Correspondence to: G. Capdeville, Laboratoire de Mécanique des Fluides, Ecole Centrale de Nantes, 1, rue de la Noë, B.P. 92101, 44321 Nantes Cedex 3, France.
†E-mail: guy.capdeville@ec-nantes.fr
‡Undergraduate Professor.

KEY WORDS: Hermitian schemes; upwind discretization; linear wave modelling; acoustics; aeroacoustics; non-reflective boundary conditions; linear hyperbolic systems

1. INTRODUCTION

In a previous work, we constructed a third-order Hermitian scheme for linear wave propagation (the Δ-P3 scheme'). By comparing this scheme with a competing finite-difference method, we demonstrated that the use of a reduced stencil makes it possible to obtain good numerical performances. The compactness in time is ensured by a Cauchy–Kovaleskaya procedure while the compactness in space uses the calculation of a local Hermite interpolator in conjunction with an upwind discretization. With the same order of accuracy, a reduced numerical stencil generates lower amplitude and phase errors. Furthermore, the treatment of the boundary conditions becomes more natural.

In the acoustic context, the number of cells per wavelength (CPW) to maintain dissipation and dispersion within acceptable bounds is a very relevant characteristic. In [1], we demonstrated that the Δ-P3 scheme only needs 15 CPW to reach an accuracy of 0.01% of error in phase and in amplitude. With such a result, the Δ-P3 scheme appears to be more advantageous (both in CPU time and accuracy) than classical finite-difference methods, even simpler formally. Such a scheme is well adapted for propagating an acoustic perturbation over long time periods.

For flow problems with sophisticated structures, high resolution becomes necessary so that structural information is correctly extracted. Computational aeroacoustics (CAA) is an area where high-resolution schemes are highly demanded for the simulation of waves with large spectral bandwidth and amplitude disparity. For example, for an imperfectly expanded supersonic jet, the Strouhal number ranges from 10^{-2} to 10^{1} [2]. The velocity fluctuations of the radiated sound wave can be four orders of magnitude smaller than that of the mean flow. The Δ-P3 scheme makes it possible to model such a kind of problem with a sufficient realism (see [3]); however, it necessitates the use of a large mesh. High-resolution schemes can be used to alleviate the demanding on a dense mesh for problems with large characteristic scales disparity.

There are many approaches to obtain high resolution in a numerical computation.

The low-dissipation and low-dispersion Runge–Kutta (LDDRK) scheme [4], the DRP scheme [5], and its recent improvements [6, 7], compact schemes [8, 9], spectral methods [10, 11], ENO-WENO schemes [12–14], Hermitian schemes [15] or the discontinuous Galerkin method [16], are all viable examples.

These schemes are among the best for many hyperbolic problems. However, the compactness of the discretization is not always ensured: natural treatment of the boundary conditions and the extension to non-uniform meshes become more difficult then. Moreover, these schemes often use a spatially centred stencil: specific terms must then be added in order to damp any non-physical wave ('spurious wave') generated by boundary conditions and/or round-off errors.

The goal of this paper is to develop a competing fifth-order method for linear CAA. Such a method must preserve the compactness of the discretization while necessitating less than 8 CPW to reach an accuracy of 0.01% of error in phase and in amplitude (roughly the performances of the DRP scheme [5], see [1] for a detailed analysis). For this purpose, we rely on the algorithm developed to generate the Δ-P3 scheme [1, 3].

DOI: 10.1002/fld

Consequently, the problem is broken up into three sub-problems, namely:

- a 'reconstruction stage';
- a 'decomposition stage'; and
- an 'evolution stage'.

In the stage of reconstruction, we use the concept of 'local spatial interpolator' in order to produce a compact Hermite polynomial, approaching the solution on a given discrete cell. Based on 1D simulations, we experiment two different strategies to generate a fifth-order polynomial. The best solution is then selected and extended to 2D non-uniform Cartesian meshes.

The stage of decomposition is based on the ideas suggested by Roe [17], and the solutions validated in [1, 3]. Indeed, according to the physical nature of the problem considered, we decompose the time derivatives into four simple waves (acoustics) [1], or six simple waves (aeroacoustics) [3]. We impose the directions of propagation arbitrarily, free of the local data or the grid directions.

With this approximation, the resulting discrete model remains linear and produces a multidimensional decomposition of the time derivatives. In this way, each simple wave generates a time fluctuation and this fluctuation is sent towards the downwind nodes in order to evolve in time the discrete variables.

Lastly, the stage of evolution employs these time fluctuations either in a procedure of Cauchy–Kovaleskaya (the 'Δ-P5 scheme') or in a five-stage Runge–Kutta algorithm (the 'RGK-P5 scheme'). The Cauchy–Kovaleskaya method simultaneously approaches the space and temporal derivatives. Thus, no intermediate semi-discrete form is generated and the numerical scheme is put in a more compact form. As demonstrated in [1, 3], such a solution produces a very competing method since the third-order version obtained compares favourably with a higher-order one such as the DRP scheme, [5].

On the other hand, the Runge–Kutta algorithm produces a discrete form algebraically simpler. This property becomes a significant advantage as soon as the governing equations become complicated or when 3D calculations are considered. However, the compactness in time is definitively lost.

Extensive numerical computations make it possible to compare the advantages and the drawbacks of both methods.

This process in three stages provides a modular method of which each stage can be optimized distinctly. Thus, this makes it possible to generate two alternative solutions for the decomposition stage.

Moreover, this approach allows the effective numerical treatment of boundary conditions. Indeed, by putting at the zero only time fluctuations brought by the waves of entry to a given boundary, there is no need to modify neither the equations nor the algorithm at the boundary points. It is the advantage of using an upwind method, which clearly identifies waves entering and outgoing the computational domain.

The organization of this paper is as follows: Section 2 presents the basic choices for generating a fifth-order algorithm that discretizes the 1D scalar advection equation. For this purpose, we develop three fifth-order versions: the Δ-P5 and the RGK-P5 schemes with, as unknowns, the primitive variables and its first and second space derivatives and the Δ-P5NC scheme with only the primitive variable and its first space derivative as unknowns. We employ a Fourier analysis in order to calculate the errors of dissipation and dispersion for each numerical scheme. Then, we

DOI: 10.1002/fld

provide numerical examples to compare the behaviour of the three schemes in terms of CPU time and computation memory cost. In Section 3, we extend the Δ-P5 and the RGK-P5 schemes to linear hyperbolic systems, following the procedures developed in [1, 3]. We propose two solutions for the evolution stage in order to obtain the algebraical expressions that characterize the Δ-P5 and the RGK-P5 scheme. The boundary conditions are briefly detailed, according to the principles explained in [1, 3].

Section 4 presents extensive numerical simulation in order to validate the boundary conditions and to compare both schemes. The CPU cost and the computation memory cost are emphasized.

To finish, we present a summary and general conclusions in Section 5.

The algebraic formulae for calculating the derivatives of the Δ-P5 or the RGK-P5 scheme on a non-uniform Cartesian grid are presented in Appendix A.

2. FIFTH-ORDER UPWIND HERMITIAN SCHEMES: ONE-DIMENSIONAL CASE

We consider the linear advection equation:

$$u_t + au_x = 0 \quad (a \equiv Cte > 0) \tag{1}$$

We define a uniform grid with mesh size $\Delta x \equiv x_{i+1} - x_i$ and time step size $\Delta t \equiv t_{n+1} - t_n$. Then, we define the CFL (Courant number), subsequently noted v, by the following formula: $v \equiv a\Delta t/\Delta x$. The problem being linear, every derivative of u remains solution of the advection operator.

In this section, we investigate three solutions in order to discretize (1) on a compact stencil with a fifth-order accuracy.

First, to get a fifth-order accuracy, we need to calculate the following polynomial, $\widetilde{u}(x)$, that interpolates the numerical data on a given discrete stencil at the time $t = t_n$:

$$\widetilde{u}(x) \equiv c_1 + c_2(x - x_i) + c_3(x - x_i)^2 + c_4(x - x_i)^3 + c_5(x - x_i)^4 + c_6(x - x_i)^5 \tag{2}$$

To begin, we describe the construction of the Δ-P5 scheme in order to discretize (1). We define such a scheme by the choice of the biased stencil $[i-1,\ i]$ and by the following discrete unknowns: $(u_i^n \equiv \widetilde{u}(x_i, t_n),\ r_i^n \equiv \Delta x \times \widetilde{u}_x(x_i, t_n),\ w_i^n \equiv \Delta x^2 \times \widetilde{u}_{xx}(x_i, t_n))$, defined on each grid point, x_i, and for each discrete time, $t_n \equiv n\Delta t$ (Figure 1).

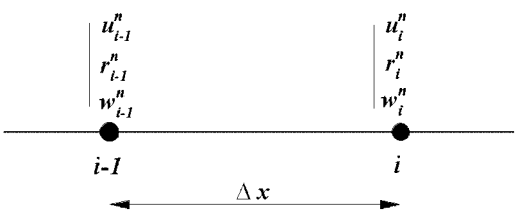

Figure 1. Δ-P5 scheme: discrete unknowns on the cell $[i-1, i]$ for the discretization of the scalar advection equation $u_t + au_x = 0$ $(a>0)$.

DOI: 10.1002/fld

2.1. The 'Δ-P5 scheme'

With the choices defining the Δ-P5 scheme, the expansion coefficients, $\{c_i\}_{i=1,2,3,4,5,6}$, are obtained from the known data on the grid, by the following relations:

$$\begin{aligned} \widetilde{u}(x_j) &\equiv u_j^n \\ \Delta x \times \widetilde{u}_x(x_j) &\equiv r_j^n \quad \text{for } j \in \{i-1, i\} \\ \Delta x^2 \times \widetilde{u}_{xx}(x_j) &\equiv w_j^n \end{aligned} \tag{3}$$

These relations generate a linear system of six equations for the six unknowns $\{c_i\}_{i=1,2,3,4,5,6}$.

This system is inverted to obtain

$$\begin{aligned} c_1 &= u_i^n \\ c_2 &= \frac{r_i^n}{\Delta x} \\ c_3 &= \frac{w_i^n}{2\Delta x^2} \\ c_4 &= -\frac{20(u_{i-1}^n - u_i^n) + 12r_i^n + 8r_{i-1}^n + w_{i-1}^n - 3w_i^n}{2\Delta x^3} \approx \frac{1}{6}u_{3x}|_i^n \\ c_5 &= -\frac{30(u_{i-1}^n - u_i^n) + 16r_i^n + 14r_{i-1}^n + 2w_{i-1}^n - 3w_i^n}{2\Delta x^4} \approx \frac{1}{24}u_{4x}|_i^n \\ c_6 &= -\frac{12(u_{i-1}^n - u_i^n) + 6(r_i^n + r_{i-1}^n) + w_{i-1}^n - w_i^n}{2\Delta x^5} \approx \frac{1}{120}u_{5x}|_i^n \end{aligned}$$

Then, by using this result and the Cauchy–Kovaleskaya procedure detailed in [1], we get the algebraical formula that discretizes (1) with a fifth-order accuracy both in time and space:

$$\begin{aligned} u_i^{n+1} &= u_i^n + v^3(6v^2 - 15v + 10)(u_{i-1}^n - u_i^n) + v(3v+1)(v-1)^3 r_i^n \\ &\quad + v^3(v-1)(3v-4)r_{i-1}^n - \frac{v^2}{2}(v-1)^3 w_i^n + \frac{v^3}{2}(v-1)^2 w_{i-1}^n \\ r_i^{n+1} &= -30v^2(v-1)^2(u_{i-1}^n - u_i^n) + \frac{v}{2}(5v-2)(v-1)^2 w_i^n - \frac{v^2}{2}(5v-3)(v-1)w_{i-1}^n \\ &\quad - v^2(3v-2)(5v-6)r_{i-1}^n - (5v+1)(3v-1)(v-1)^2 r_i^n \\ w_i^{n+1} &= 60v(2v-1)(v-1)(u_{i-1}^n - u_i^n) - (v-1)(10v^2 - 8v + 1)w_i^n \\ &\quad + v(10v^2 - 12v + 3)w_{i-1}^n + 12v(5v-2)(v-1)r_{i-1}^n + 12v(5v-3)(v-1)r_i^n \end{aligned} \tag{4}$$

This algebraic system defines the 'Δ-P5 scheme'. This way, we produced a fifth-order upwind Hermitian scheme.

DOI: 10.1002/fld

Decomposing the discrete solution, $U_i^n \equiv [u_i^n, r_i^n, w_i^n]^t$ in Fourier series ($U_i^n = \hat{U}(t_n) \times e^{jkx_i}$, $j^2 = -1$, k: wavenumber) and substituting in (4), we get the following relation:

$$\hat{U}(t_{n+1}) = G(\beta, \nu) \times \hat{U}(t_n) \quad (\beta \equiv k \times \Delta x : \text{phase angle})$$

The complex 'amplification matrix', $G(\beta, \nu)$, possesses three distinct eigenvalues. The first eigenvalue, noted $\lambda_1(\beta, \nu)$, is defined as the 'accurate eigenvalue' and approximates $e^{-j\nu\beta}$, the amplification factor of the exact solution of (1).

The two remaining eigenvalues, $\lambda_{2,3}$ (β, ν), will be referred to as 'spurious eigenvalues' (see [1] for more details). Then, we can decompose the complex amplitude of the solution in the following way:

$$\underbrace{\hat{U}(t_{n+1})}_{\substack{\text{num. sol.} \\ \text{at } t=t_{n+1}}} = \underbrace{\lambda_1 \times \hat{U}_1(t_n)}_{\text{'accurate' sol.}} + \underbrace{\lambda_2 \times \hat{U}_2(t_n)}_{\text{'spurious' sol.}} + \underbrace{\lambda_3 \times \hat{U}_3(t_n)}_{\text{'spurious' sol.}} \tag{5}$$

The spurious components of the solution must be damped in order to prevent the development of numerical inaccuracies or instabilities. Therefore, the relation $|\lambda_{2,3}(\beta, \nu)| \leqslant 1 \ \forall \nu \leqslant 1, \ \forall \beta$, must be verified for such a scheme.

Practically, the accurate eigenvalue is dissociated from the spurious ones, by using the property of consistency: $\lambda_1(\beta, \nu) \underset{\beta \to 0}{\to} 1 \ \forall \nu \in [0, 1]$. These eigenvalues are calculated by using the Maple symbolic computer package.

To study the stability and accuracy of the Δ-P5 scheme, let us plot $|\lambda_{1,2,3}(\beta, \nu)| = f(\beta)$ or $f(N)$, for ν fixed.

As in [1], N is defined as the number of CPW ($N \times \beta = 2\pi$) required to maintain dissipation error, $(1 - |\lambda_1(\beta, \nu)|)$ and dispersion error, $(\text{Arg}(\lambda_1(\beta, \nu))/\nu\beta - 1)$, within acceptable bounds.

As a relevant target for accuracy (see [1] for a detailed discussion on this point), we assume that the position of a wave (quantified by the phase error per time step and its amplitude error per time step), is required to within 0.01% for N below the value of 8.

Figure 2 plots the amplitude of the accurate and spurious eigenvalues, with the value $\nu = 3/4$ for which the amplitude error is maximum.

At the highest frequency resolvable by the mesh, $\beta = \pi$ (2 CPW), the accurate component of the numerical solution is attenuated with a minimum value for $|\lambda_1|$ ($|\lambda_1| = 0.989$). In a similar way, the damping of the two spurious components by the eigenvalues $\lambda_{2,3}$, appears in an obvious way since $|\lambda_{2,3}| \leqslant 1$ everywhere.

To be quantitative, let us consider Figures 3(a) and (b), which shows amplitude and phase error trends for the Δ-P5 scheme.

The contours are plotted for several levels of dissipation and dispersion errors, in a plane that shows the number of CPW, N, *versus* the Courant number, ν.

As one can note, less than 5 CPW are necessary to reach the limit of 0.01% in the range $1/2 \leqslant \nu \leqslant 1$. This result is true both in amplitude and phase error. To compare, we plot in Figures 3(c) and (d), the results obtained with the FDo11p/RKo6s scheme [6]. This scheme is a recently improved version of the DRP scheme of Tam and Webb [5]: the FDo11p/RKo6s scheme is a fourth-order scheme and it uses a 11-points finite-difference stencil (FDo11p scheme) of which the coefficients are partially selected in order to minimize the dispersion error for a large range of wavenumbers. To ensure the time integration, a fourth-order six-stage Runge–Kutta algorithm

DOI: 10.1002/fld

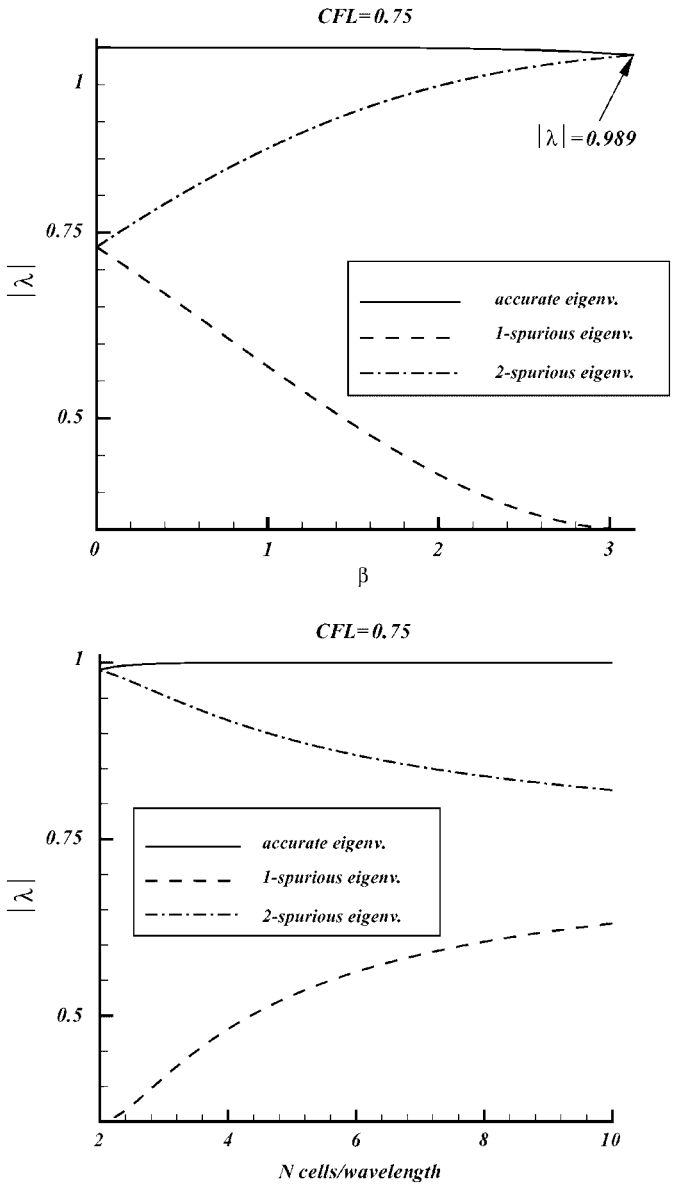

Figure 2. Δ-P5 scheme: accurate and spurious eigenvalues amplitudes ($v = 0.75$).

(RKo6s scheme) is constructed by optimizing the dissipation and the dispersion errors (see [6] for more details). As can be seen from Figures 3(c) and (d), the FDo11p/RKo6s scheme produces very good results for the amplitude error since it only necessitates 4 CPW to reach the limit of 0.01%, Figure 3(c). However, the results obtained for the phase error are less convincing since more than 5 CPW become necessary to lower the phase error under the limit of 0.01%, Figure 3(d): in such a

DOI: 10.1002/fld

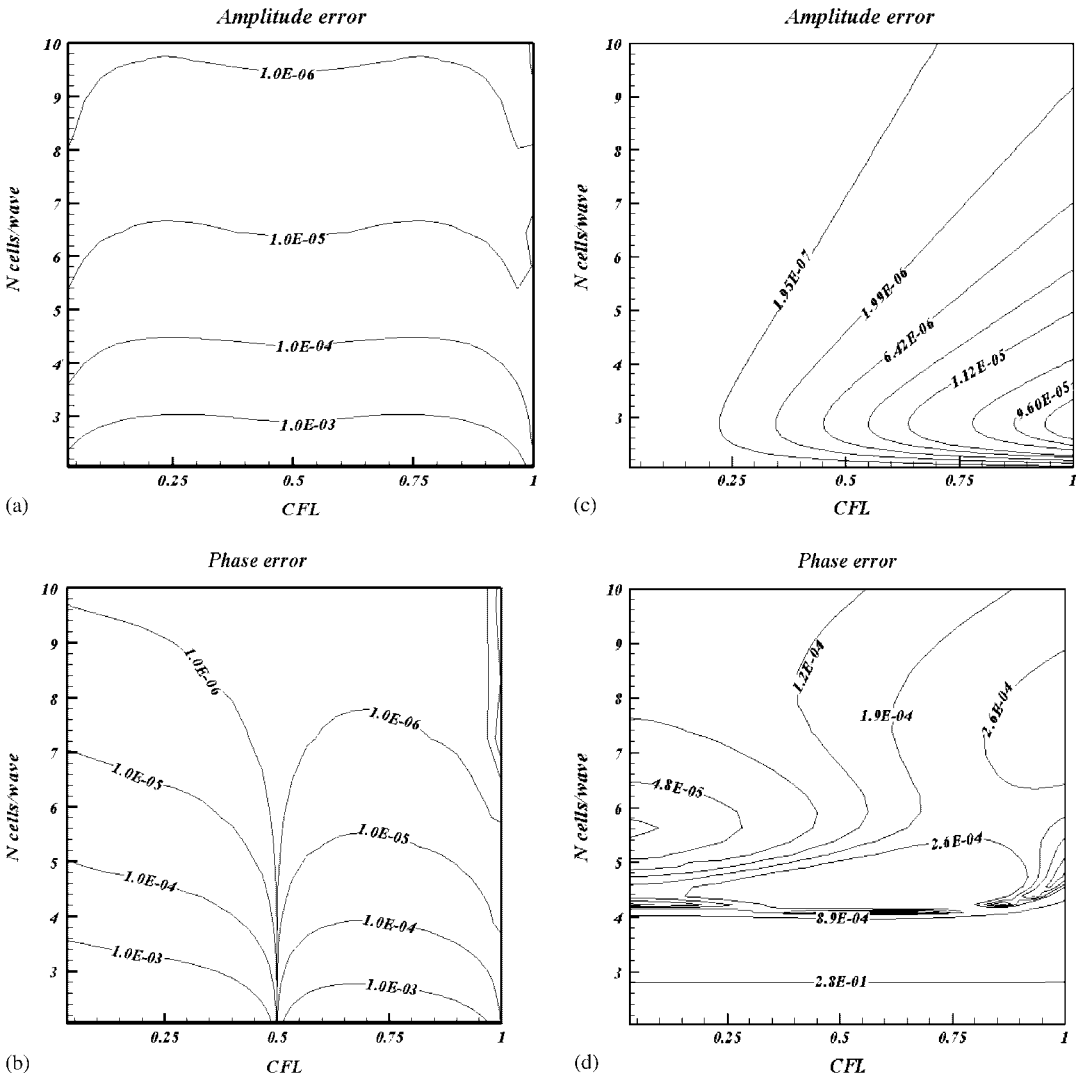

Figure 3. Amplitude and phase errors trends for the scalar advection equation $u_t + au_x = 0$. Δ-P5 scheme: (a) amplitude error; (b) phase error. FDo11p/RKo6s scheme: (c) amplitude error; (d) phase error.

case, the Δ-P5 scheme produces better results, Figure 3(b). The compactness of the discretization explains the advantage obtained by the Δ-P5 scheme in terms of phase error.

At first sight, however, the Δ-P5 scheme has two important drawbacks when it is compared with schemes that are more classical: an increased algebraic complexity and an important memory requirement since two additional variables are necessary at each discrete point.

A first solution to remedy these problems is to calculate (2) on the extended discrete stencil $[i-2, i-1, i]$ by only using the following discrete unknowns: $u_i^n \equiv \widetilde{u}(x_i, t_n),\ r_i^n \equiv \Delta x \times \widetilde{u}_x(x_i, t_n)$,

DOI: 10.1002/fld

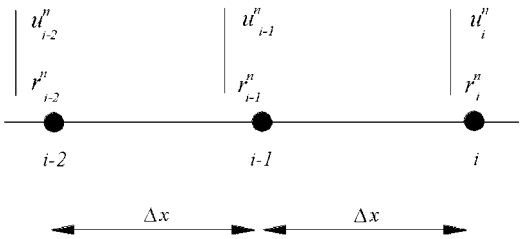

Figure 4. Δ-P5NC scheme: discrete unknowns on the cells $[i-2, i-1, i]$ for the discretization of the scalar advection equation $u_t + au_x = 0$ $(a>0)$.

Figure 4. Compared with the Δ-P5 scheme, this scheme has lost its compactness. For this reason, this scheme will be subsequently named the 'Δ-P5NC scheme'.

2.2. *The Δ-P5NC scheme*

By following the same procedure as the previous one, we generate the following algebraic system discretizing (1) on the stencil $[i-2, i-1, i]$:

$$\begin{aligned}
u_i^{n+1} &= -\tfrac{1}{4}\nu^2(\nu-2)(\nu-1)^2 r_{i-2}^n - \tfrac{1}{4}\nu\times(\nu-1)^2(\nu-2)^2 r_i^n \\
&\quad - \tfrac{1}{4}\nu^2(3\nu-7)(\nu-1)^2 u_{i-2}^n - \nu^2(\nu-1)(\nu-2)^2 r_{i-1}^n \\
&\quad + \tfrac{1}{4}(1+3\nu)(\nu-1)^2(\nu-2)^2 u_i^n + \nu^2(\nu-2)^2 u_{i-1}^n \\
r_i^{n+1} &= \tfrac{1}{4}\nu\times(\nu-1)(5\nu^2-11\nu+4) r_{i-2}^n + \tfrac{1}{4}(\nu-1)(\nu-2)(5\nu^2-9\nu+2) r_i^n \\
&\quad + \tfrac{1}{4}\nu\times(\nu-1)(\nu-2)(15\nu-7) u_{i-2}^n + \nu\times(\nu-2)(5\nu^2-10\nu+4) r_{i-1}^n \\
&\quad - \tfrac{1}{4}\nu\times(\nu-1)(\nu-2)(15\nu-23) u_i^n - 4\nu\times(\nu-1)(\nu-2) u_{i-1}^n
\end{aligned} \tag{6}$$

Although the algebraic complexity remains important, such a scheme makes it possible to decrease the computer memory requirement with, in theory, the same accuracy as the Δ-P5 scheme.

To verify this, let us plot the eigenvalues of the complex amplification matrix of this scheme.

Figure 5 plots the amplitude of the accurate and spurious eigenvalues, for $\nu = 3/4$. As can be seen, $|\lambda_1| = 0.990$ for $\beta = \pi$: this result is comparable with the one obtained with the Δ-P5 scheme. One can also note that the modulus of the spurious eigenvalue remains everywhere less than one.

However, Figure 6 shows that more than 6 CPW are needed in order to reach the limit of 0.01%: although the theoretical accuracy is unchanged, the numerical characteristics of the Δ-P5NC scheme are slightly deteriorated. The weaker compactness explains this discrepancy.

Nevertheless, the numerical performances of such a scheme remain at a very high level. Therefore, the Δ-P5NC scheme can be considered as a viable alternative to the Δ-P5 scheme as far as linear wave propagation is concerned.

Lastly, the third alternative to generate a fifth-order scheme with a reduced stencil is to replace the Cauchy–Kovaleskaya procedure by a Runge–Kutta algorithm. By doing so, the algebraic complexity of the resulting scheme is decreased but the compactness in time is lost.

DOI: 10.1002/fld

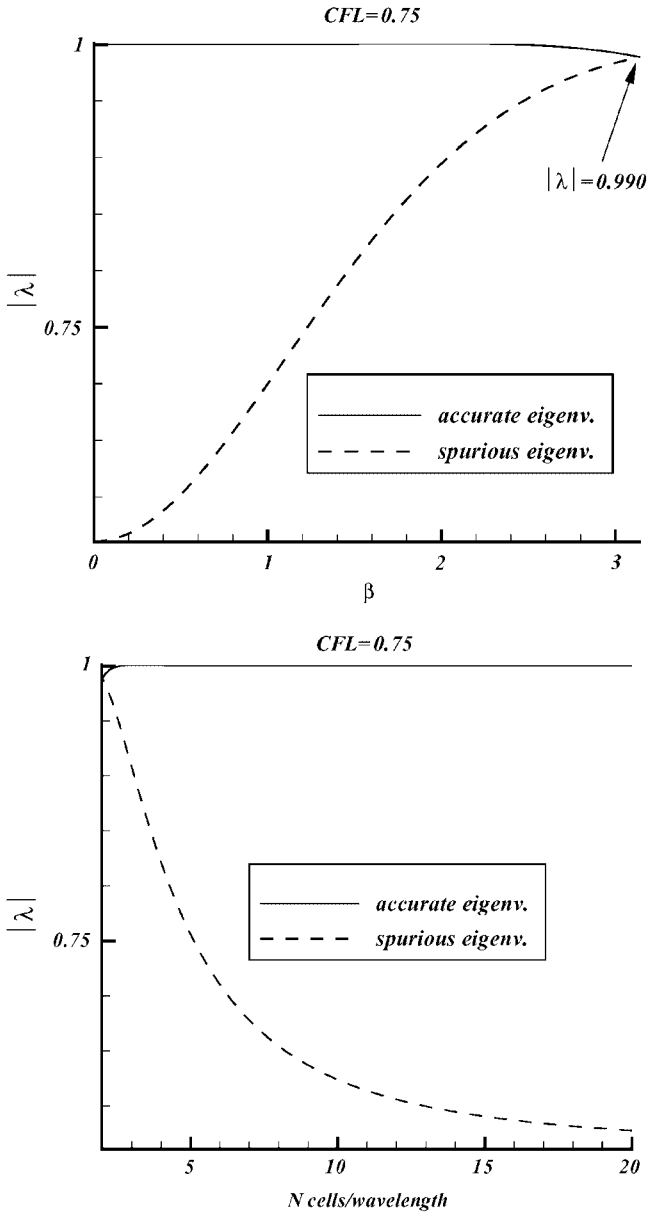

Figure 5. Δ-P5NC scheme: accurate and spurious eigenvalues amplitudes ($v = 0.75$).

In order to maintain a fifth-order accuracy in space and to calculate (2), we select as discrete unknowns $u_i^n,\ r_i^n,\ w_i^n$ on the discrete stencil $[i-1, i]$. Integration in time is then ensured by a classical five stages Runge–Kutta method. We call the resulting scheme the 'RGK-P5 scheme'.

DOI: 10.1002/fld

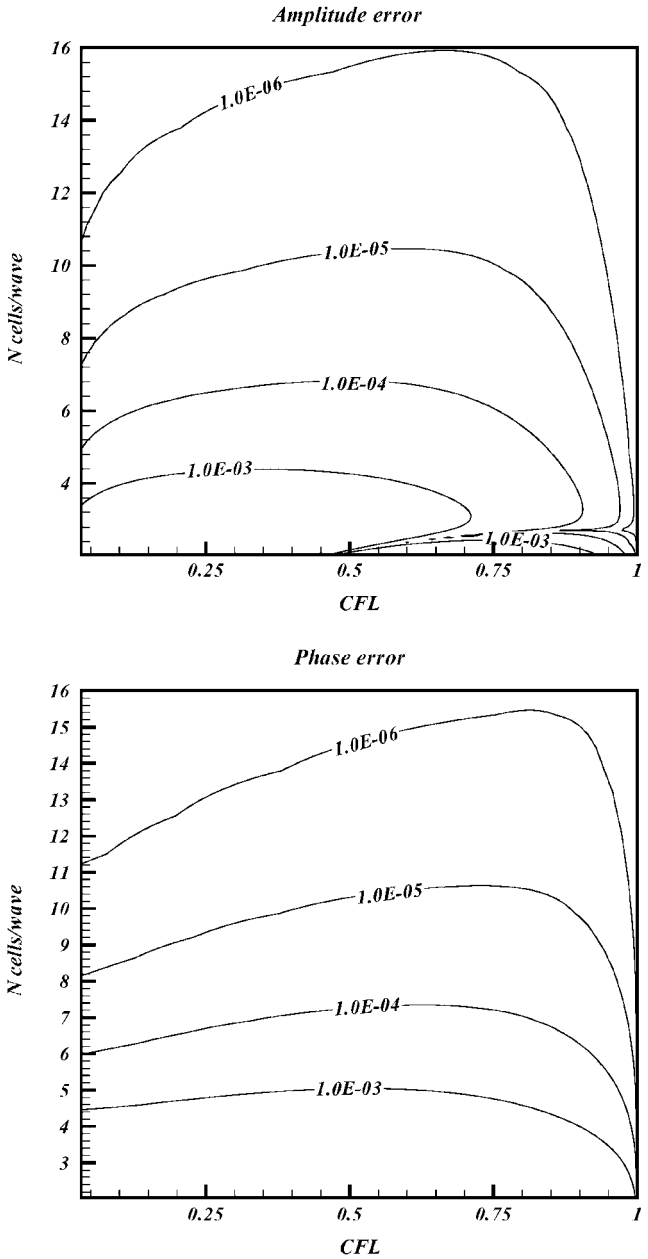

Figure 6. Δ-P5NC scheme: amplitude and phase errors trends for the scalar advection equation $u_t + au_x = 0$.

2.3. The RGK-P5 scheme

Since we use the same discrete unknowns on the same stencil, the polynomial $\widetilde{u}(x)$ is identical to the one defining the Δ-P5 scheme. Then, we use this polynomial to calculate the spatial derivatives arising in the Runge–Kutta process.

DOI: 10.1002/fld

Thus, the resulting procedure discretizing (1) with a fifth-order accuracy is the following:

$$\begin{aligned} u_i^{(k)} &= u_i^n - \alpha_k \times v \times r_i^{(k-1)} \\ r_i^{(k)} &= r_i^n - \alpha_k \times v \times w_i^{(k-1)}, \quad k \in \{1, \ldots, 5\} \\ w_i^{(k)} &= w_i^n + \alpha_k [60v \times (u_{i-1}^{(k-1)} - u_i^{(k-1)}) + 12v \times (2r_{i-1}^{(k-1)} \\ &\quad + 3r_i^{(k-1)}) + 3v \times (w_{i-1}^{(k-1)} - 3w_i^{(k-1)})] \end{aligned} \tag{7}$$

with the following definitions:

$$\alpha_k \equiv \frac{1}{(6-k)}, \quad \varphi_i^{(0)} \equiv \varphi_i^n, \quad \varphi_i^{n+1} \equiv \varphi_i^{(5)}, \quad \varphi \equiv \begin{cases} u \\ r \\ w \end{cases}$$

Compared with the previous formulae, the algebraic complexity of the RGK-P5 scheme is reduced. Thus, the hope is to maintain the high accuracy given by the use of a fifth-order polynomial, $\widetilde{u}(x)$, while reducing the computational effort.

To verify this, let us calculate the complex eigenvalues characterizing (7).

For the kth stage of (7), the amplification matrix, $G_k(\beta, v)$, written as

$$G_k(\beta, v) = \begin{bmatrix} 0 & -v & 0 \\ 0 & 0 & -v \\ 60v(\mathrm{e}^{-j\beta} - 1) & 12v(3 + 2\mathrm{e}^{-j\beta}) & 3v(\mathrm{e}^{-j\beta} - 3) \end{bmatrix}$$

Consequently, for the whole process we get the following result:

$$\hat{U}^{n+1} = \underbrace{[I + \alpha_5 G_k[I + \alpha_4 G_k[I + \alpha_3 G_k[I + \alpha_2 G_k[I + \alpha_1 G_k]]]]]}_{\equiv G(\beta, v)} \hat{U}^n$$

Then, the global amplification matrix, $G(\beta, v)$, is deduced from this result. This matrix is calculated by using the MAPLE symbolic language. Therefore, we can calculate the complex eigenvalues that characterize such a scheme.

Figure 7 plots the amplitude of the accurate and spurious eigenvalues, for $v = 0.25$. As one can note, the minimum value reached by the accurate eigenvalue is comparable with the results obtained with the Δ-P5 and Δ-P5NC schemes: $|\lambda_1| = 0.981$; moreover, all the spurious eigenvalues remain below the unity. However, this latter result is not always true: Figure 8 shows the same results for a larger CFL: $v = 0.5$. For such a value, the modulus of the spurious eigenvalues becomes greater than unity. This result is troublesome. Indeed, this result indicates that the spurious component may become greater than the accurate one in the numerical solution (see [1]). For instance, the spurious component can be triggered by the boundary conditions or if the initial solution is not specified carefully. Therefore, the numerical performances of the RGK-P5 scheme seem to be limited to low CFL numbers.

Figure 9 confirms this analysis. The CFL number must remain below the value of 0.5 to reach the target of 0.01% with no more than 8 CPW.

DOI: 10.1002/fld

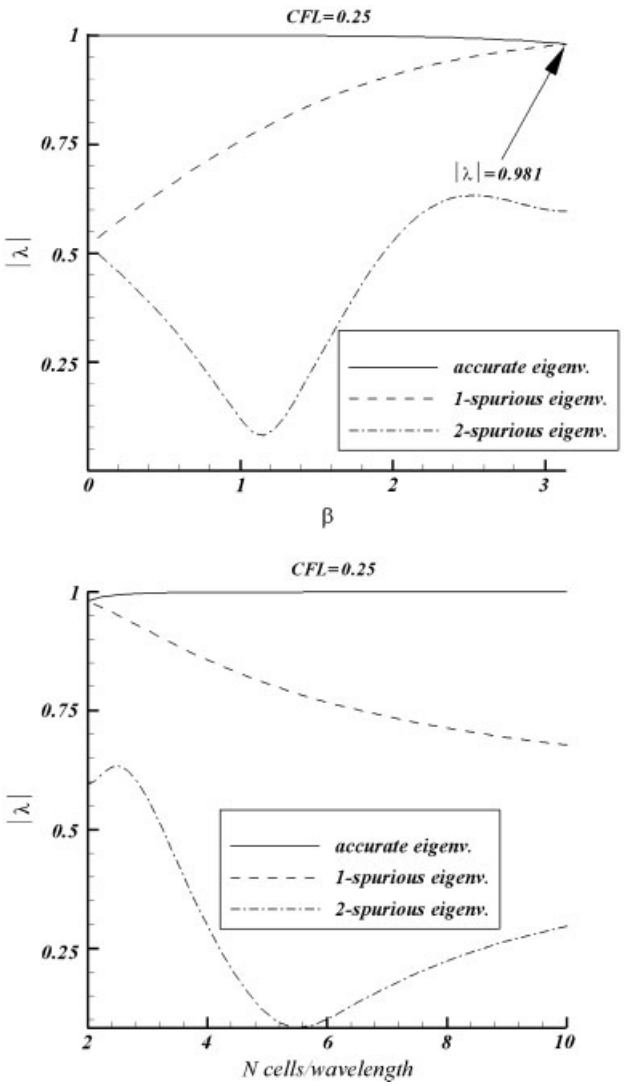

Figure 7. RGK-P5 scheme: accurate and spurious eigenvalues amplitudes ($\nu = 0.25$).

Consequently, despite a lower algebraic complexity, the computational efficiency of the RGK-P5 does not seem to be improved with respect to the previous schemes.

To quantify this, let us compute a linear problem with the three schemes we derived.

2.4. Computational efficiency analysis

We test the accuracy and the computational efficiency of the three previous schemes by considering the model problem: $u_t + u_x = 0, \ \forall x \in [0, 1]$.

DOI: 10.1002/fld

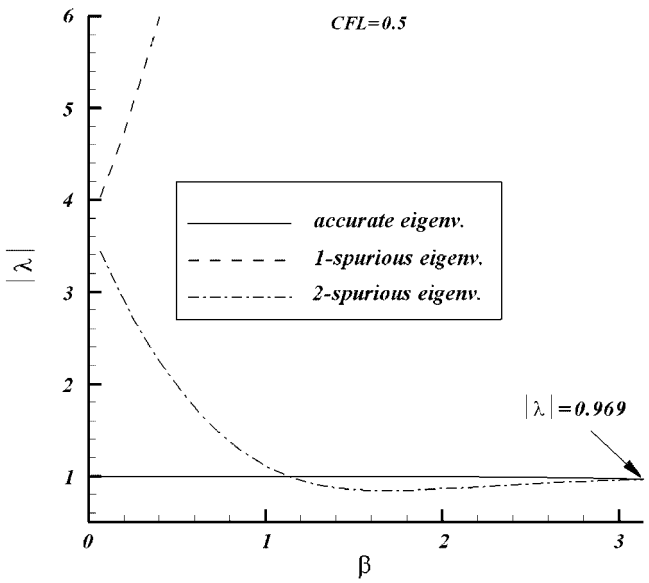

Figure 8. RGK-P5 scheme: amplification of the spurious modes for $v = 0.5$.

With the initial condition, $u(x, t = 0) = \sin(2\pi x)$ and 2-periodic boundary conditions, we compute the solution up to $T = 10$, i.e. after 10 time periods. A uniform mesh with N cells is used for all the test cases. For each scheme, the CFL number is selected in order to provide the lowest numerical error on a given mesh. Thus, the CFL number is taken equal to 0.75 for both Δ-P5 and Δ-P5NC schemes. For the RGK-P5 scheme, we take a CFL equal to 0.25. In addition, we take the Δ-P3 scheme [1], the LDDRK6-DRP scheme [4, 5], the FDo11p/RKo6s scheme [6] and the compact prefactored scheme of Ashcroft and Zhang [9] as references (CFL $= 0.75$). The compact scheme of Ashcroft is an improved version of Lele's scheme, in order to reduce the stencil support [8]; for this test, we made the choice of the optimized fourth-order five-point scheme ('8/4 scheme'), combined with a four-stage Runge–Kutta scheme (see [9] for more practical details). Concerning the accuracy, the results are shown in Tables I (Δ-P5), II (Δ-P5NC), III (RGK-P5), IV (LDDRK6-DRP), V (Compact scheme) and VI (FDo11p/RKo6s scheme). As can be seen, all the schemes achieve their designed order of accuracy. However, the Δ-P5 scheme produces better levels of error on every mesh whereas the Δ-P5NC and the RGK-P5 schemes give similar results: the utility of a reduced stencil in the discretization process is then emphasized. The compact scheme [9] produces better results than the LDRRK6-DRP scheme and equivalent results with those of the FDo11p/RKo6s scheme. However, these schemes remain less accurate than the schemes we derived.

Now, Figures 10 and 11 compare these schemes in terms of CPU efficiency. As a reference, we added the results obtained with the Δ-P3 scheme [1]. First, whatever the formulation considered, we obtain the encouraging result that demonstrates that a fifth-order scheme is less CPU-time consuming than a fourth-order or even a third-order discretization. In addition, the FDo11p/RKo6s scheme is almost equivalent to the DRP scheme, Figure 11, but it remains much less advantageous that the schemes we derived. Therefore, algebraic complexity is not such a decisive element to

DOI: 10.1002/fld

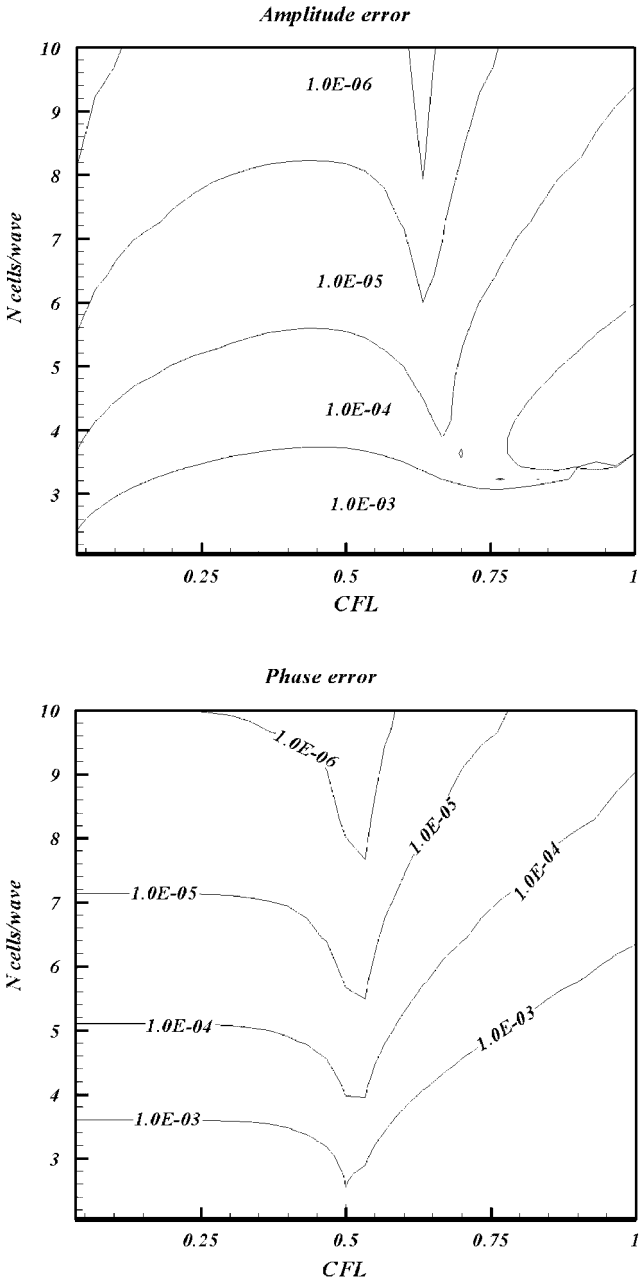

Figure 9. RGK-P5 scheme: amplitude and phase errors trends for the scalar advection equation $u_t + au_x = 0$.

make it possible to select or discard a numerical scheme, at least for the results presented here. On coarse meshes, the RGK-P5 scheme is more advantageous than the Δ-P5NC scheme; on the finest meshes, the Δ-P5NC scheme becomes preferable: this fact seems to be a consequence of the

DOI: 10.1002/fld

Table I. $u_t + u_x = 0$; $u(x, t = 0) = \sin(2\pi \times x)$; Δ-P5 scheme with periodic boundary conditions; $T = 10$; $v = 0.75$; L_1 and L_∞ errors.

N	L_1 error	L_1 order	L_∞ error	L_∞ order
10	1.22×10^{-4}	—	1.92×10^{-4}	—
20	2.91×10^{-6}	5.2	4.57×10^{-6}	5.2
40	8.01×10^{-8}	5.1	1.25×10^{-8}	5.1
80	2.34×10^{-9}	5	3.68×10^{-9}	5
160	7.10×10^{-11}	5	1.11×10^{-11}	5
320	4.11×10^{-12}	4	6.47×10^{-12}	4

Table II. $u_t + u_x = 0$; $u(x, t = 0) = \sin(2\pi \times x)$; Δ-P5NC scheme with periodic boundary conditions; $T = 10$; $v = 0.75$; L_1 and L_∞ errors.

N	L_1 error	L_1 order	L_∞ error	L_∞ order
10	2.02×10^{-3}	—	3.15×10^{-3}	—
20	6.22×10^{-5}	4.5	9.76×10^{-5}	4.5
40	1.86×10^{-6}	4.9	2.92×10^{-6}	4.9
80	5.59×10^{-8}	5	8.77×10^{-8}	5
160	1.70×10^{-9}	5	2.67×10^{-9}	5
320	5.24×10^{-11}	5	8.24×10^{-11}	5

Table III. $u_t + u_x = 0$; $u(x, t = 0) = \sin(2\pi \times x)$; RGK-P5 scheme with periodic boundary conditions; $T = 10$; $v = 0.25$; L_1 and L_∞ errors.

N	L_1 error	L_1 order	L_∞ error	L_∞ order
10	8.94×10^{-4}	—	1.40×10^{-3}	—
20	2.16×10^{-5}	5.2	3.40×10^{-5}	4.5
40	5.96×10^{-7}	5.1	9.36×10^{-7}	4.9
80	1.75×10^{-8}	5	2.75×10^{-8}	5
160	5.30×10^{-10}	5	8.33×10^{-10}	5
320	1.66×10^{-11}	5	2.61×10^{-11}	5

Table IV. $u_t + u_x = 0$; $u(x, t = 0) = \sin(2\pi \times x)$; LDDRK6-DRP scheme with periodic boundary conditions; $T = 10$; $v = 0.75$; L_1 and L_∞ errors.

N	L_∞ error	L_∞ order	L_1 error	L_1 order
10	0.20	—	0.13	—
20	1.36×10^{-2}	3.60	8.65×10^{-3}	3.60
40	8.12×10^{-4}	3.70	5.17×10^{-4}	3.70
80	4.89×10^{-5}	4	3.11×10^{-5}	4
160	3.0×10^{-6}	4	1.90×10^{-6}	4
320	1.85×10^{-7}	4	1.18×10^{-7}	4

DOI: 10.1002/fld

Table V. $u_t + u_x = 0$; $u(x, t=0) = \sin(2\pi \times x)$; factored compact scheme [9] with periodic boundary conditions; $T = 10$; $v = 0.75$; L_1 and L_∞ errors.

N	L_∞ error	L_∞ order	L_1 error	L_1 order
10	2.40×10^{-2}	—	4.50×10^{-2}	—
20	2.51×10^{-4}	3.80	4.80×10^{-4}	3.80
40	2.90×10^{-5}	3.80	5.60×10^{-5}	3.80
80	2.80×10^{-6}	4	5.40×10^{-6}	4
160	2.34×10^{-7}	4	4.50×10^{-7}	4
320	1.85×10^{-8}	4	2.90×10^{-8}	4

Table VI. $u_t + u_x = 0$; $u(x, t=0) = \sin(2\pi \times x)$; Fdo11p/RKo6s scheme [6] with periodic boundary conditions; $T = 10$; $v = 0.75$; L_1 and L_∞ errors.

N	L_∞ error	L_∞ order	L_1 error	L_1 order
10	4.51×10^{-3}	—	2.88×10^{-3}	—
20	1.16×10^{-3}	2	7.40×10^{-4}	2
40	8.41×10^{-5}	3.80	5.21×10^{-5}	3.70
80	5.29×10^{-6}	4	3.36×10^{-6}	4
160	3.27×10^{-7}	4	2.08×10^{-7}	4
320	2.02×10^{-8}	4	1.29×10^{-8}	4

reduced stability domain of the RGK-P5 scheme, when the grid resolution increases, the round-off error increases more rapidly with the RGK-P5 scheme.

Lastly, Figure 12 makes it possible to compare the Δ-P5 and RGK-P5 schemes with the other centred methods, this in terms of cost/efficiency for a given accuracy (0.01% in L_1 norm for the error level). As can be seen, the best power of resolution is obtained both with the Δ-P5 and the RGK-P5 schemes, Figure 12(a), since those schemes approximately necessitate 15 grid points for the global numerical discretization error to remain below the 0.01% level. The RGK-P5 scheme can even be regarded as the best one in terms of CPU efficiency; however, this scheme suffers from a reduced domain of stability (CFL$\leqslant$0.30) in comparison with the Δ-P5 scheme, Figure 12(b). Moreover, one can note that the resolution power of the compact scheme [9] decreases as the CFL increases (more than 40 grid points become necessary when the CFL is higher than the 0.5 value) while the DRP and the FDo11p/RKo6s do not exhibit this behaviour: owing to their optimization process of the Runge–Kutta algorithm, those schemes can even tolerate CFLs larger than one, while preserving their accuracy. The FDo11p/RKo6s even gives the best results obtained with a centred scheme (less than 30 grid points to reach the 0.01% target when the CFL is larger than one), Figure 12(a).

To conclude, this analysis puts forward two numerical schemes. The Δ-P5 scheme is the more efficient in terms of CPU time and resolution power, whereas the RGK-P5 scheme is easier to implement while keeping a high degree of performances. In some sense, one could consider that the Δ-P5NC scheme joins the drawbacks presented by the two previous numerical schemes. For this latter reason, only the Δ-P5 and the RGK-P5 scheme will be preserved in what follows.

DOI: 10.1002/fld

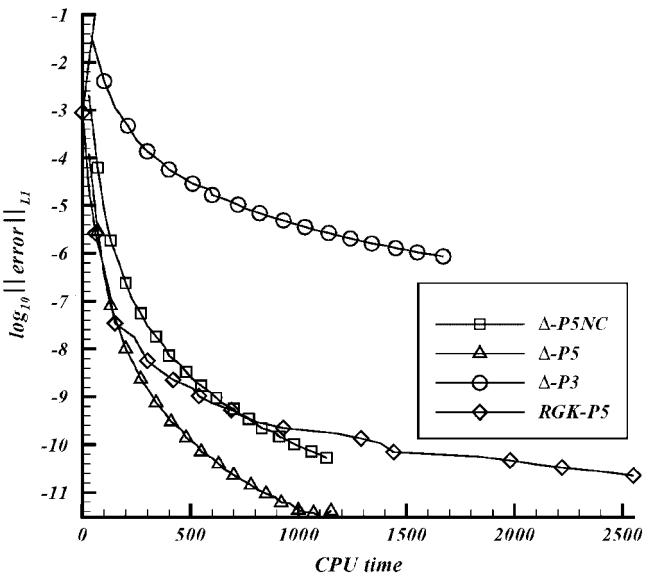

Figure 10. Linear advection $u_t + u_x = 0$ ($u(x, t=0) = \sin(2\pi \times x)$ $\forall x \in [0, 1]$). L_1 error by CPU time for the Δ-P3 (v=0.75), Δ-P5 (v=0.75), RGK-P5 (v=0.25) and Δ-P5NC (v=0.75) schemes. N=10→320 points.

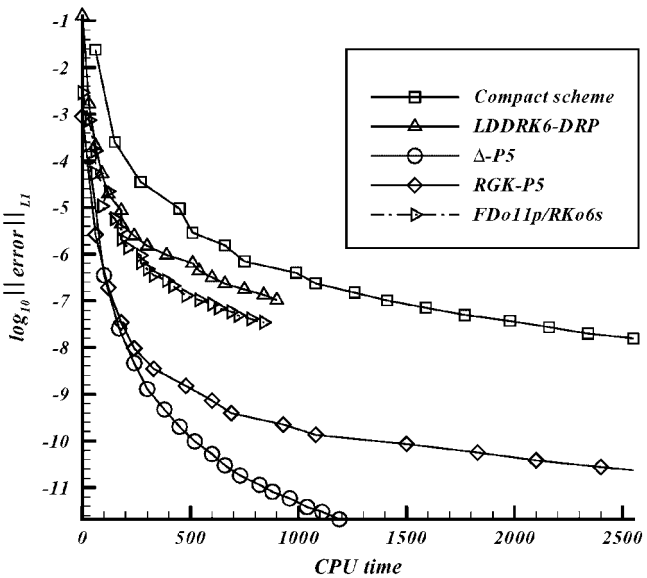

Figure 11. Linear advection $u_t + u_x = 0$ ($u(x, t = 0) = \sin(2\pi \times x)$ $\forall x \in [0, 1]$). L_1 error by CPU time for the Δ-P5 ($v = 0.75$), RGK-P5 ($v = 0.25$), LDDRK6-DRP [4, 5] ($v = 0.75$), FDo11p/RKo6s [6] ($v = 0.75$) schemes and the prefactored compact scheme [9] ($v = 0.75$). $N = 10 \rightarrow 320$ points.

DOI: 10.1002/fld

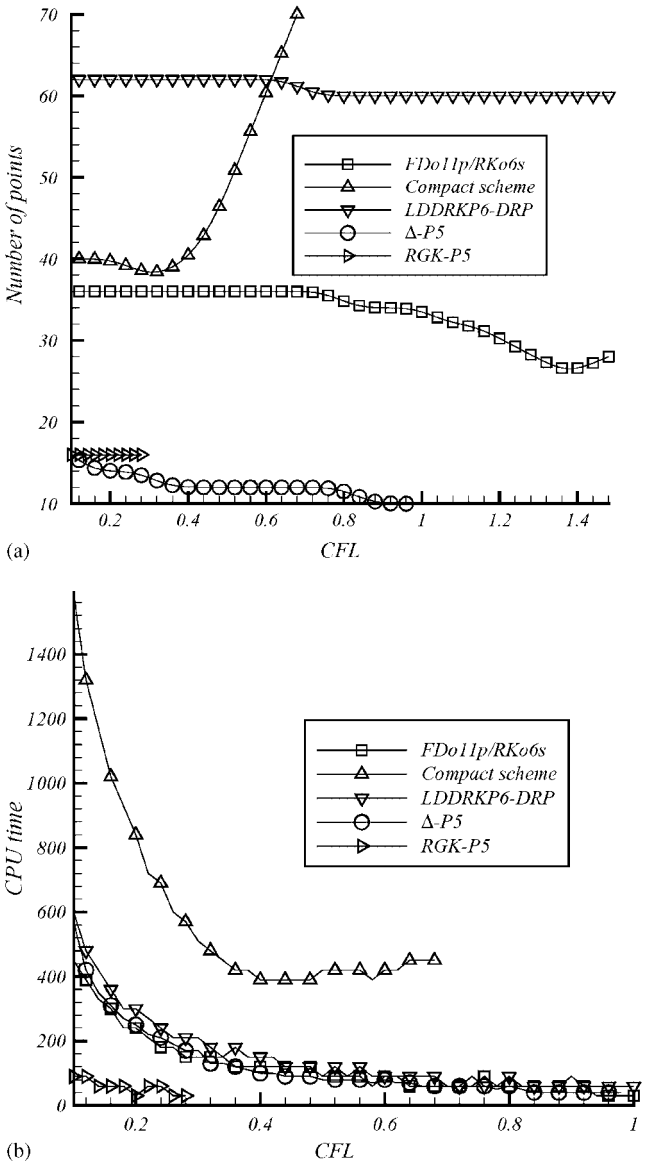

Figure 12. Linear advection $u_t + u_x = 0$ $(u(x, t{=}0) = \sin(2\pi \times x)\ \forall x \in [0, 1])$. Number of points by CFL and CPU time by CFL for $\|\text{error}\|_{L_1} = 1 \times 10^{-4}$. Δ-P5, RGK-P5, LDDRK6-DRP [4, 5], FDo11p/RKo6s [6], schemes and the prefactored compact scheme [9].

In the following sections, we generalize these schemes to bi-dimensional problems in order to model linear acoustics and aeroacoustics. To this aim, we extend and modify the numerical procedures detailed in [1, 3]. In what follows, we preserve the FDo11p/RKo6s and the Δ-P3

DOI: 10.1002/fld

schemes as references, the FDo11p/RKo6s scheme being easier to implement and giving a better CPU efficiency than the compact scheme [9] or a better error level than the DRP scheme.

3. FIFTH-ORDER UPWIND SCHEMES FOR LINEAR HYPERBOLIC SYSTEMS

In this section, we present the main characteristics and innovations of our algorithm in order to discretize, with an accuracy of the fifth order, a linear hyperbolic system. For more practical details, the reader is invited to consult Reference [1] (acoustics) or [3] (aeroacoustics).

In Cartesian coordinates (x, y, t), we define the following linear hyperbolic system:

$$\varphi_t + \mathscr{L}(\varphi) = 0 \tag{8}$$

$\mathscr{L}$ is a linear non-dissipative operator and φ, is defined as a generic variable representing the acoustic fluctuations, solutions of (8). For a 2D acoustic problem, we have $\varphi \equiv [u, v, p]^{\mathrm{t}}$.

If one solves the 2D linear acoustics, $\mathscr{L}(\varphi)$ is such as

$$\mathscr{L}(\varphi) \equiv \begin{cases} \dfrac{\partial p}{\partial x} \\ \dfrac{\partial p}{\partial y} \\ \left(\dfrac{\partial u}{\partial x} + \dfrac{\partial v}{\partial y}\right) \end{cases} \tag{9}$$

If we solve the 2D linear aeroacoustics, $\mathscr{L}(\varphi)$ is written as

$$\mathscr{L}(\varphi) \equiv \begin{cases} \dfrac{\partial p}{\partial x} + M_{xo}\dfrac{\partial u}{\partial x} + M_{yo}\dfrac{\partial u}{\partial y} \\ \dfrac{\partial p}{\partial y} + M_{xo}\dfrac{\partial v}{\partial x} + M_{yo}\dfrac{\partial v}{\partial y} \\ \left(\dfrac{\partial u}{\partial x} + \dfrac{\partial v}{\partial y}\right) + M_{xo}\dfrac{\partial p}{\partial x} + M_{yo}\dfrac{\partial p}{\partial y} \end{cases} \tag{10}$$

where $M_{xo} \equiv u_o/a_o$, $M_{yo} \equiv v_o/a_o$ (a_o: sound of speed in the background flow).

System (8) being linear, the first derivatives, $D\varphi$, and the second derivatives in space, $D^2\varphi$, of φ, verify the following system:

$$\begin{aligned} (D\varphi)_t + \mathscr{L}(D\varphi) &= 0 \\ (D^2\varphi)_t + \mathscr{L}(D^2\varphi) &= 0 \end{aligned} \tag{11}$$

Doing so, we have generated a generalized hyperbolic system (8)–(11) for the primitive variables, φ and its spatial derivatives, $D\varphi$ and $D^2\varphi$. To solve this problem with an upwind numerical scheme, we decompose the discretization into three stages: the 'decomposition stage'; the 'reconstruction stage'; and the 'evolution stage'.

DOI: 10.1002/fld

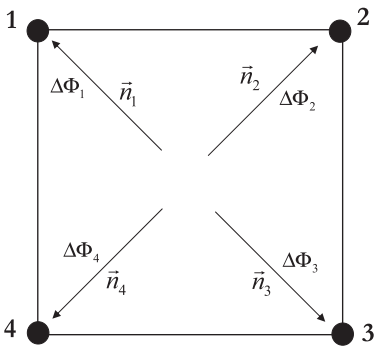

Figure 13. Generic cell: numbering of the vertices according to the multi-dimensional wave decomposition for linear acoustics (Δ-P5 and RGK-P5 schemes).

3.1. Decomposition of the time derivatives: the 'decomposition stage'

Following the method initiated in [1, 3] and the theory elaborated by Roe [17], we decompose the time derivatives of (8) in simple waves, according to the formula

$$\Delta t \times \varphi_t \equiv -\Delta t \times \mathscr{L}(\varphi) = -\Delta t \sum_{\ell=1}^{N_w} \lambda_{n\ell} \psi_\ell \mathbf{r}_{n\ell} \equiv \sum_\ell \Delta\phi_\ell \tag{12}$$

$\lambda_{n\ell}$ is the ℓth eigenvalue of the operator, $\mathscr{L}$, and represents the speed of the ℓth simple wave in the direction $\mathbf{n}_\ell$, while $\mathbf{r}_{n\ell}$ is its associated eigenvector. Lastly, ψ_ℓ is the intensity of the ℓth simple wave: it is a combination of the space derivatives of φ.

The model that decomposes the time derivatives is typified by the choice of N_w, number of simple waves in the decomposition, and the orientations of these waves, parametrized by the angle θ_ℓ such that: $\mathbf{n}_\ell \equiv \mathbf{n}(\theta_\ell)$.

For example, in 2D acoustics, $N_w = 4$ and $\theta_\ell \equiv 3\pi/4 - (\ell - 1)\pi/2,\ \ell \in \{1, 2, 3, 4\}$: this means that two pairs of mutually orthogonal acoustic waves are selected to decompose the time derivatives (see Reference [1] for more details).

Alternatively, when (8) represents the 2D aeroacoustics (10), we select $N_w = 6$ (four mutually orthogonal acoustic waves and two orthogonal shear waves) and $\theta_\ell \equiv A\tan(v/u) + (\ell - 1)\pi/2$, $\ell \in \{1, 2, 3, 4, 5, 6\}$ (see Reference [3] for more details).

For each case, the expressions of the eigenvalues, eigenvectors and wave strengths are given and detailed in [1, 3].

Thus, decomposition (12) makes it possible to identify the first-order perturbation (or 'fluctuation') in the variable φ, namely $\Delta\Phi_\ell$, caused by the ℓth wave being propagated in the direction given by $\mathbf{n}_\ell$. According to the sign of $\lambda_{n\ell}$, this fluctuation is sent to the vertex towards which the ℓth wave moves. Figure 13 illustrates this principle when (8) represents the 2D acoustics (9).

In the same way, by using the linearity of the operator, $\mathscr{L}$, one can derive from (12), the two following decompositions for the first ($D\varphi$) and second space derivatives ($D^2\varphi$):

$$\begin{aligned} \Delta t \times (D\varphi)_t &\equiv -\Delta t \times \mathscr{L}(D\varphi) = -\Delta t \sum_\ell \lambda_{n\ell} D\psi_\ell \mathbf{r}_{n\ell} \equiv \sum_\ell \Delta(D\Phi_\ell) \\ \Delta t \times (D^2\varphi)_t &\equiv -\Delta t \times \mathscr{L}(D^2\varphi) = -\Delta t \sum_\ell \lambda_{n\ell} D^2\psi_\ell \mathbf{r}_{n\ell} \equiv \sum_\ell \Delta(D^2\Phi_\ell) \end{aligned} \tag{13}$$

DOI: 10.1002/fld

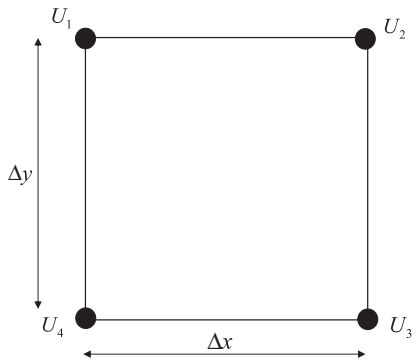

Figure 14. Δ-P5 and RGK-P5 schemes: definition of the discrete stencil for the calculation of the local spatial interpolator of the variable $U \equiv [\varphi, \varphi_x, \varphi_y, \varphi_{xx}, \varphi_{xy}, \varphi_{yy}]^t$ on the generic cell $(1, 2, 3, 4)$:
$\widetilde{\varphi}(x, y) = \text{complete fifth-order polynomial} + c_{22}x^5y + c_{23}x^3y^3 + c_{24}xy^5 \sim \varphi(x, y)$.

This means that the same ℓth simple wave is supposed to generate 'fluctuations' not only in the primitive variable φ ($\Delta\Phi_\ell$) but also in the first ($\Delta(D\Phi_\ell)$) and second derivatives ($\Delta(D^2\Phi_\ell)$) of this variable.

Therefore, this first stage makes it possible to decompose each temporal increase of the generalized system (8)–(11) into simple wave contributions. Now, to calculate ψ_ℓ and all its derivatives, we need to define the 'reconstruction stage'.

3.2. The form of the local spatial interpolator: the 'reconstruction stage'

In 2D, we define, on the generic cell $(1, 2, 3, 4)$, the local interpolator of φ, namely $\widetilde{\varphi}(x, y)$, at the discrete time $t_n \equiv n\Delta t$. In order to calculate a fifth-order polynomial interpolating the solution on this generic cell, all the variables $(\varphi, D\varphi, D^2\varphi)$ are considered at each discrete point (see Figure 14). Such a choice generates 24 parameters for calculating the polynomial.

Consequently, we select the generic form defining $\widetilde{\varphi}(x, y)$, as being

$$\widetilde{\varphi}(x, y) \equiv \text{complete fifth-order polynomial} + c_{22}x^5y + c_{23}x^3y^3 + c_{24}xy^5 \tag{14}$$

Such a form was selected because it produces an invertible algebraic system.

The expansion coefficients $\{c_i\}$ are obtained from the known data at the vertices of the mesh, by the relationships equivalent to (3). The system so generated can be formally inverted by using the MAPLE symbolic mathematical computer package. By doing so, we are able to calculate all the derivatives that we need in the calculation of the wave strengths, ψ_ℓ.

We give in Appendix A the formulae for these derivatives on a non-uniform Cartesian mesh.

3.3. Evolving in time the variable and its derivatives: the 'evolution stage'

The decomposition stage combined with the definition of the spatial interpolator, (14), produces the semi-discrete system (12) and (13).

Depending on the scheme considered, we use either the Cauchy–Kovaleskaya process (Δ-P5 scheme) or the Runge–Kutta algorithm (RGK-P5 scheme) to evolve the discrete solution in time.

DOI: 10.1002/fld

First, in the 'Cauchy–Kovaleskaya procedure', we expand in Taylor series in time the generic variable, φ, and its derivatives, $D\varphi$ and $D^2\varphi$, around the basis point (x_i, y_i, t_n). Then, we replace all the time derivatives by their spatial counterpart by using (8)–(11).

This generates the following algebraical system for the variables $(\varphi, D\varphi, D^2\varphi)$:

$$\begin{aligned}
\varphi_i^{n+1} &= \varphi_i^n + \sum_{k=1}^{5} (-1)^k \frac{\Delta t^k}{k!} \times \mathscr{L}^{[k]}(\varphi_i^n) \\
D\varphi_i^{n+1} &= D\varphi_i^n + \sum_{k=1}^{4} (-1)^k \frac{\Delta t^k}{k!} \times \mathscr{L}^{[k]}(D\varphi_i^n) \\
D^2\varphi_i^{n+1} &= D^2\varphi_i^n + \sum_{k=1}^{3} (-1)^k \frac{\Delta t^k}{k!} \times \mathscr{L}^{[k]}(D^2\varphi_i^n)
\end{aligned} \tag{15}$$

with the following definitions:

$$\mathscr{L}^{[k]} \equiv \underbrace{\mathscr{L} o \ldots\ldots o \mathscr{L}}_{k}, \quad \mathscr{L}^{[0]} \equiv Id, \quad \mathscr{L}^{[1]} \equiv \mathscr{L}$$

Then, by using (12), we can write

$$-\mathscr{L}^{[k]}(\varphi) = -\sum_{\ell} \mathscr{L}^{[k-1]}(\Delta\Phi_\ell) \equiv \sum_{\ell} \Delta\Phi_\ell^{[k]}$$

This relation makes it possible to identify $\Delta\Phi_\ell^{[k]}$, the 'fluctuation' of order k, for the variable φ, associated with the ℓth simple wave. The same identification is done for the variables $D\varphi$ ('fluctuation' $\Delta(D\Phi_\ell^{[k]})$) and $D^2\varphi$ ('fluctuation' $\Delta(D^2\Phi_\ell^{[k]})$), by using (13). These temporal increases of order k are treated in the same way as the 'fluctuations' of first order: they are sent towards the downwind vertices. In this way, each vertex receives all the 'fluctuations' (all the orders) for all the variables; these 'fluctuations' are then gathered to update the variables at the nodes of the mesh.

Doing so, this algorithm generates a fifth-order upwind method discretizing (8)–(11), namely the 'Δ-P5 scheme'.

If the operator, $\mathscr{L}$, is complicated (2D or 3D aeroacoustics), a Runge–Kutta procedure may become algebraically simpler.

Indeed, the equivalent of (15) in order to integrate the semi-discrete form (8)–(11), is then written as follows:

$$\begin{aligned}
\varphi_i^{[m]} &= \varphi_i^n - \alpha_m \Delta t \times \mathscr{L}(\varphi_i^{[m-1]}) \equiv \varphi_i^n + \alpha_m \sum_{\ell} \Delta\Phi_\ell(\varphi^{[m-1]}) \\
D\varphi_i^{[m]} &= D\varphi_i^n + \alpha_m \sum_{\ell} \Delta(D\Phi_\ell)(\varphi^{[m-1]}) \\
D^2\varphi_i^{[m]} &= D^2\varphi_i^n + \alpha_m \sum_{\ell} \Delta(D^2\Phi_\ell)(\varphi^{[m-1]})
\end{aligned} \tag{16}$$

for $m \in \{1, \ldots, 5\}$ and with the following definitions: $\varphi^{[0]} \equiv \varphi^n$, $\varphi^{n+1} \equiv \varphi^{[5]}$, $\alpha_m \equiv 1/(6-m)$.

This system is simpler than (15) because there is no more need to calculate 'fluctuations' of order k. However, the compactness in time is definitively lost. The algorithm so generated defines the 'RGK-P5' scheme.

DOI: 10.1002/fld

3.4. Non-reflecting boundary conditions for aeroacoustics

The physical principle we use to derive boundary conditions is very simple: each outgoing disturbance must leave the computation domain without being reflected.

In order to discriminate the numerical boundary conditions from the physical conditions, we strictly follow the procedure developed in [1, 3]: unless there exist physical ingoing disturbances, every wave entering the computation domain, must be set to zero in decomposition (15) or (16).

For example, if we suppose that a wall lies along the y-axis, normal velocity u being zero at every point situated on this boundary, whatever the time t, we can deduce from (11) the following

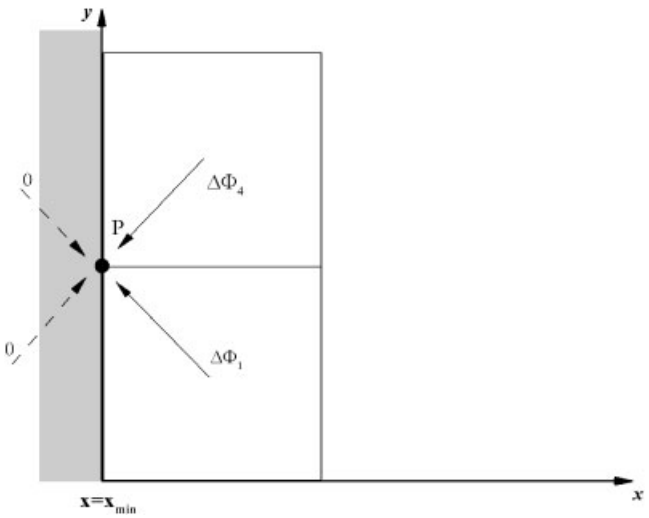

Figure 15. Wall boundary conditions for a solid wall located at $x = x_{\min}$.

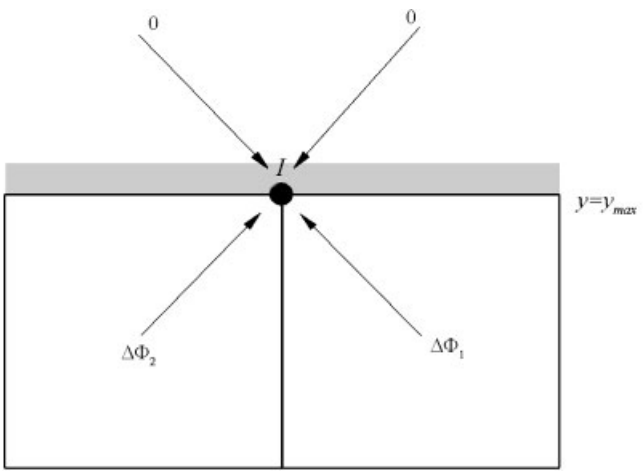

Figure 16. Radiation boundary conditions along the line $y = y_{\max}$.

DOI: 10.1002/fld

conditions for the derivatives of u, v, p. The remaining variables, which are not explicitly specified, are updated by using the algebraic expressions defining either the Δ-P5 scheme (15) or the RGK-P5 scheme (16). In this example, the contribution of the waves ψ_2 and ψ_3 is then set to zero (Figure 15) into these formulae since these waves do not physically exist. In other words, the remaining variables at the wall are only updated by the waves coming from the interior of the computational domain.

Now, if we suppose that the line $y = y_{\max}$ is a radiation boundary, Figure 16, then points situated on this line are updated by using the two outgoing acoustic waves ψ_2 and ψ_1. The non-reflective boundary conditions will be ensured by setting the ingoing waves, ψ_4 and ψ_3, to zero in (15) or (16), (Figure 16).

4. NUMERICAL TESTS

Firstly, we present the results for numerical comparisons made with the Δ-P5 and RGK-P5 schemes in computing the solution of the aeroacoustic equations. Since it gave the best results in [1, 3], the Δ-P3 scheme is used as reference. Unless specified, a square grid with uniform spacing, h, in the two space directions, is employed. The CFL number is thus defined by $\mathrm{CFL} \equiv \Delta t / h$. The derivatives used to calculate the wave strengths are calculated according to the Cartesian formulae proposed in Appendix A.

4.1. Computational efficiency analysis

The numerical experiments are carried out with a doubly periodic solution of (8) when $\mathscr{L}$ is the 2D acoustic operator, (9), and for (x, y) defined in the domain $[-10, 10] \times [-10, 10]$. We use the following exact solution as reference:

$$\begin{aligned} p(x, y, t) &= \cos(2\pi\sqrt{2} \times t) \times \sin(2\pi \times x) \times \sin(2\pi \times y) \\ u(x, y, t) &= -\frac{1}{\sqrt{2}} \sin(2\pi\sqrt{2} \times t) \times \cos(2\pi \times x) \times \sin(2\pi \times y) \\ v(x, y, t) &= -\frac{1}{\sqrt{2}} \sin(2\pi\sqrt{2} \times t) \times \sin(2\pi \times x) \times \cos(2\pi \times y) \end{aligned} \tag{17}$$

The numerical solution is computed until $T = 1$.

Grid refinement data are provided for mesh sizes going from 20×20 points to 320×320 points. For all the schemes investigated, the CFL number is optimized to get the best results in terms of resolution *versus* computational cost. All the results obtained by the Δ-P3 scheme are got from [1] or [3].

The CFL number is taken equal to 0.45 for the Δ-P3 scheme and 0.15 for the Δ-P5 scheme. For the RGK-P5 scheme, we select the CFL number in the range $[5 \times 10^{-2}, 0.10]$. Figure 17 plots the $\log_{10}$ of L_1 error in pressure against the $\log_{10}$ of the mesh size.

As one can note it, both Δ-P5 and RGK-P5 schemes exhibit the same behaviour in terms of error level (Figure 17(a)). However, when we plot the L_1 error against the CPU time required for each level of grid, the differences between those two schemes become obvious. Indeed, the results obtained in Section 2.2 are verified: despite a greater algebraic complexity the Δ-P5 scheme is

DOI: 10.1002/fld

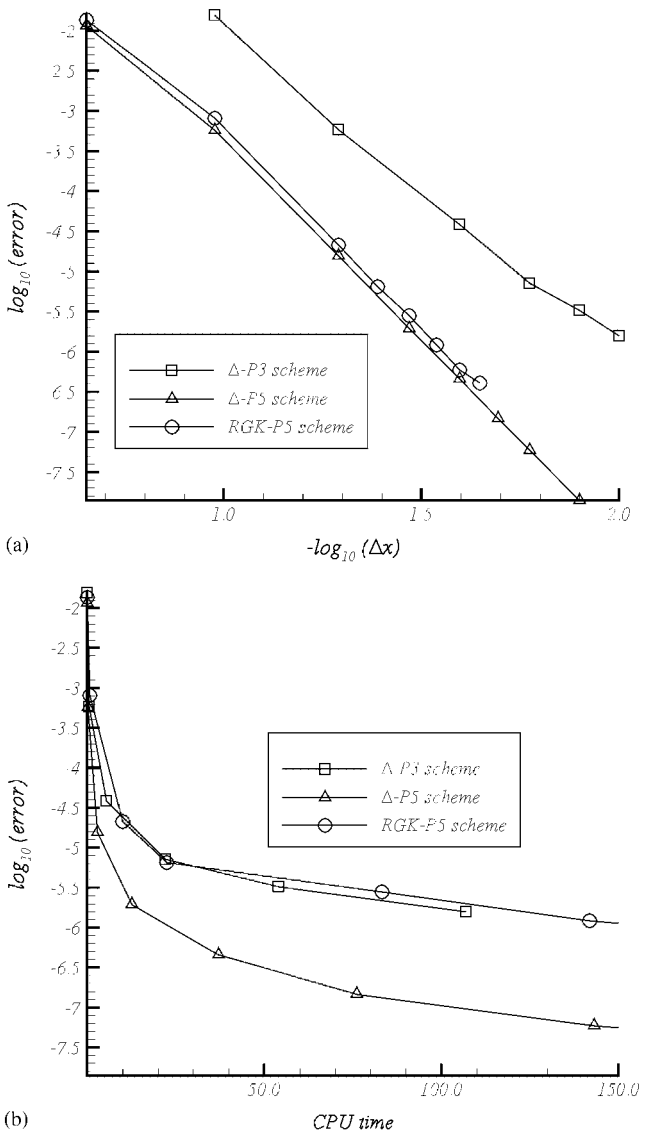

Figure 17. Linear acoustics: doubly periodic solution. Top: L_1 error by grid resolution; bottom: L_1 error by CPU time; $T=1$; Δ-P3 (CFL=0.45); Δ-P5 (CFL=0.15); and RGK-P5 (CFL$\in[5\times10^{-2}, 0.10]$) schemes.

more advantageous in terms of CPU time (Figure 17(b)). The RGK-P5 scheme produces results similar with the Δ-P3 scheme.

However, another parameter should be considered in order to get a fair comparison between these schemes. Indeed, the computing memory requirement becomes an important point in multi-dimensional computations. Table VII presents the results obtained with the previous

DOI: 10.1002/fld

Table VII. Memory requirement and CPU time for a given level of error on a Cartesian mesh.

Memory	Δ-P5 (CFL = 0.15)			RGK-P5 (CFL = 2.5×10^{-2})			Fdo11p/RKo6s (CFL = 0.15)		
(Mo)	$N_x \times N_y$	L_1 error	t_{CPU}	$N_x \times N_y$	L_1 error	t_{CPU}	$N_x \times N_y$	L_1 error	t_{CPU}
5	40×40	1.6×10^{-5}	2.74	60×60	2.8×10^{-6}	84	180×180	2×10^{-5}	50
7	50×50	4.9×10^{-6}	6	70×70	1.2×10^{-6}	142	240×240	5.2×10^{-6}	90
10	60×60	1.9×10^{-6}	11	90×90	4.1×10^{-7}	340	300×300	2×10^{-6}	147

Note: Acoustics: doubly periodic solution on $[-10, 10] \times [-10, 10]$.

computations. In addition, we specify the CPU time associated with each result. Moreover, we use the FDo11p/RKo6s scheme [6] as reference, since such a scheme is a low memory consumer.

As one can note it, the RGK-P5 scheme produces the best results: for a given level of error, this scheme necessitates less computer memory than the Δ-P5 scheme or even the FDo11p/RKo6s scheme. However, this scheme is much more costly than the Δ-P5 scheme in terms of CPU time. In an unexpected way, the Δ-P5 scheme appears to be equivalent to the FDo11p/RKo6s scheme in terms of computer memory, for a given level of error. In addition, we have counted the number of floating point operations needed per grid points for each scheme: the FDo11p/RKo6s scheme needs 6×10^4 operations per grid points to perform these computations, while the Δ-P5 scheme necessitates 1×10^5 and the RGK-P5 scheme, 2.5×10^5.

Therefore, though the RGK-P5 scheme is not the best one in terms of CPU efficiency because of its reduced domain of stability, it presents the great advantage to be algebraically simpler and cheaper in computer memory than the Δ-P5 scheme. Consequently, we can consider the RGK-P5 scheme as a viable alternative to the Δ-P5 scheme for computing very weak signals on long time periods. The numerical performances of such a scheme are better than the ones of the FDo11p/RKo6s scheme; this is true not only in terms of CPU time but also in terms of computer memory requirement. However, the FDo11p/RKo6s scheme appears to be more advantageous, in terms of accuracy and stability, than the LDDRK6-DRP scheme (see [1] for more details).

In order to save CPU time, the numerical results presented in what follows are obtained by using the Δ-P5 scheme. However, exactly the same results can be obtained with the RGK-P5 scheme with the drawback of an increase of the computational cost around 40%.

4.2. Acoustic pulse reflected from a rigid wall [*18*]

In order to show the effectiveness of the radiation and wall boundary conditions used in this article, we consider, as a second example, an acoustic pulse reflected from a rigid wall. We use system (8) and (9) to model this problem.

This problem indicated under the name of problem of 'category IV' in [18] has an exact solution.

An initial Gaussian distribution for the impulse of pressure produces an acoustic pulse. Then, at the time $t = 0$, the initial conditions are as follows:

$$u = v = 0$$

$$p(x, y, 0) = \mathrm{e}^{-[10 \times (x^2 + (y+0.7)^2)]}$$

The derivation of this condition makes it possible to initialize the derivative variables.

The domain of solution is $-1 \leqslant x \leqslant 1$, $-1 \leqslant y \leqslant 1$, with the wall located at $y = -1$.

DOI: 10.1002/fld

Two boundary conditions are used: a wall condition along $y=-1$ and an acoustic radiation condition in the far field (top, left and right boundaries). Radiation and wall boundary conditions are imposed in the way detailed in Section 3.4.

The computation is carried out on a 20×20 Cartesian mesh and a CFL number of 0.5 is employed in this simulation.

Figure 18 gives the numerical results for the Δ-P5 scheme at $T=1, 1.2$ and 1.4, when the impulse of pressure reflected from the wall and reached the far-field boundaries.

Even on a so coarse mesh, the method functions correctly at the boundaries since no artificial reflections appear. Moreover, the comparisons between the solutions exact and calculated along a vertical cut at $x=0$, show a good agreement.

4.3. The two-dimensional piston problem [*19*]

Now, we consider the simulation of the acoustic field produced by an oscillating piston fixed on an infinite wall. We use system (8) and (9) to model this problem.

In dimensionless coordinates, the piston half-width is one unit and it oscillates at a rate such that the normal component of velocity to its surface is $u(t)=\cos(2\pi\times f\times t)$, with the frequency, f, selected as $f\in\{0.5, 1\}$. The computational domain is $0\leqslant x\leqslant 20$ and $-10\leqslant y\leqslant 10$. The exact solution to this problem is available in [19].

Firstly, let us consider the case $f=0.5$.

All computations are made with a CFL number of 0.15. At $t=0$, all the derivative variables are set to zero. Time integration is carried out for 10 cycles of oscillation for the piston: this is a sufficient time so that the initial wave front created by the piston, reaches the opposite corner of the domain. Figure 19 shows the results computed with the Δ-P5 scheme on a 60×60 Cartesian mesh (6 CPW for the wave propagating along the symmetry line). With such a grid resolution, the results are good. As we can note it, there are no spurious waves at the boundaries: the outgoing acoustic waves leave smoothly the computation domain without any obvious spurious oscillation. This fact remains unchanged when the simulation time is increased.

Figure 20 compares this solution with the exact one for a lateral and longitudinal pressure distribution: considering the accuracy of the results, such a grid resolution is sufficient. The CPU time needed to obtain this solution is 24 s.

This result must be compared with the CPU time necessary to obtain the same accuracy level with the Δ-P3 scheme: a grid of 160×160 points must be employed and the CPU time rises to 126 s. Moreover, on such a grid, the LDDRK6-DRP scheme gives numerical results that are insufficiently accurate (see numerical results in [1]).

Lastly, Figure 21 presents the numerical solution for the more oscillating case, $f=1$.

With such a value, 120×120 grid points become necessary in order to get a good resolution with the Δ-P5 scheme (6 CPW for the wave propagating along the symmetry line). The agreement with the analytical results is very good, Figure 22. Figure 21 shows contour plots of pressure profile calculated by the Δ-P5 scheme: the major part of acoustic energy is now transmitted along the x-axis and only one smaller part of this one is radiated diagonally. Once more, the non-reflective boundary conditions function correctly at the open boundaries, even when the time of simulation is increased. For this simulation, the CPU time is now 360 s.

On the same mesh, Figure 23 presents a comparison between the Δ-P5 and the Δ-P3 scheme: the lack of resolution of the third-order scheme is obvious. A grid of 320×320 points is necessary with this latter scheme to obtain the same accuracy level. In such a case, the CPU time needed is 1381 s (see numerical results in [1]).

DOI: 10.1002/fld

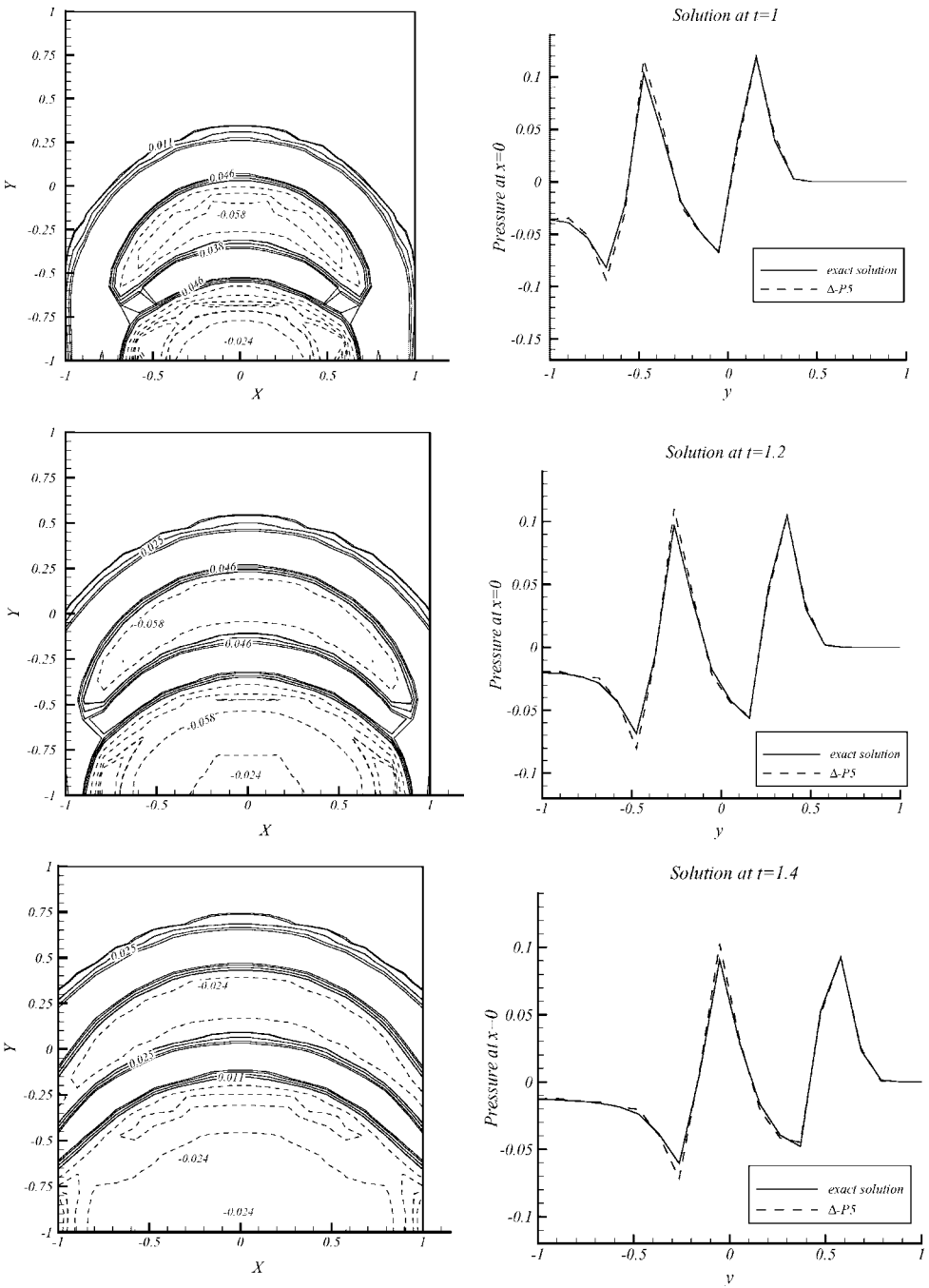

Figure 18. Linear acoustics: acoustic pulse reflected from a wall. Δ-5 scheme: CFL = 0.5; 20 × 20 Cartesian mesh.

DOI: 10.1002/fld

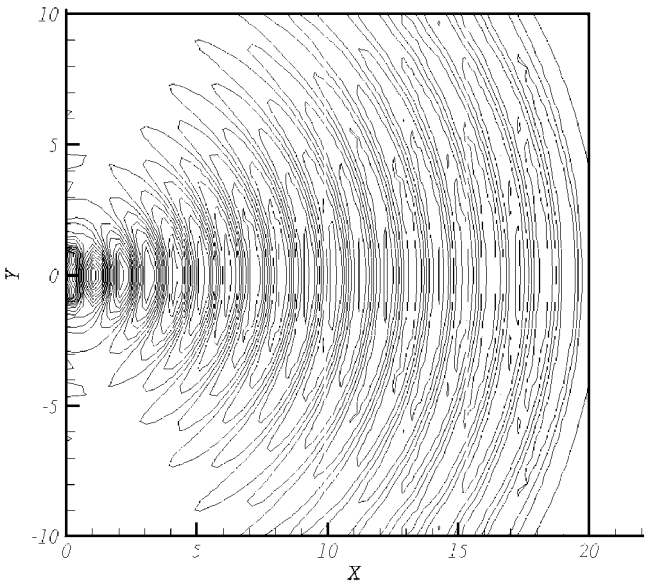

Figure 19. The piston problem: pressure contour plots at $T = 20$ for $f = 0.5$. Δ-P5 scheme (60×60 grid points, CFL $= 0.15$).

4.4. Acoustic scattering from a cylinder: *Category 1*, *problems 1 and 2* [*18*]

This example is one of the second CAA benchmark problems, [18]. The goal of this study is to check the effectiveness of curvilinear wall boundary and far-field non-reflecting conditions, when a source term exists. For the first problem considered, a 2D cylinder of radius $R = 0.5$ is located at the origin. At time $t = 0$, a first impulse of pressure is initiated at the point S, located at $(r = 4, \theta = 0°)$. The solution of this problem is typified by the temporal evolution of the pressure in three selected points: A $(r = 5, \theta = 90°)$, B $(r = 5, \theta = 135°)$ and C $(r = 5, \theta = 180°)$. The time of measurement goes from $t = 0$ to 10. We compare the measured values of pressure with the analytical ones, given in [18].

We obtain the solution to this problem by solving the acoustic equations, system (8) and (9), in polar coordinates. Therefore, the methodology described in Chapter 2 can be used, see [1].

The initial solution is as follows:

$$\begin{aligned} u_r(r, \theta) &= u_\theta(r, \theta) = 0 \\ p(r, \theta) &= \exp\{-17 \times [(r \cos\theta - 4)^2 + (r \sin\theta)^2]\} \end{aligned} \tag{18}$$

The derivation of this relation initializes all the derivative variables. This initial value problem is referenced as the category 1, problem 2 in [18].

For the numerical simulations, the distance of the far-field boundary from the cylinder is $r_{\max} = 3\pi$. The problem is symmetric about the x-axis, thus, only the solution in the upper-half x–y domain is computed. The grid resolution is selected such that $\Delta r = \Delta\theta \equiv h$. The Courant number remains defined by $\Delta t / h$. We employ 250 radial grid points and 84 azimuthal grid points for these simulations.

DOI: 10.1002/fld

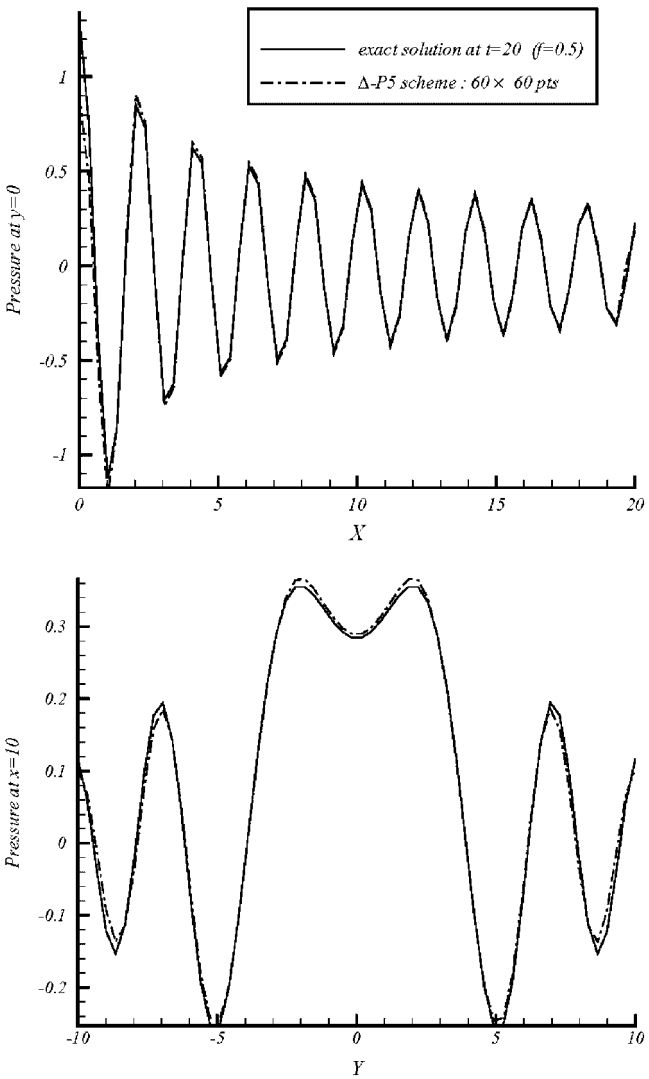

Figure 20. The piston problem: axial (top) and lateral pressure distribution (bottom) for the Δ-P5 scheme: CFL = 0.15; $T = 20$; $f = 0.5$; and 60×60 grid points.

Three boundary conditions are used for this problem: a wall condition on the cylinder ($r = 0.5$); an acoustic radiation condition in the far field ($r = r_{\max}$), following the procedure defined by Figure 16; a symmetry condition at the azimuthal boundaries $\theta = 0$ and π. A low CFL number is employed in this simulation (CFL = 0.10) in order to avoid any numerical problem at the points of stagnation ($r = 0.5$, $\theta = 0$ and π), situated at the same time on the wall and the line of symmetry.

Figure 24 shows pressure contour plots of these results with $t = 5$ and 6.2 (dashed lines represent negative values of the pressure). Figure 25 shows the temporal evolutions of the pressure at the points A, B, C. As one can note it, the numerical solution is in close agreement with the exact

DOI: 10.1002/fld

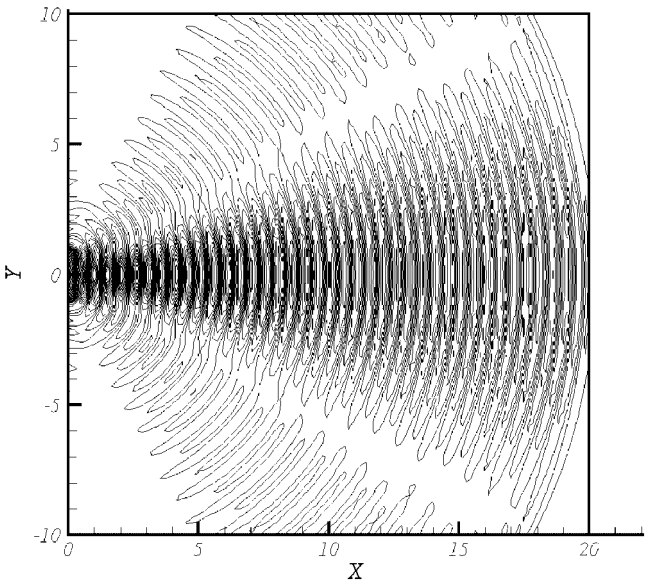

Figure 21. The piston problem: pressure contour plots at $T = 20$ for $f = 1$. Δ-P5 scheme (120×120 grid points, CFL $= 0.15$, $t_{\mathrm{CPU}} = 360$ s).

solution. The peak pressure value at the point A is $p_{\max} = 0.0604$, while the exact solution gives a value of 0.0610. To get similar results with the Δ-P3 scheme or the LDDRK6-DRP scheme, 400×134 grid points are necessary (see numerical results in [1]).

Now, with the same geometry, we consider the following initial solution:

$$u_r(r, \theta) = u_\theta(r, \theta) = p(r, \theta) = 0$$

Moreover, a time-periodic acoustic source centred at $(x = 4, y = 0)$ is introduced in the equation for the pressure:

$$S(x, y, t) = \exp\{-17 \times [(r\cos\theta - 4)^2 + (r\sin\theta)^2]\} \times \sin(\omega \times t)$$

This problem is referenced as the category 1, problem 1 in [18]. An analytic solution for the directivity $(r \times \bar{p}^2)$ is given in [18]. This problem is solved by the Δ-P5 scheme in exactly the same manner as in problem 2.

Figure 26 presents the numerical solution. The numerical results reported here use a spatial resolution of 301×101 grid points for the Δ-P5 scheme. The frequency of the acoustic source is selected such that $\omega = 6\pi$. We obtain the computed results by time marching to a time-periodic state. With a CFL number of 0.10 and a simulation time of $T = 14$, the CPU time is 7000 s. Figure 26 shows the computed directivity function at $r = 5$. It is easy to see that there is uniformly good agreement with the exact solution: this suggests that the Cartesian boundary treatment proposed is effective and accurate. The effectiveness of the boundary conditions is also clearly visible with the contours of pressure acoustic waves, Figure 26. To compare, we present the numerical results obtained with the Δ-P3 scheme. With 551×184 grid points, a CFL number of 0.25, the CPU time

DOI: 10.1002/fld

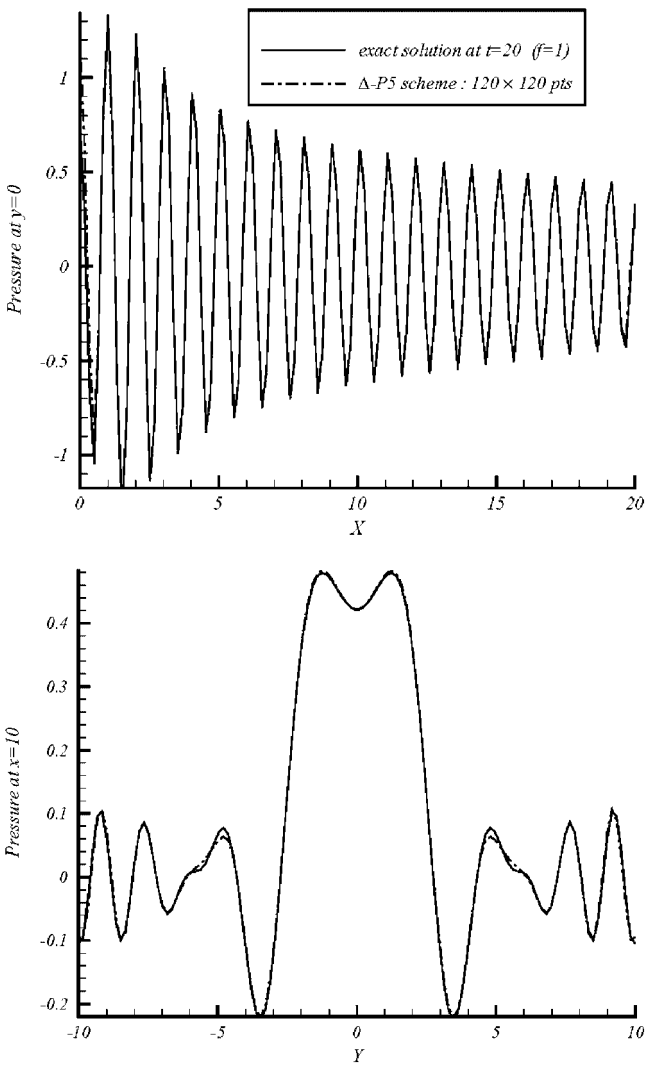

Figure 22. The piston problem: axial (top) and lateral pressure distribution (bottom) for the Δ-P5 scheme: CFL = 0.15; $T = 20$; $f = 1$; and 120×120 grid points.

needed to reach the time $T = 14$ is 8025 s. As one can note it, such a spatial resolution remains insufficient to give good results, Figure 26.

4.5. Axisymmetric pressure pulse [*20*]

With this problem, we test the effectiveness of the 'decomposition stage', defined in Section 3.1.

Our main goal is to validate the wave model used in the decomposition step, (12), for the Δ-P5 scheme. For this purpose, we solve the linear aeroacoustic equations, system (8)–(10), on the computation domain $[-1, 1] \times [-1, 1]$.

DOI: 10.1002/fld

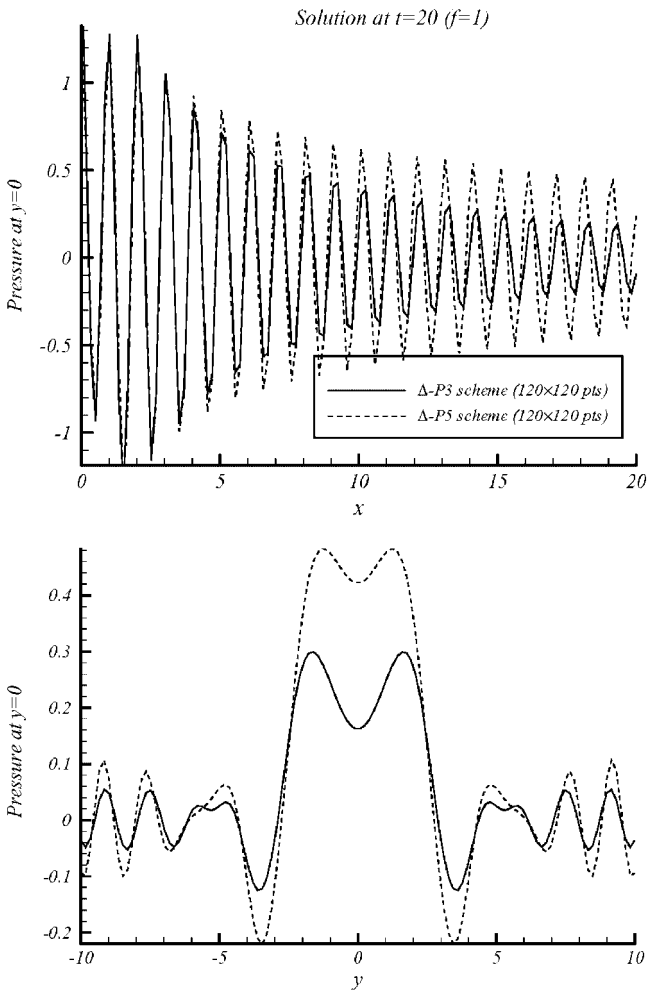

Figure 23. The piston problem: axial (top) and lateral pressure distribution (bottom) for the Δ-P5 (CFL = 0.15, $t_{\mathrm{CPU}} = 320$ s) and Δ-P3 (CFL = 0.25, $t_{\mathrm{CPU}} = 72$ s) schemes for 120×120 grid points.

An acoustic pulse is generated by an initial Gaussian pressure pulse distribution at the centre of the computation domain. The mean flow Mach number is defined by $\mathbf{M} \equiv (M_{xo}, M_{yo})^t$. The wave front of the acoustic pulse expands radially; but because of the mean flow, the whole wave pattern is being convected downstream at the same time, without any distortion.

Exact solution of this problem is reported in [20] and provides a reference to estimate the accuracy and the quality of the computed solutions.

This problem is parametrized by the mean flow Mach number. Thus, two flow regimes are considered: a subsonic flow regime characterized by ($M_{xo} = 0.8, M_{yo} = 0.5$) and a supersonic one, distinguished by ($M_{xo} = 2, M_{yo} = 1$).

DOI: 10.1002/fld

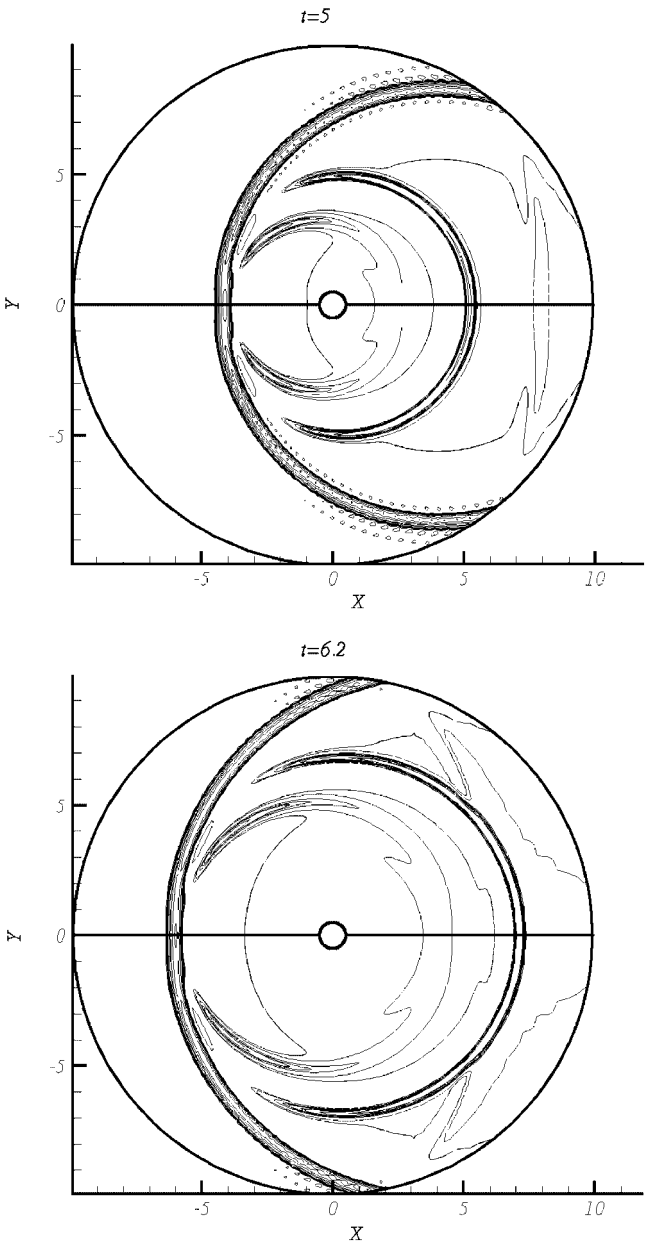

Figure 24. Acoustic scattering from a cylinder. Δ-P5 scheme: pressure pulse contours; 250×84 grid points; and CFL $= 0.10$.

DOI: 10.1002/fld

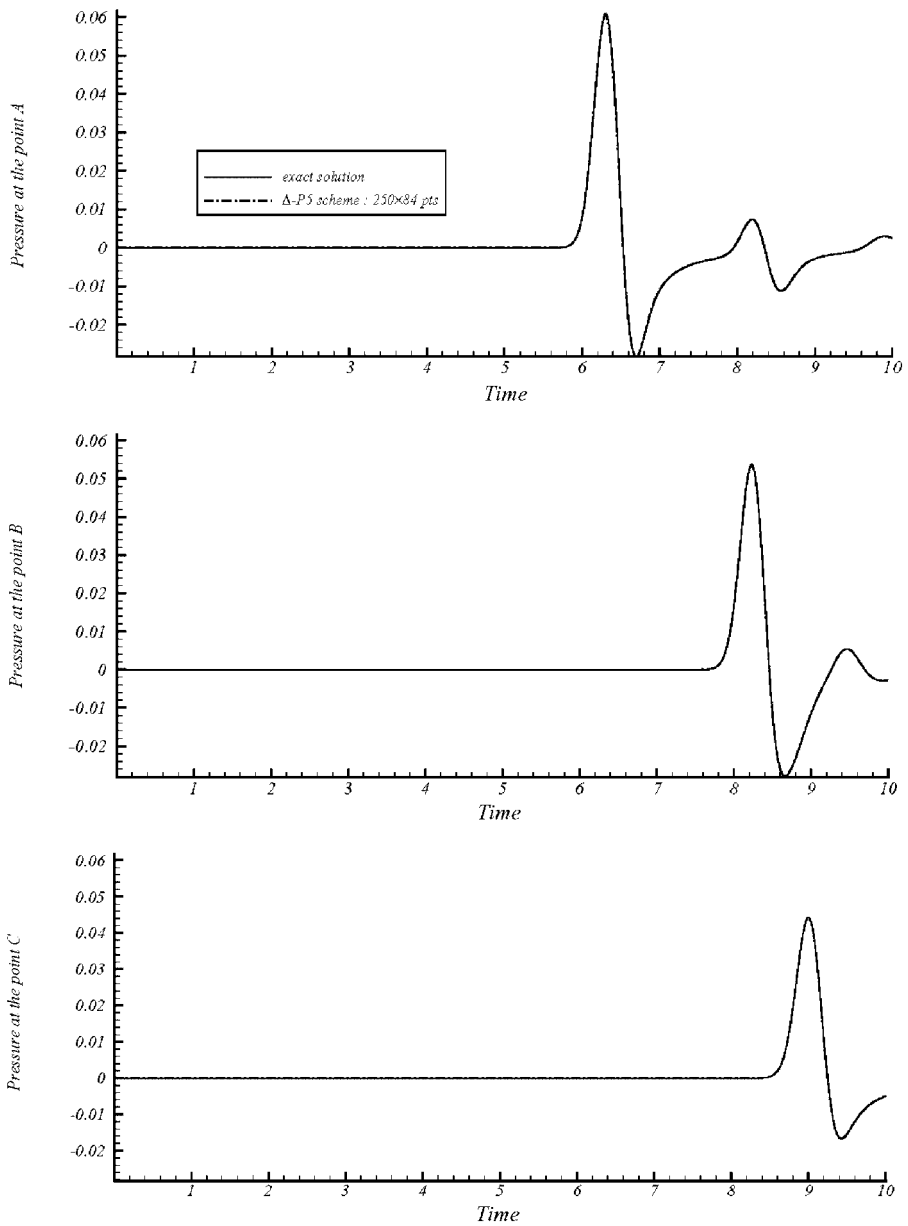

Figure 25. Acoustic scattering from a cylinder. Δ-P5 scheme: time history of pressure fluctuations at points A, B, C. CFL = 0.10 and $p_{\max}$ at point A = 0.0604 (250 × 84 grid points).

DOI: 10.1002/fld

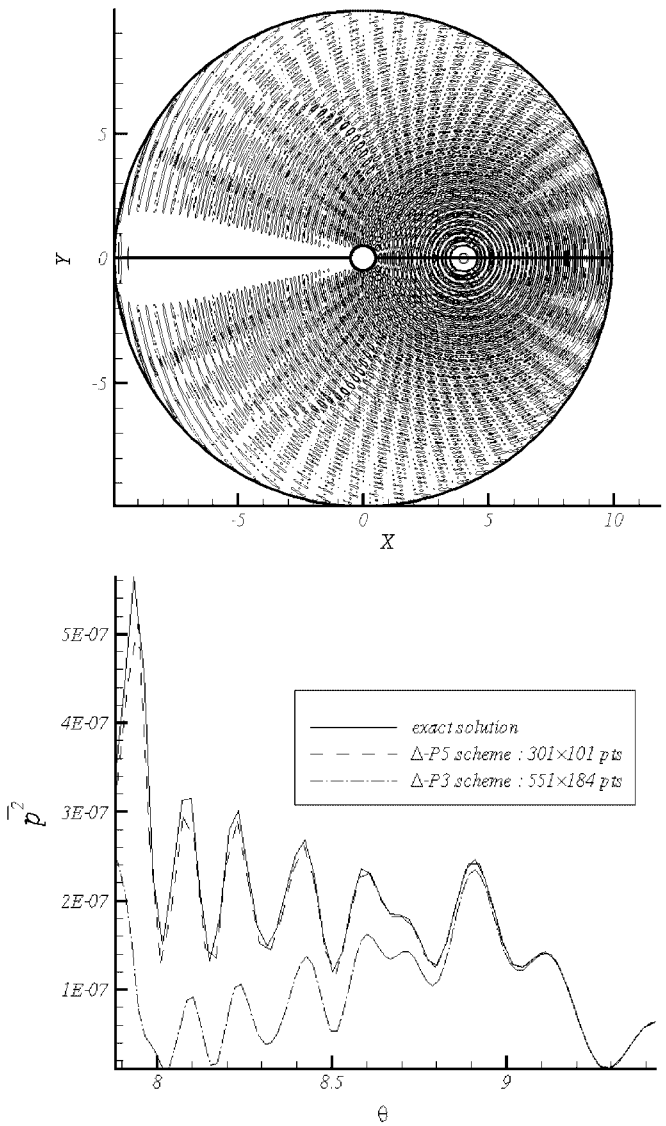

Figure 26. Acoustic scattering from a cylinder: time periodic source ($\omega = 6\pi$). Δ-P5 scheme (301×101 grid points, CFL $= 0.10$, $t_{\text{CPU}} = 7000$ s) and Δ-P3 scheme (551×184 grid points, CFL $= 0.25$, $t_{\text{CPU}} = 8025$ s). Top: contour of pressure acoustic waves at $T=14$; bottom: the computed and the exact directivity at $r=5$.

The initial condition is as follows:

$$\begin{aligned} p(x, y) &= \mathrm{e}^{-20(x^2+y^2)} \\ u(x, y) &= v(x, y) \equiv 0 \end{aligned} \tag{19}$$

The initial conditions for the first derivatives ($r_\varphi \equiv \varphi_x, s_\varphi \equiv \varphi_y$) are easily inferred from (19).

DOI: 10.1002/fld

Now, we present the results of the simulations.

Firstly, we consider the subsonic regime: $M_{xo} = 0.8$, $M_{yo} = 0.5$. The simulation time is $T = 2$ and the CFL number is 0.25. For each case, the maximum norm of the absolute error in pressure profile is calculated.

Figure 27 shows the pressure contours for the acoustic pulse at various time levels in the simulation. The computation domain is divided into a 20×20 mesh. The acoustic pulse leaves smoothly the computation domain through the top, bottom and right boundaries. Moreover, we can remark that the expanding circles are circular so that the computed acoustic wave speed is the same in all directions.

Figure 28 provides comparisons between the computed and the exact pressure waveforms along the x-axis at $y = 0$. As one can note it, the discretization error remains stable as the time grows up.

Now, let us examine the supersonic regime: M_{xo}=2, M_{yo}=1. The CFL number is equal to 0.25.

Firstly, Figure 29 shows pressure contours. These results are obtained on a 20×20 mesh. The acoustic pulse leaves freely the computation domain through the top and right boundaries. This simulation is easier to compute than the previous one since the flow regime is supersonic. The non-reflecting boundary conditions are useless in such a case: all the simple waves, ψ_p, leave the computation domain. For this reason, no reflections are noticed at the boundaries, Figure 29. Lastly, Figure 30 provides a comparison between the computed and the exact solutions, at $T = 0.125$ and 0.375, on a 10×10 mesh: the agreement with the exact solution is very good.

4.6. Generation and radiation of acoustic waves from a 2D shear layer [*20*]

This test case is the category 5 problem of the 2000-CAA-Workshop on benchmark problems [20].

A fixed known acoustic source, $S(t)$, pulsating at a single frequency, ω, is located inside a 2D supersonic jet (Mach number, M_j) with the mean flow parallel to and symmetric about the x-axis. To model this problem, we use the linear aeroacoustic equations, system (8)–(10) (see [3] for a detailed explanation of the procedure).

The non-reflecting boundary conditions are applied at the subsonic boundaries by setting to zero wave strengths entering the computation domain while waves leaving this domain are used to update the boundary values. At supersonic inflow, all variables are set to zero (no ingoing perturbations).

Due to symmetry in the mean flow, only the solution in the upper half-plane is computed. Therefore, symmetry conditions are applied for u, v, p and its derivatives at every point, S, located on the x-axis. The remaining variables are updated by the simple waves leaving the computation domain along the x-axis.

The computation domain is defined by the conditions: $-5 \leqslant x \leqslant 60$ and $0 \leqslant y \leqslant 12$.

A uniformly distributed mesh size, Δx, is used in the x-direction. The mesh is geometrically stretched in the y-direction in order to place more grid points between $y = 0.6$ and 1.8, where $\partial M/\partial y$ is large and the perturbation solution varies significantly.

The discrete time step, Δt, is calculated by using the formula derived in [3]:

$$\Delta t \equiv \mathrm{CFL} \times \min_i \left[\sqrt{\frac{\Delta x \Delta y_i}{\pi}} \times \frac{1}{1 + M(y_i)} \right] \tag{20}$$

Computations are carried out until the dimensionless time $t = 50$, so that the numerical solution becomes periodic in time.

DOI: 10.1002/fld

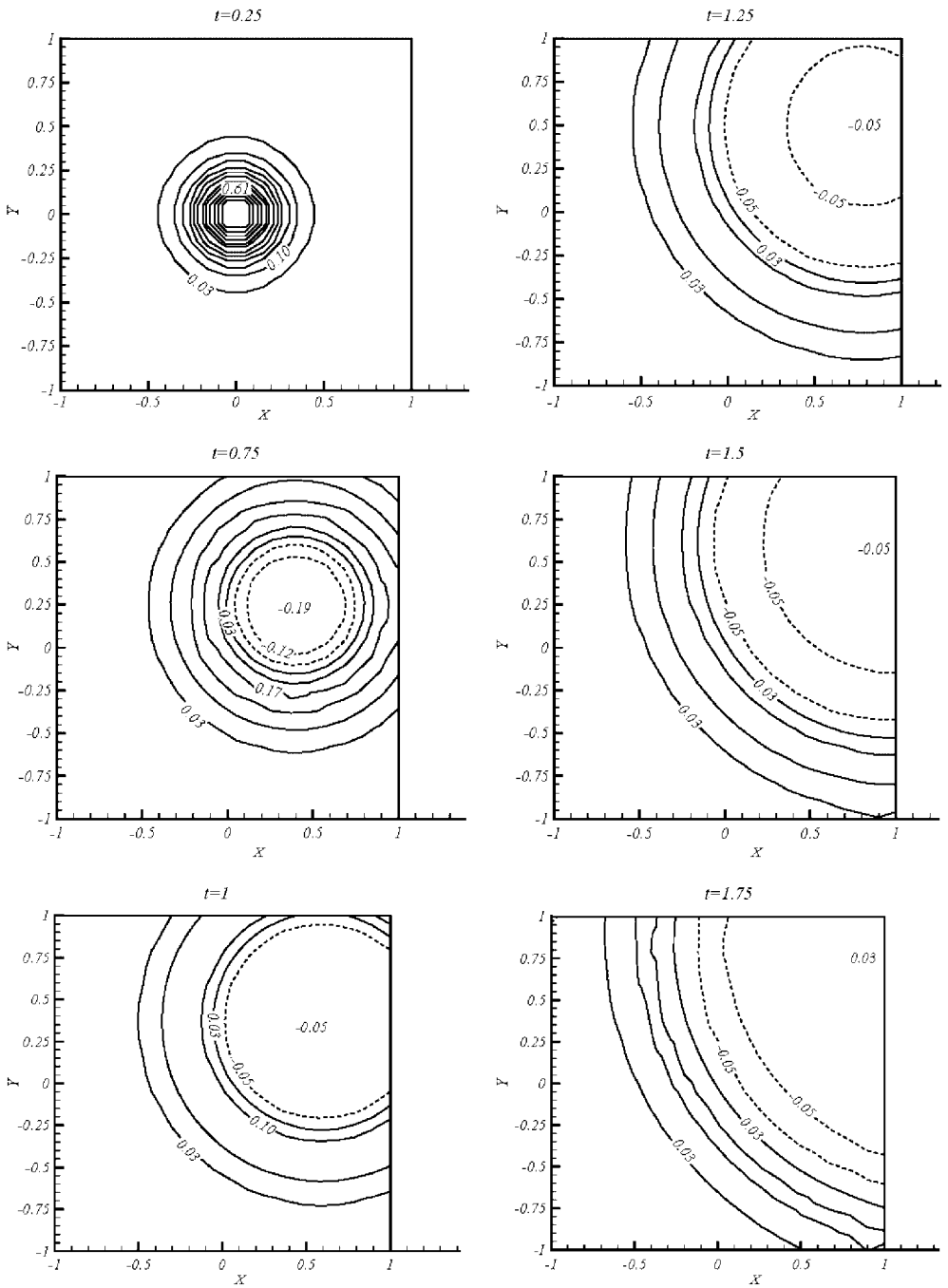

Figure 27. Linear aeroacoustics. Axisymmetric pulse. Pressure contours. Time evolution of the pressure pulse. Δ-P5 scheme: $M_{xo} = 0.8$; $M_{yo} = 0.5$; 20×20 grid points; and CFL $= 0.25$.

DOI: 10.1002/fld

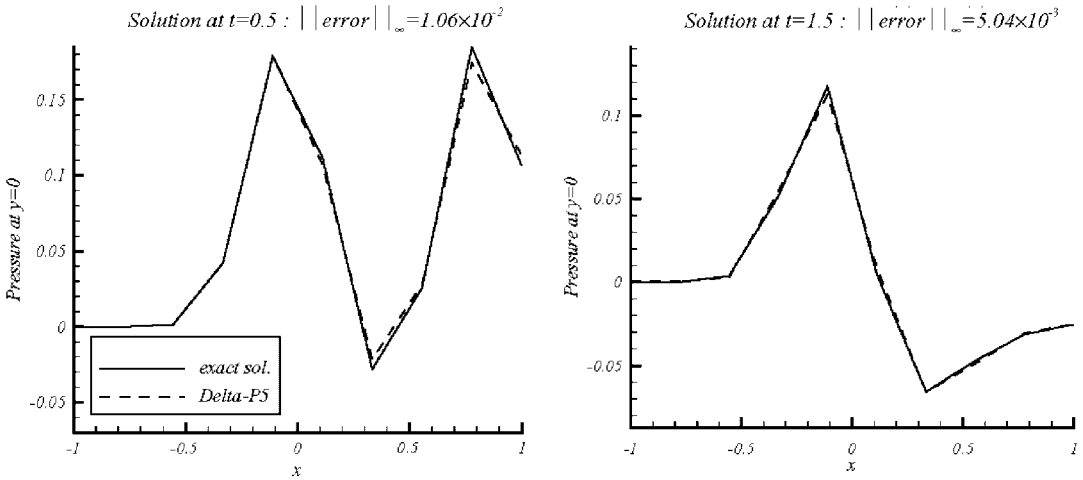

Figure 28. Linear aeroacoustics. Axisymmetric pulse. Pressure waveforms along the x-axis at $y = 0$. Δ-P5 scheme: $M_{xo} = 0.8$; $M_{yo} = 0.5$; 10×10 grid points; and CFL $= 0.25$.

Figure 31 shows the mean quadratic fluctuations of the pressure, $\overline{p^2}$ (averaged over one temporal period), along both lines $y = 10$ and $x = 50$. For this simulation, 80×40 grid points are used (80 points in the x-direction). The mesh is stretched in such a way that six cells are located in the region $0.6 \leqslant y \leqslant 1.8$. The CFL number is taken equal to 0.35, the size of the time step is $\Delta t = 1 \times 10^{-2}$ and the required CPU time is 135 s. Considering the characteristic wavelengths, this grid resolution is equivalent to 11 points per wavelength for the slowest wave and 17 points per wavelength for the fastest one.

The sound intensity is in close agreement with the analytic solution at $x = 50$ [21], while it is underestimated at $y = 10$, mostly near the acoustic source. However, we can compare this first result with the one obtained with the Δ-P3 scheme, [3], on a more refined grid. Figure 32 presents such a comparison (for both schemes, the CFL numbers are optimized in order to get the lowest CPU time possible, while preserving the numerical stability).

The results obtained with the Δ-P3 scheme, are typified by the following values: CFL=0.50, 175×100 grid points and $t_{\mathrm{CPU}} = 950$ s. As can be seen, the solutions obtained are similar. However, the Δ-P5 scheme produces better results in the source region. Moreover, Figure 33 shows us that the convected pressure pulses in the supersonic region are captured in a proper way with the Δ-P5 scheme. Therefore, even on a coarse mesh, the Δ-P5 scheme is more advantageous in terms of accuracy/CPU cost.

Figure 34 presents a new simulation with 150×75 grid points. For this resolution, 16 cells are located within the physical range $0.6 \leqslant y \leqslant 1.8$. This grid resolution is nearly equivalent to 17 points per wavelength for the slowest wave and 50 points for the fastest one. The CFL number is taken equal to 0.35, the time-step size is now $\Delta t = 6.9 \times 10^{-3}$ and the required CPU time is 1000 s to reach the simulation time $t = 50$. The level of the mean square pressure is improved at the line $y = 10$. However, the capturing of the exponential growing of the instability wave across the shear layer is not improved since the sound intensity level remains unchanged at $x = 50$, in comparison with Figure 31. Figure 35 compares the Δ-P5 scheme (80×40 grid points) and the

DOI: 10.1002/fld

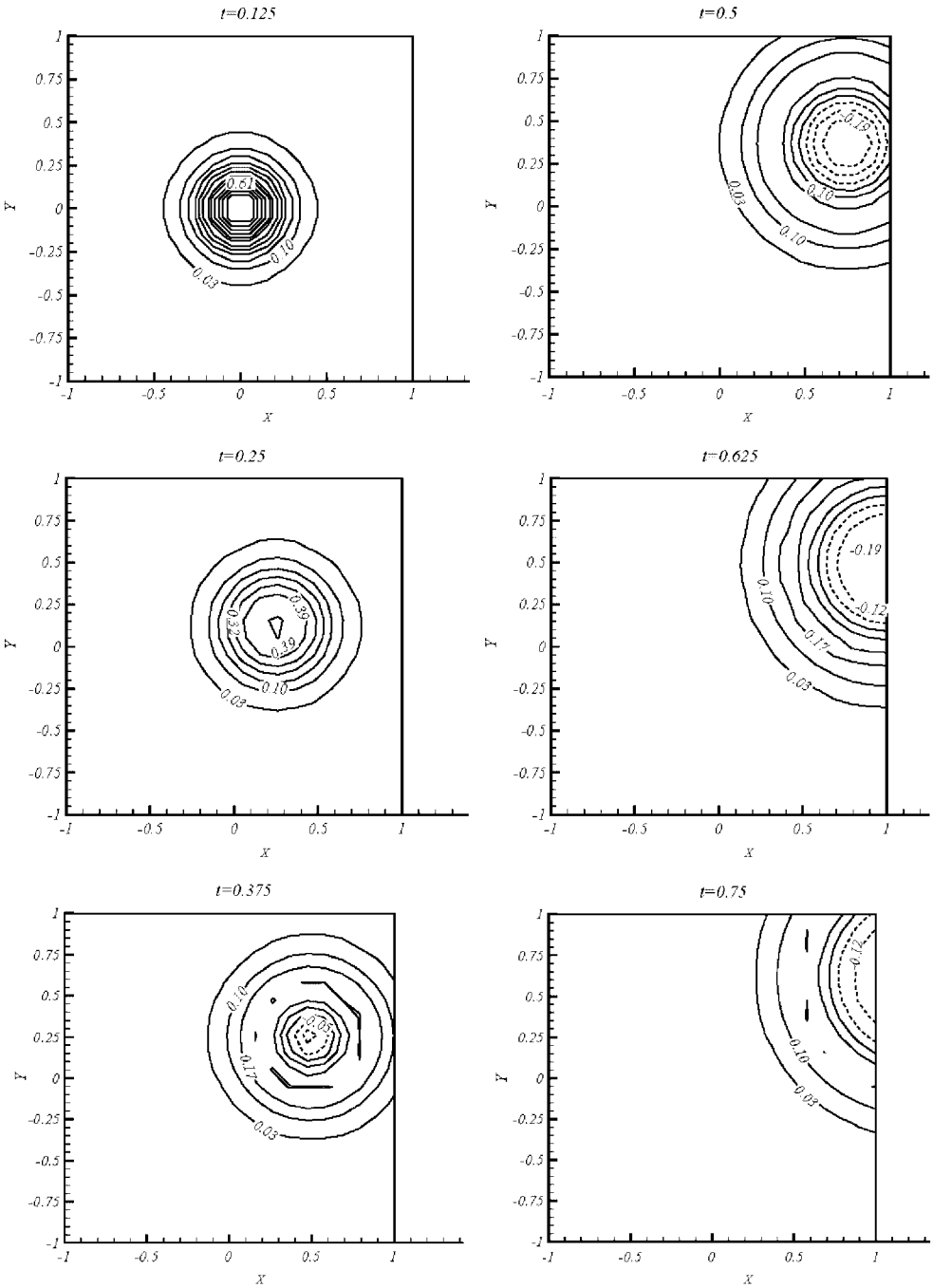

Figure 29. Linear aeroacoustics. Axisymmetric pulse. Pressure contours. Time evolution of the pressure pulse. Δ-P5 scheme: $M_{xo} = 2$; $M_{yo} = 1$; 20×20 grid points; and CFL $= 0.25$.

DOI: 10.1002/fld

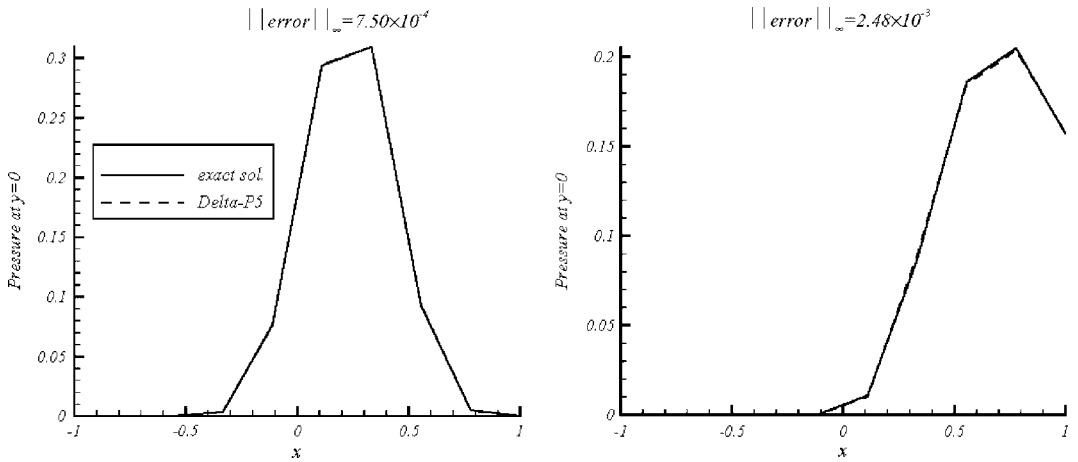

Figure 30. Linear aeroacoustics. Axisymmetric pulse. Pressure waveforms along the x-axis at $y = 0$. Δ-P5 scheme: $M_{xo} = 2$; $M_{yo} = 1$; 10×10 grid points; and CFL = 0.25.

FDo11p/RKo6s scheme (300×150 grid points, CFL = 0.50, $t_{\mathrm{CPU}} = 750$ s): although the numerical results are slightly better than those obtained with the LDDRK6-DRP scheme (see [2]), the spatial resolution of the FDo11p/RKo6s scheme, on such a mesh, remains insufficient, especially in the source region.

Finally, as can be seen from Figure 36, there are no oscillations at the boundaries: this result illustrates the non-reflecting nature of the specific boundary conditions we derived in Section 4.2.

5. SUMMARY AND CONCLUDING REMARKS

The main idea of this paper is that a fifth-order time-marching scheme with a high degree of compactness can be more advantageous in terms of CPU efficiency and memory requirement than algebraically simpler methods of lower order.

A convenient manner to build such a scheme is to rely on the third-order algorithm derived in [1, 3]. Thus, the problem of discretizing a linear hyperbolic system is decomposed into three stages:

- a reconstruction stage;
- a decomposition stage; and
- an evolution stage.

In the reconstruction stage, we use the function and its derivatives in order to define a local spatial interpolator at each step of time. To generate a fifth-order local interpolator, the solution and its first and second derivatives are stored at each vertex defining the discrete cell.

In the decomposition stage, the time derivatives are broken up into simple waves, according to the general ideas of Roe [17] and the specific solutions proposed in [1, 3]. This decomposition allows a suitable upwinding of the discrete stencil by selecting the waves that move towards a given

DOI: 10.1002/fld

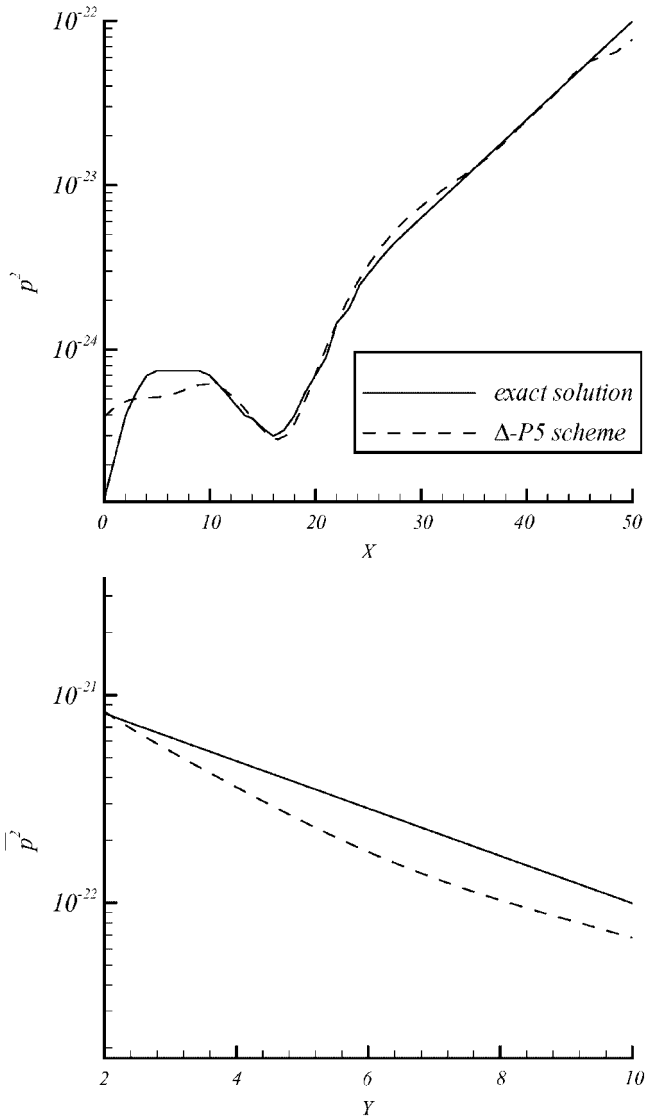

Figure 31. 2D supersonic shear layer. Δ-P5 scheme: intensity of radiated sound, $\overline{p^2}$, along $y = 10$ (top) and $x = 50$ (bottom). $S_{\text{tr}} = 0.14$, $T = 50$, CFL $= 0.35$, 80×40 grid points.

discrete point. Moreover, this upwinding supports a natural treatment of the boundary conditions by distinguishing waves entering and outgoing from the calculation domain.

In the evolution stage, two alternative solutions are investigated. First, the function and its derivatives updated according to a procedure of Cauchy–Kovaleskaya (the 'Δ-P5 scheme'). This is the best solution available for modelling a wave propagation problem, since there is a strong coupling between space and time. However, the 'price to pay' is a greater algebraic complexity. This drawback can even become crucial in 3D calculations.

DOI: 10.1002/fld

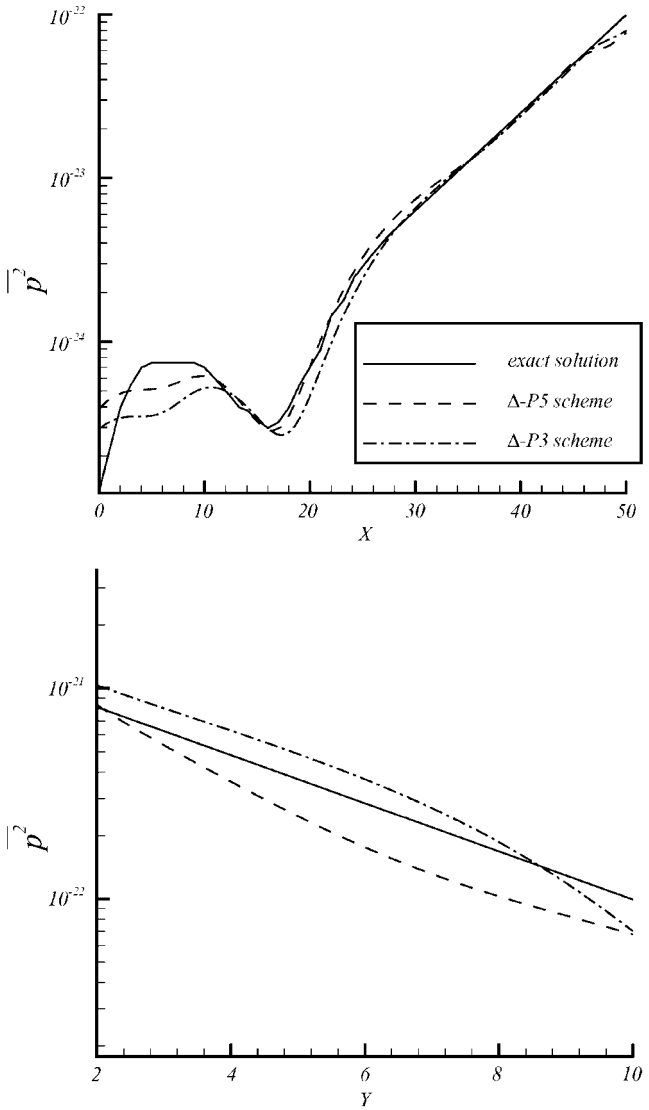

Figure 32. 2D supersonic shear layer. Intensity of radiated sound. Comparisons between the Δ-P5 scheme (80×40 grid points, CFL $= 0.35$, $t_{\mathrm{CPU}} = 135$ s) and the Δ-P3 scheme (175×100 grid points, CFL $= 0.60$, $t_{\mathrm{CPU}} = 792$ s).

An alternative is to replace the Cauchy–Kovaleskaya process by a Runge–Kutta algorithm (the 'RGK-P5 scheme'). The main advantage of doing so is to greatly simplify the resulting discrete equations. However, the compactness in time is lost.

We demonstrated that both resulting schemes produce a low level of dissipative and dispersive errors by necessitating less than 6 CPW. However, the loss of compactness in time contributes in reducing the domain of stability of a numerical scheme: despite its simplified form, the RGK-P5

DOI: 10.1002/fld

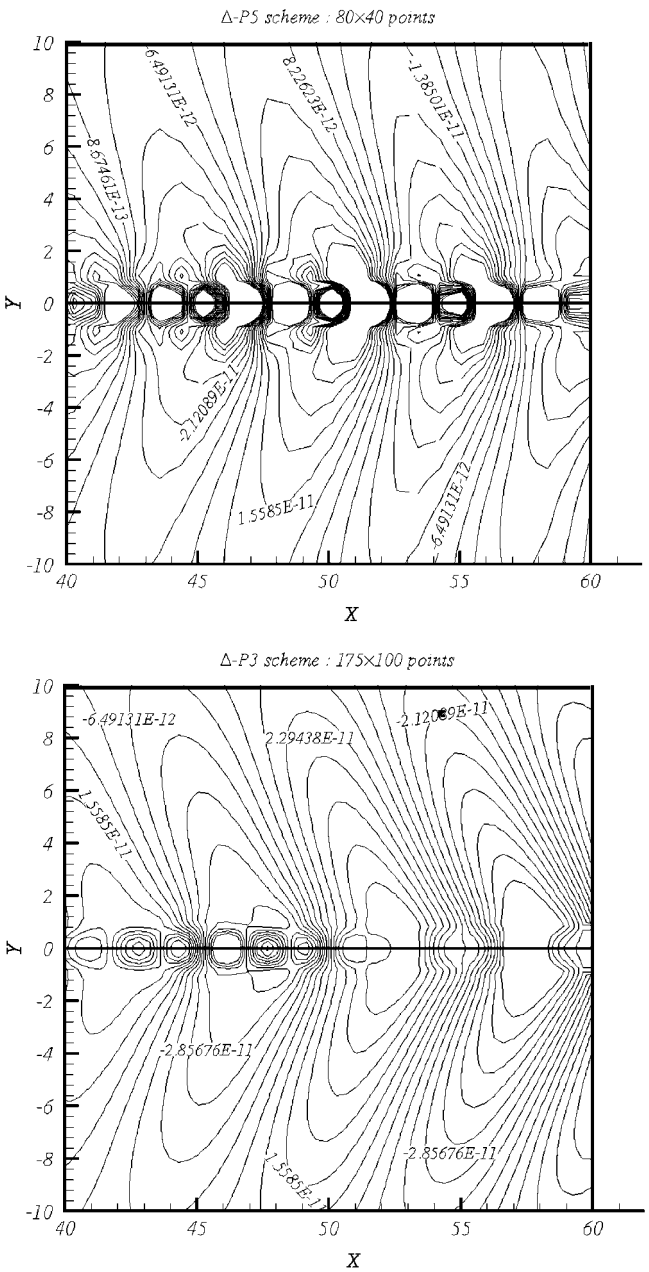

Figure 33. 2D supersonic shear layer. Instantaneous pressure contours at $T = 50$. Zoom in the region of the outflow/radiation boundary. Comparisons between the Δ-P5 scheme and the Δ-P3 scheme.

DOI: 10.1002/fld

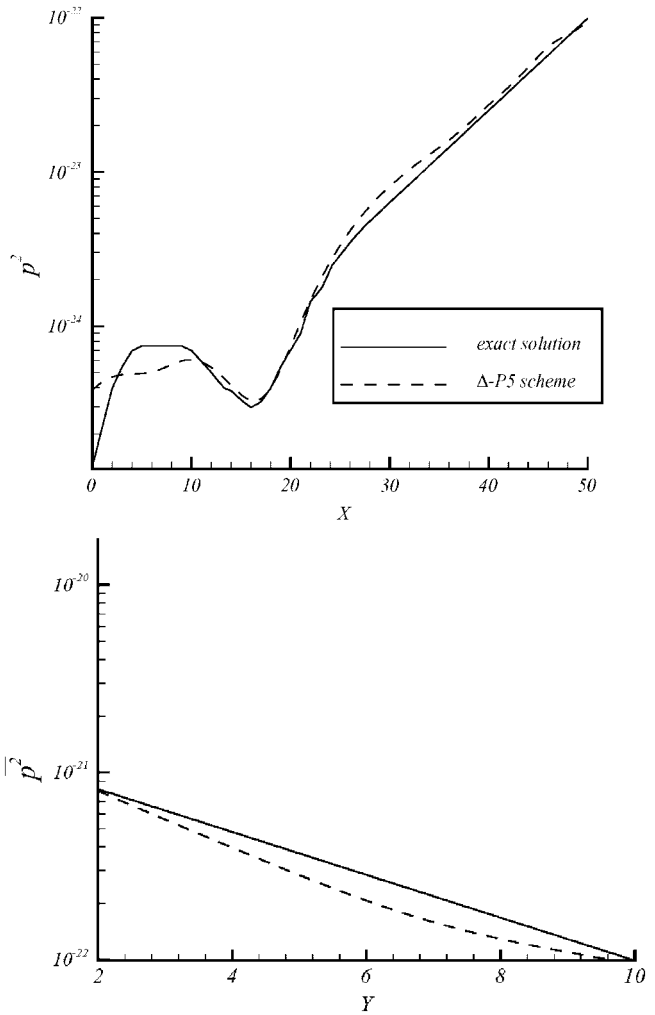

Figure 34. 2D supersonic shear layer. Δ-P5 scheme: intensity of radiated sound, $\overline{p^2}$, along $y = 10$ (top) and $x = 50$ (bottom). $S_{\text{tr}} = 0.14$, $T = 50$, CFL $= 0.35$, 150×75 grid points.

scheme is more costly in terms of CPU time than the Δ-P5 scheme. Indeed, compared with the latter one, the RGK-P5 scheme necessitates a CPU time 40% greater for 2D calculations. However, we demonstrated that the RGK-P5 scheme remains more advantageous than centred finite-difference methods such as the FDo11p/RKo6s scheme [6] or the Hermitian third-order scheme: the 'Δ-P3 scheme' [1, 3]. The Δ-P3 scheme is well suited for linear wave propagation. However, this paper demonstrates that for strongly oscillating solutions, the Δ-P5 or the RGK-P5 scheme becomes largely preferable.

Consequently, depending on the priority, three possibilities henceforth exist to model linear wave propagation in a different way as classical methods already existing. If the solution is weakly

DOI: 10.1002/fld

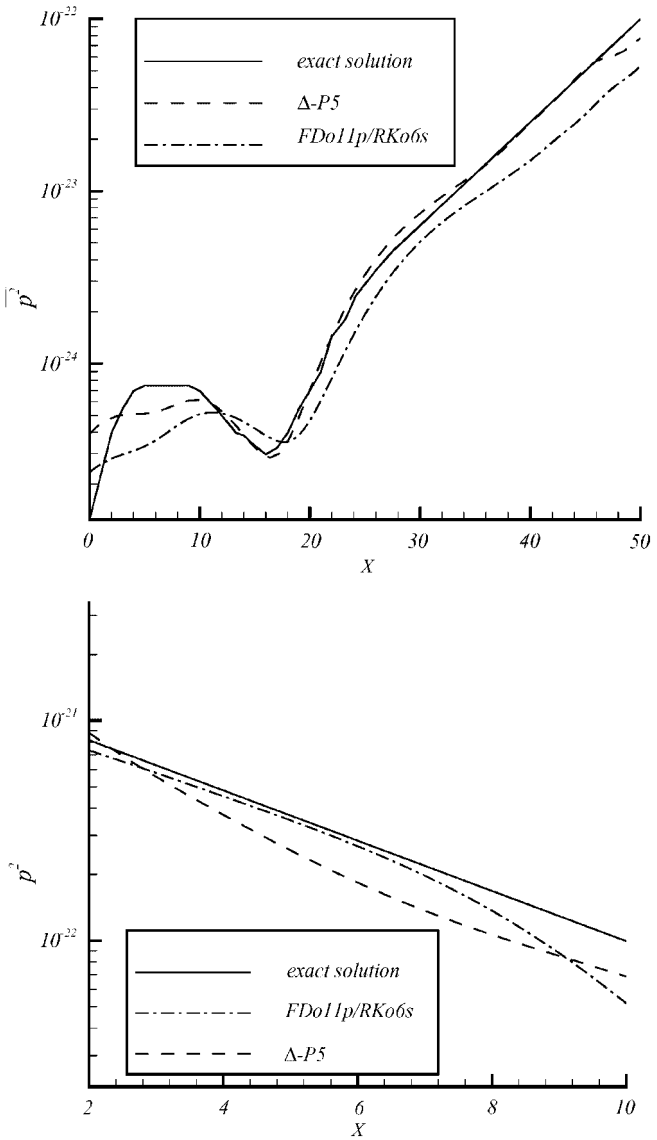

Figure 35. 2D supersonic shear layer. Intensity of radiated sound. Comparisons between the Δ-P5 scheme (150×75 grid points, CFL $= 0.35$, $t_{\mathrm{CPU}} = 1000$ s) and the FDo11p/RKos scheme (300×150 grid points, CFL$=0.50$, $t_{\mathrm{CPU}} = 750$ s).

varying in time or if the computational cost is of little matter, then the Δ-P3 scheme is well suited. On the other hand, with less than 6 CPW required to lower the amplitude and phase errors per time step under 0.01%, the Δ-P5 or the RGK-P5 scheme give a good solution to save computation time and memory. In spite of a reduced domain of stability, the RGK-P5 scheme supplies the best answer for modelling with a high accuracy the acoustics or the aeroacoustics

DOI: 10.1002/fld

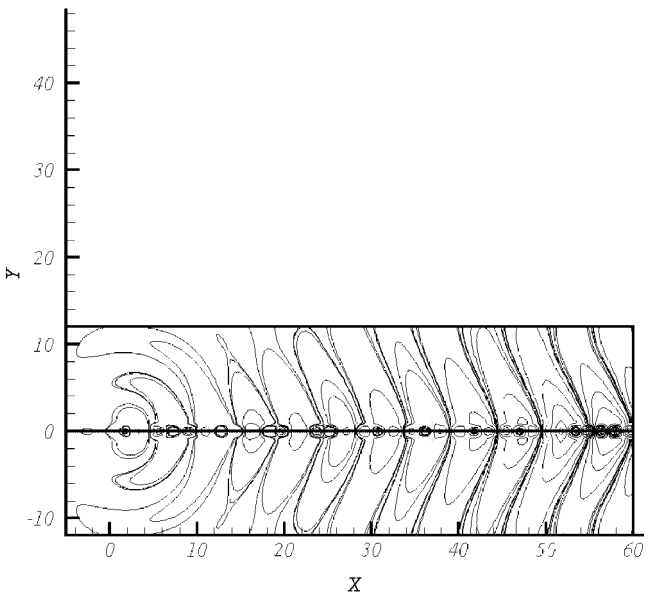

Figure 36. 2D supersonic shear layer. Δ-P5 scheme: instantaneous pressure contours at $t = 50$. Twenty-one isolines: 6.5×10^{-11} to -1.40×10^{-10} and 150×75 grid points.

in multi-dimensional configurations. However, such a method is still perfectible. More work is needed in order to extend its domain of stability. Probably, the optimization of the coefficients used in the time integration, while relaxing the constraint of time accuracy, would make it possible to ensure this.

In a future work, the class of Hermitian algorithms we derived will be extended to the discretization of nonlinear hyperbolic systems of equations.

APPENDIX A: ALGEBRAIC FORMULAE FOR CALCULATING THE DERIVATIVES OF THE Δ-P5 OR THE RGK-P5 SCHEME ON A NON-UNIFORM CARTESIAN GRID

We provide calculations for the space derivatives on a non-uniform Cartesian mesh, by using the generic stencil defined by Figure 13. To begin, we define on the generic cell $(1, 2, 3, 4)$ the following form for the local spatial interpolator:

$$\widetilde{\varphi}(x, y) \equiv \text{complete fifth-order polynomial } + c_{22}x^5y + c_{23}x^3y^3 + c_{24}xy^5 \tag{A1}$$

The generic variable, φ, identified with $\widetilde{\varphi}(x, y)$ at a given discrete point, stands for the acoustic variables u, v, p when discretizing the acoustic or aeroacoustic equations.

DOI: 10.1002/fld

Then, the third derivatives of $\widetilde{\varphi}$ at the point 2, are

$$\widetilde{\varphi}_{3x}|_2 = \frac{3}{\Delta x^3} \times [20(\varphi_2 - \varphi_1) - \Delta x \times (8r_\varphi|_1 + 12r_\varphi|_2) + \Delta x^2 \times (3r2_\varphi|_2 - r2_\varphi|_1)]$$

$$\widetilde{\varphi}_{xxy}|_2 = \frac{1}{\Delta x^2 \Delta y} \times \begin{bmatrix} 6(\varphi_2 + \varphi_4 - \varphi_5 - \varphi_1) + 2\Delta x \times r_\varphi|_4 + 2r_\varphi|_5 - 2r_\varphi|_2 - r_\varphi|_1) \\ + \Delta x^2 \times (r2_\varphi|_2 - r2_\varphi|_5) + 6\Delta y \times (s_\varphi|_1 - s_\varphi|_2) \\ + 2\Delta x \Delta y \times (w2_\varphi|_1 + 2w2_\varphi|_2) \end{bmatrix}$$

$$\widetilde{\varphi}_{xyy}|_2 = \frac{1}{\Delta x \Delta y^2} \times \begin{bmatrix} 6(\varphi_2 + \varphi_4 - \varphi_5 - \varphi_1) + 2\Delta y \times (s_\varphi|_4 + 2s_\varphi|_1 - 2s_\varphi|_2 - s_\varphi|_5) \\ + 6\Delta x (r_\varphi|_5 - r_\varphi|_2) + \Delta y^2 \times (s2_\varphi|_2 - s2_\varphi|_1) \\ + 2\Delta x \Delta y \times (w2_\varphi|_5 + 2w2_\varphi|_2) \end{bmatrix} \tag{A2}$$

$$\widetilde{\varphi}_{3y}|_2 = \frac{3}{\Delta y^3} \times [20(\varphi_2 - \varphi_5) - \Delta y \times (8s_\varphi|_5 + 12s_\varphi|_2) + \Delta y^2 \times (3s2_\varphi|_2 - s2_\varphi|_5)]$$

At the same point, the fourth derivatives are

$$\begin{aligned}
\frac{\Delta x^4}{12} \times \widetilde{\varphi}_{4x}|_2 &= 30(\varphi_2 - \varphi_1) + \Delta x^2 (3r2_\varphi|_2 - 2r2_\varphi|_1) - 2\Delta x \times (7r_\varphi|_1 + 8r_\varphi|_2) \\
-\frac{\Delta x^3 \Delta y}{3} \times \widetilde{\varphi}_{3xy}|_2 &= 24(\varphi_2 - \varphi_1 + \varphi_5 - \varphi_4) + \Delta x^2 [3(r2_\varphi|_5 - r2_\varphi|_2) \\
&\quad + (r2_\varphi|_1 - r2_\varphi|_4)] + 4\Delta y (s_\varphi|_2 - s_\varphi|_1) \\
&\quad - 2\Delta x \Delta y (w2_\varphi|_1 + w2_\varphi|_2) + \Delta x [14(r_\varphi|_2 - r_\varphi|_5) \\
&\quad + 10(r_\varphi|_1 - r_\varphi|_4)] \\
-\frac{\Delta x^2 \Delta y^2}{4} \times \widetilde{\varphi}_{2x2y}|_2 &= 9(\varphi_1 - \varphi_2 + \varphi_5 - \varphi_4) - \Delta x \Delta y [w2_\varphi|_4 + 4w2_\varphi|_2 \\
&\quad + 2(w2_\varphi|_5 + w2_\varphi|_1)] + 3\Delta x [2(r_\varphi|_2 - r_\varphi|_5) + (r_\varphi|_1 - r_\varphi|_4)] \\
&\quad + 3\Delta y [(s_\varphi|_5 - s_\varphi|_4 + 2(s_\varphi|_2 - s_\varphi|_1)] \\
-\frac{\Delta x \Delta y^3}{3} \times \widetilde{\varphi}_{x3y}|_2 &= 24(\varphi_2 - \varphi_1 + \varphi_5 - \varphi_4) + \Delta y^2 [(s2_\varphi|_5 - s2_\varphi|_4) \\
&\quad + 3(s2_\varphi|_1 - s2_\varphi|_2)] + 4\Delta x (r_\varphi|_2 - r_\varphi|_5) \\
&\quad - 2\Delta x \Delta y (w2_\varphi|_5 + w2_\varphi|_2) + \Delta y [14(s_\varphi|_2 - s_\varphi|_1) \\
&\quad + 10(s_\varphi|_5 - s_\varphi|_4)] \\
\frac{\Delta y^4}{12} \times \widetilde{\varphi}_{4y}|_2 &= 30(\varphi_2 - \varphi_5) + \Delta y^2 (3s2_\varphi|_2 - 2s2_\varphi|_5) - 2\Delta y \times (7s_\varphi|_5 + 8s_\varphi|_2)
\end{aligned} \tag{A3}$$

DOI: 10.1002/fld

Finally, the fifth derivatives are

$$
\begin{aligned}
\frac{\Delta x^5}{60}\times\widetilde{\varphi}_{5x}|_2 &= 12(\varphi_2-\varphi_1)+\Delta x^2(r2_\varphi|_2-r2_\varphi|_1)-6\Delta x\times(r_\varphi|_1+r_\varphi|_2)\\
-\frac{\Delta x^4\Delta y}{12}\times\widetilde{\varphi}_{4xy}|_2 &= 30(\varphi_5+\varphi_1-\varphi_2-\varphi_4)+\Delta x^2[2(r2_\varphi|_1-r2_\varphi|_4)\\
&\quad +3(r2_\varphi|_5-r2_\varphi|_2)]+\Delta x[16(r_\varphi|_2-r_\varphi|_5)+14(r_\varphi|_1-r_\varphi|_4)]\\
-\frac{\Delta x^3\Delta y^2}{12}\times\widetilde{\varphi}_{3x2y}|_2 &= 6(\varphi_1-\varphi_2+\varphi_5-\varphi_4)-\Delta x\Delta y[w2_\varphi|_4+w2_\varphi|_5\\
&\quad +2(w2_\varphi|_2+w2_\varphi|_1)]+\Delta x[3(r_\varphi|_1-r_\varphi|_5+r_\varphi|_2-r_\varphi|_4)]\\
&\quad +2\Delta y[2(s_\varphi|_2-s_\varphi|_1)+(s_\varphi|_5-s_\varphi|_4)]\\
-\frac{\Delta x^2\Delta y^3}{12}\times\widetilde{\varphi}_{2x3y}|_2 &= 6(\varphi_1-\varphi_2+\varphi_5-\varphi_4)-\Delta x\Delta y[w2_\varphi|_4+w2_\varphi|_1\\
&\quad +2(w2_\varphi|_2+w2_\varphi|_5)]+\Delta y[3(s_\varphi|_5-s_\varphi|_1+s_\varphi|_2-s_\varphi|_4)]\\
&\quad +2\Delta x[2(r_\varphi|_2-r_\varphi|_5)+(r_\varphi|_1-r_\varphi|_4)]\\
\frac{\Delta x\Delta y^4}{12}\times\widetilde{\varphi}_{x4y}|_2 &= 30(\varphi_2-\varphi_1+\varphi_4-\varphi_5)+\Delta y^2[2(s2_\varphi|_4-s2_\varphi|_5\\
&\quad +3(s2_\varphi|_2-s2_\varphi|_1)]+\Delta y[16(s_\varphi|_1-s_\varphi|_2)+14(s_\varphi|_4-s_\varphi|_5)]\\
-\frac{\Delta y^5}{60}\times\widetilde{\varphi}_{5y}|_2 &= 12(\varphi_5-\varphi_2)+\Delta y^2(s2_\varphi|_5-s2_\varphi|_2)+6\Delta y\times(s_\varphi|_5+s_\varphi|_2)
\end{aligned}
\tag{A4}
$$

These formulae are used to calculate the wave strengths ψ_ℓ in (15) and (16). The formulae for the discrete points (1), (3) and (4) are simply deduced by circular permutation.

ACKNOWLEDGEMENT

The author wishes to thank with recognition, Dr G. B. Deng (Ecole Centrale de Nantes—France) to have provided a technical support essential to this work.

REFERENCES

1. Capdeville G. A new category of Hermitian upwind schemes for computational acoustics. *Journal of Computational Physics* 2005; **210**:133–170.
2. Tam CKW. Computational aeroacoustics: issues and methods. *AIAA Journal* 1995; **13**(10):1788–1796.
3. Capdeville G. A new category of Hermitian upwind schemes for computational acoustics—II. Two-dimensional aeroacoustics. *Journal of Computational Physics* 2006; **217**:530–562.
4. Hu FQ, Hussaini MY, Manthey JL. Low-dissipation and low-dispersion Runge–Kutta schemes for computational acoustics. *Journal of Computational Physics* 1996; **124**:177–191.
5. Tam CKW, Webb JC. Dispersion-relation-preserving finite difference schemes for computational acoustics. *Journal of Computational Physics* 1993; **107**:262–281.
6. Bogey C, Bailly C. A family of low dispersive and low dissipative explicit schemes for flow and noise computations. *Journal of Computational Physics* 2004; **194**:194–214.
7. Calvo M, Franco JM, Randez L. A new minimum storage Runge–Kutta scheme for computational acoustics. *Journal of Computational Physics* 2004; **201**:1–12.

DOI: 10.1002/fld

8. Lele SK. Compact finite-difference scheme with spectral-like resolution. *Journal of Computational Physics* 1992; **103**:16–42.
9. Ashcroft G, Zhang X. Optimized prefactored compact schemes. *Journal of Computational Physics* 2003; **190**: 459–477.
10. Canuto C, Hussaini MY, Quarteroni A, Zang TA. *Spectral Methods in Fluid Mechanics*. Springer: Berlin, 1987.
11. Zhow YC, Wei GW. High resolution conjugate filters for the simulation of flows. *Journal of Computational Physics* 2003; **189**:159–179.
12. Harten A, Engquist B, Osher S, Chakravarthy S. Uniform high-order accurate essentially non-oscillatory schemes, III. *Journal of Computational Physics* 1987; **71**:231–303.
13. Liu XD, Osher S, Chan T. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics* 1994; **115**:200–212.
14. Wang ZJ, Chen RF. Optimized weighted essentially non-oscillatory schemes for linear waves with discontinuity. *Journal of Computational Physics* 2001; **174**:381–404.
15. Qiu J, Shu CW. Hermite WENO schemes and their application as limiters for Runge–Kutta discontinuous Galerkin method: one-dimensional case. *Journal of Computational Physics* 2003; **196**:115–135.
16. Cockburn B, Hou S, Shu CW. The Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case. *Mathematics of Computation* 1990; **54**:545–581.
17. Roe PL. Discrete models for the numerical analysis of time-dependent multi-dimensional gas dynamics. *Journal of Computational Physics* 1986; **63**:458–476.
18. Tam CKW, Hardin JC (eds). *Second Computational Aeroacoustics Workshop on Benchmark Problems*, NASA Conference Paper 3352, June 1997.
19. Dowling AP, Ffowcs-Williams JE. *Sound and Source of Sound*. Wiley: New York, 1983.
20. *Third Computational Aeroacoustics Workshop on Benchmark Problems*, NASA/CP-2000-209790, 2000.
21. Dahl MD. Solution to the category 5 problem: generation and radiation of acoustic waves from a 2D shear layer. *Third CAA Workshop on Benchmark Problems*, NASA/CP-2000-209790, 2000; 87–92.

DOI: 10.1002/fld