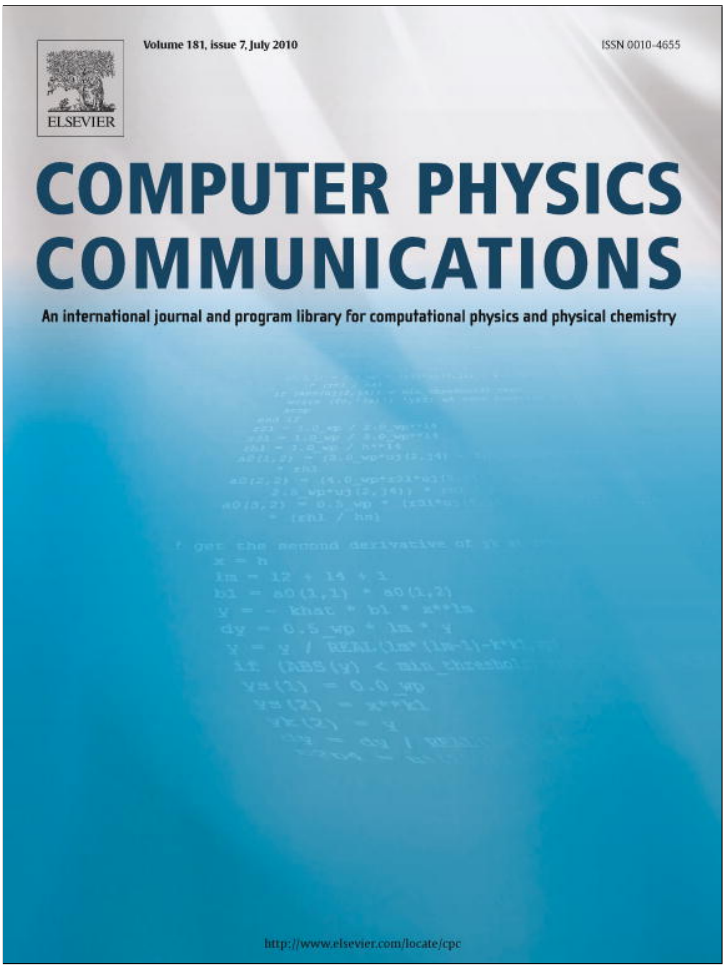

Computer Physics Communications 181 (2010) 1186–1194

Contents lists available at ScienceDirect

Computer Physics Communications

www.elsevier.com/locate/cpc

Higher order discontinuous Galerkin method for acoustic pulse problem

A.V. Wolkov [a], N.B. Petrovskaya [b,*]

[a] Central Aerohydrodynamic Institute, Zhukovsky, Moscow Region, 140180, Russia
[b] University of Birmingham, B15 2TT, Birmingham, UK

ARTICLE INFO

Article history:
Received 17 February 2009
Received in revised form 18 January 2010
Accepted 8 March 2010
Available online 11 March 2010

Keywords:
Computational aeroacoustics
Discontinuous Galerkin
Higher order schemes

ABSTRACT

We discuss the methodology of the validation of a higher order discontinuous Galerkin (DG) scheme for acoustic computations. That includes an accurate definition of the exact solution in the problem as well as careful study of convergence properties of a higher order DG scheme for a chosen acoustic problem. The efficiency of a higher order scheme will be confirmed for computations on coarse meshes.

1. Introduction

In past decades methods of computational aeroacoustics (CAA) have received a lot of attention in scientific computing community. While the complexity of modern CAA problems is well documented (e.g., see [1]), the design of adequate numerical techniques is still in progress. In particular, CAA problems require numerical methods that allow for calculations with a small dissipation and dispersion error. Thus the current attention is focused on higher order discretization schemes that can meet the above requirement without using extensive computational resource.

A discontinuous Galerkin (DG) discretization is a higher order scheme that nowadays is intensively used in many computational applications. First introduced for the transport equation and further developed for many applications (see [2] for the review of DG schemes), the DG method is a finite element scheme which uses piecewise polynomial approximation in space. The discretization also involves an approximate Riemann solver, since the approximate solution is discontinuous at grid interfaces. Among the advantages of the method are a compact scheme stencil that allows one to vary the order of approximation on each grid cell, easy parallel implementation of the scheme, and flexibility in choice of a computational mesh. The above advantages make a higher order DG scheme potentially attractive for CAA applications and in recent years intensive study of higher order DG schemes for CAA problems has been performed [3–5,8]. In particular, the scheme appeared to be attractive for a numerical solution of the linearized Euler equations (LEE) [6,7], as the formulation of a DG method naturally stems from physical properties of hyperbolic equations. However, despite the careful study of the LEE, the need still remains to investigate systematically a higher order DG discretization for the full Euler equations, as some important acoustic phenomena such as intensity of sound sources can be captured only with the numerical solution of fully nonlinear equations [8].

Implementation of higher order schemes for the numerical solution of the Euler equations is a challenging issue, especially when unstructured grids are generated in the problem. Most of industrial codes used currently for aerodynamic computations on unstructured grids exploit second order accurate finite volume schemes. For aeroacoustics computations fourth order finite volume schemes have been designed and successfully exploited (e.g., see [10,11]), but their implementation is restricted by carefully generated structured grids. Meanwhile, one of basic problems with the use of higher order finite volume schemes on unstructured meshes is that the scheme requires an expanded stencil for the discretization of the derivatives. This requirement rises a number of still unresolved questions such as the increased cost of the algorithm, the parallelization technique and the accuracy of a discretization on expanded stencils when anisotropic unstructured grids should be generated for complex geometries [12,13]. Thus in our paper we explore a higher order DG discretization for the Euler equations to investigate whether the scheme can be considered as a reliable alternative to existing approaches.

The methodology of the validation of a DG scheme for aeroacoustic computations should include an accurate definition of the exact solution in the problem as well as careful study of convergence properties of the scheme for an acoustic problem under consideration. In our paper we provide a comprehensive study of a higher order DG scheme for an acoustic pulse problem, where the solution error will be carefully computed to evaluate the accu-

* Corresponding author at: School of Mathematics, University of Birmingham, Edgbaston, Birmingham, B15 2TT, United Kingdom. Tel.: +44 121 4146591.
E-mail address: n.b.petrovskaya@bham.ac.uk (N.B. Petrovskaya).

0010-4655/$ – see front matter
doi:10.1016/j.cpc.2010.03.003

racy and efficiency of the scheme. The widespread ideology behind many industrial aerodynamic codes is that a lower order scheme combined with grid refinement allows one to achieve the same accuracy as a higher order scheme would provide on a coarse mesh, while a more significant computational effort is required in the latter case. Thus we compare the results of higher order DG computations on coarse meshes with those for a second order accurate scheme on fine meshes to demonstrate that implementation of a higher order scheme results in better accuracy in comparison with a second order scheme that has the same number of degrees of freedom. This result is very important for CAA applications, as it allows one to reduce the computational effort required for solution of acoustic problems.

Another issue discussed in the paper is the choice of a time marching scheme to solve a semi-discrete problem obtained as the result of a spatial discretization. Acoustic problems usually require computations over a big time interval and a time marching scheme should provide accurate results as time progresses. Implementation of an explicit time integration scheme such as a Runge–Kutta (R–K) method may not always lead to accurate computations if the order of the time marching scheme is not consistent with the order of the spatial discretization [2]. However, using a higher order total variation diminishing (TVD) R–K scheme is a computationally intensive task and some authors try to avoid it by combining a higher order DG discretization with a lower order TVD R–K scheme. Hence, we compare a 6-stage TVD R–K method with a 3-stage TVD R–K method that has previously been employed in acoustic computations [7,8]. The results of our study demonstrate the advantage of a higher order TVD R–K scheme that is consistent with a higher order spatial DG discretization.

2. Discontinuous Galerkin discretization for the Euler equations

A problem of acoustic pulse propagation requires solution of the Euler equations in a two-dimensional domain D. We consider the Euler equations written in the domain D in conservative form,

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_1(\mathbf{U})}{\partial x} + \frac{\partial \mathbf{F}_2(\mathbf{U})}{\partial y} = 0, \tag{1}$$

where the components of vector $\mathbf{U} = (\rho, \rho u, \rho v, \rho E)^T$ are mass, x-momentum, y-momentum, and energy values per unit volume of gas. The inviscid fluxes in Eqs. (1) are

$$\mathbf{F}_1(\mathbf{U}) = \left(\rho u, \rho u^2 + p, \rho uv, \rho uH\right)^T \quad \text{and}$$

$$\mathbf{F}_2(\mathbf{U}) = \left(\rho v, \rho uv, \rho v^2 + p, \rho vH\right)^T,$$

where p is the gas pressure. The total enthalpy per unit mass H is defined as

$$H = \frac{\gamma}{\gamma - 1}\frac{p}{\rho} + \frac{1}{2}\left(u^2 + v^2\right),$$

where $\gamma = c_p/c_v$ is the specific heat ratio obtained from the equation of state considered for a perfect gas.

Let us now introduce a computational grid G as a set of non-overlapping triangles e_i, $i = 1, 2, \ldots, N$, in the domain D. The details of grid generation will be discussed below in the text for numerical test cases under consideration. A DG discretization scheme defines the approximate solution $u_h(t, x, y)$ for each variable $u(t, x, y)$ on each grid cell e_i as

$$u_h(t, x, y) = \sum_{m=0}^{M} u_m(t)\phi_m(x, y),$$
$$m = 0, 1, \ldots, M, \; x, y \in e_i, \tag{2}$$

where the basis functions are defined as $\phi_m(x, y) = (x - x_{0i})^\alpha (y - y_{0i})^\beta$, $\alpha + \beta = 0, 1, \ldots, K$. The functions $\phi_m(x, y)$ are piecewise polynomial, as they are only defined within the grid cell e_i. For a cell-centered DG scheme, x_{0i} and y_{0i} are the coordinates of the grid cell centroid. The number M of expansion coefficients is determined by the maximum polynomial degree K as $M = M(K) = (K + 1)(K + 2)/2$.

In the DG method a weak formulation of the problem is used to find the vector $\mathbf{U}(t, x, y)$. The test functions belong to the same approximating space as the basis functions. For the sake of discussion, it is convenient to write the Euler equations in condensed form

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \vec{\mathbf{F}}(\mathbf{U}) = 0, \tag{3}$$

where a hypervector $\vec{\mathbf{F}} = (\mathbf{F}_1(\mathbf{U}), \mathbf{F}_2(\mathbf{U}))^T$. The vector equation (3) is then multiplied by a test function $\phi_l(x, y)$ and is integrated by parts over the cell e_i to arrive at

$$\frac{d}{dt}\int_{e_i} \mathbf{U}\phi_l \, d\Omega + \oint_{\partial e_i} \vec{\mathbf{F}}\phi_l \, d\mathbf{s} - \int_{e_i} \vec{\mathbf{F}}\nabla\phi_l \, d\Omega = 0,$$
$$l = 0, 1, \ldots, M, \tag{4}$$

where $d\mathbf{s} = \partial e_i \mathbf{n}$, $\mathbf{n} = (n_x, n_y)$, is the outward unit normal vector, and the notation ∂e_i is used for the boundary of the cell e_i. Further substitution of the approximate solution (2) into the integrals (4) results in

$$\frac{d}{dt}\int_{e_i} \mathbf{U}_h\phi_l \, d\Omega + \oint_{\partial e_i} \vec{\mathbf{H}}\left(\mathbf{U}^-, \mathbf{U}^+\right)\phi_l \, d\mathbf{s} - \int_{e_i} \vec{\mathbf{F}}\nabla\phi_l \, d\Omega = 0,$$
$$l = 0, 1, \ldots, M. \tag{5}$$

Since the approximate solution $\mathbf{U}_h$ is discontinuous at any grid edge, we need to compute a numerical flux $\vec{\mathbf{H}}(\mathbf{U}^-, \mathbf{U}^+)$ to approximate the continuous flux $\vec{\mathbf{F}}$ at cell interfaces. For this purpose we implement the Roe numerical flux in the problem. Namely, the fluxes at cell interfaces are computed as follows

$$\vec{\mathbf{H}}\left(\mathbf{U}^-, \mathbf{U}^+\right) = \frac{1}{2}\left[\vec{\mathbf{F}}\left(\mathbf{U}^-\right) + \vec{\mathbf{F}}\left(\mathbf{U}^+\right) - \left|\mathbf{A}\left(\mathbf{U}^*\right)\right|\left(\mathbf{U}^+ - \mathbf{U}^-\right)\right], \tag{6}$$

where $\mathbf{A}(\mathbf{U}^*) = \partial\mathbf{F}/\partial\mathbf{U}$ is the Jacobian of the inviscid flux and is evaluated at the Roe averaged state $\mathbf{U}^*$ (e.g., see [14]). The absolute value $|\mathbf{A}|$ is obtained by decomposing matrix $\mathbf{A}$ as $\mathbf{A} = \mathbf{T}\boldsymbol{\Lambda}\mathbf{T}^{-1}$. Then $|\mathbf{A}| = \mathbf{T}|\boldsymbol{\Lambda}|\mathbf{T}^{-1}$, where $\mathbf{T}$ is the matrix of the left eigenvectors and $\boldsymbol{\Lambda}$ is the matrix of eigenvalues. It is worth noticing here that the choice of a numerical flux in the problem remains an open question in the study of DG schemes [15–17]. For instance, a simple space-centered numerical flux (e.g., the Lax–Friedrichs numerical flux) could be implemented for the computational problem considered in the paper in order to save the computational resource. At the same time, it is well known [14,15] that space-centered fluxes are more dissipative than upwind fluxes, so that the choice of such a flux may not be acceptable in problems where compressibility is essential (e.g. see [17]). Thus we use a slightly more expensive but generally more accurate Roe numerical flux in our computations.

For accurate evaluation of line and surface integrals in (5) an appropriate Gaussian quadrature rule is used. The number of Gauss points is taken according to the order K of DG approximation, as Gaussian quadratures for line integrals are exact for any polynomial of degree $2K + 1$, and they are exact for any polynomial of degree $2K$ if a surface integral is considered. For instance, three Gaussian points should be taken to compute a line integral for a piecewise linear polynomial $K = 1$.

A space discretization of Eq. (3) should be augmented by discretization of boundary conditions in a problem. In our work we consider wave propagation at times $t < T_c$, where T_c is the time when a wave approaches a boundary of the domain D. Thus

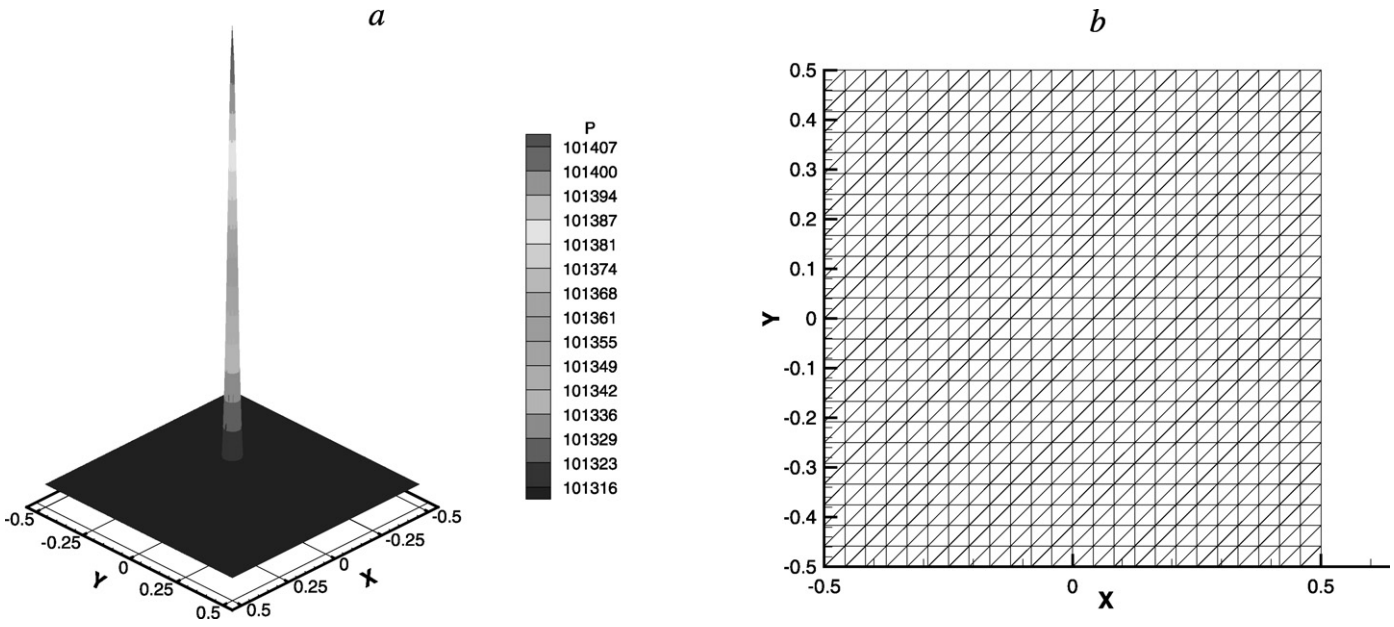

Fig. 1. (a) The initial pressure distribution (11). (b) A uniform grid for computational problem (1), (10).

boundary conditions are $\rho u = 0$, $\rho v = 0$ at any boundary of the domain. The condition $t < T_c$ allows us to avoid the impact of boundary conditions on the accuracy of a numerical solution and to focus our attention entirely on the properties of a DG discretization.

The semi-discrete system obtained as a result of space discretization by a DG method is as follows

$$\mathbf{M}\frac{\mathbf{U}^{n+1}-\mathbf{U}^{n}}{\Delta t}+\mathbf{R}\left(\mathbf{U}^{n}\right)=0, \tag{7}$$

where $\mathbf{R}(\mathbf{U})$ is the residual of a DG discretization, $\mathbf{M}$ is the mass matrix. The above system is integrated in time by the explicit Runge–Kutta (R–K) method. The general explicit R–K method is

$$\mathbf{U}^{(i)}=\mathbf{U}^{(0)}+\Delta t\sum_{j=0}^{i-1}\alpha_{ij}\mathcal{L}\left(\mathbf{U}^{j}\right),\quad i=1,2,\ldots,N,$$
$$\mathbf{U}^{(0)}=\mathbf{U}^{n},\qquad \mathbf{U}^{(N)}=\mathbf{U}^{n+1}, \tag{8}$$

where the spatial operator $\mathcal{L}$ determined by DG approximation is

$$\mathcal{L}=\mathbf{M}^{-1}\mathbf{R}, \tag{9}$$

and the coefficients α_{ij} depend on the order of the scheme. It was discussed in [2] that a spatial DG discretization of order p requires the R–K method of order $p+1$ to provide accurate time integration. In our work we use the R–K method of order 5, as the maximum order of a spatial DG discretization considered in the work is $K=4$.

3. Numerical results for the acoustic pulse problem

One difficulty arising in consideration of CAA problems is that a closed-form solution is not readily available for the test cases where the Euler equations should be solved to compute propagation of acoustic waves. Thus alternative approaches are often suggested to evaluate the convergence rate of the numerical scheme. For instance, in work [5] a DG solution has been computed at small times only and the initial profile has been referred to as an accurate solution to investigate the order of the scheme. Alternatively, the accuracy of a DG scheme has been validated by computing a numerical solution in a bigger domain and using it as a reference solution to avoid the impact of boundary conditions [8]. Meanwhile the nature of aeroacoustic problems requires painstaking attention to the features of a discretization scheme used in computations and the above-mentioned approaches are not always applicable for the scheme validation, as they may result in a wrong conclusion about the scheme accuracy and efficiency. In our work we choose a simple test case where the exact solution is available in order to carry out comprehensive study of a higher order DG scheme for an acoustic wave propagation problem.

Consider the following initial conditions to the system (1) in the domain $D=[-0.5,0.5]\times[-0.5,0.5]$:

$$u(x,y,0)=v(x,y,0)=0,\qquad p(x,y,0)=P(x,y),$$
$$\rho(x,y,0)=\rho_0(x,y). \tag{10}$$

The initial acoustic pressure perturbation $P(x,y)$ is

$$P(x,y)=p_\infty\left(1+A2^{-(R/r_0)^2}\right), \tag{11}$$

where a location R is defined as $R^2=x^2+y^2$, and the pressure parameters are taken as $p_\infty=1$, $A=0.001$ and $r_0=0.02$ in our computations. The initial pressure distribution is shown in Fig. 1a. The initial density distribution is the same as in (11) and is given by

$$\rho_0(x,y)=\rho_\infty\left(1+\frac{A}{c^2}2^{-(R/r_0)^2}\right),$$

where the density $\rho_\infty=1$ and c is the speed of sound.

The exact solution to the problem (1), (10) is defined as (e.g., see [18])

$$p(x,y,t)=p_\infty\left(1+Ap_I(R,t)\right), \tag{12}$$

where

$$p_I(R,t)=\int_0^\infty k^{-1}J_0(kR)\cos(ckt)\int_0^\infty \xi\, J_0(\xi)2^{-(\frac{\xi}{kr_0})^2}\,d\xi\,dk. \tag{13}$$

In the integral above $J_0(y)$ is the Bessel function of order zero.

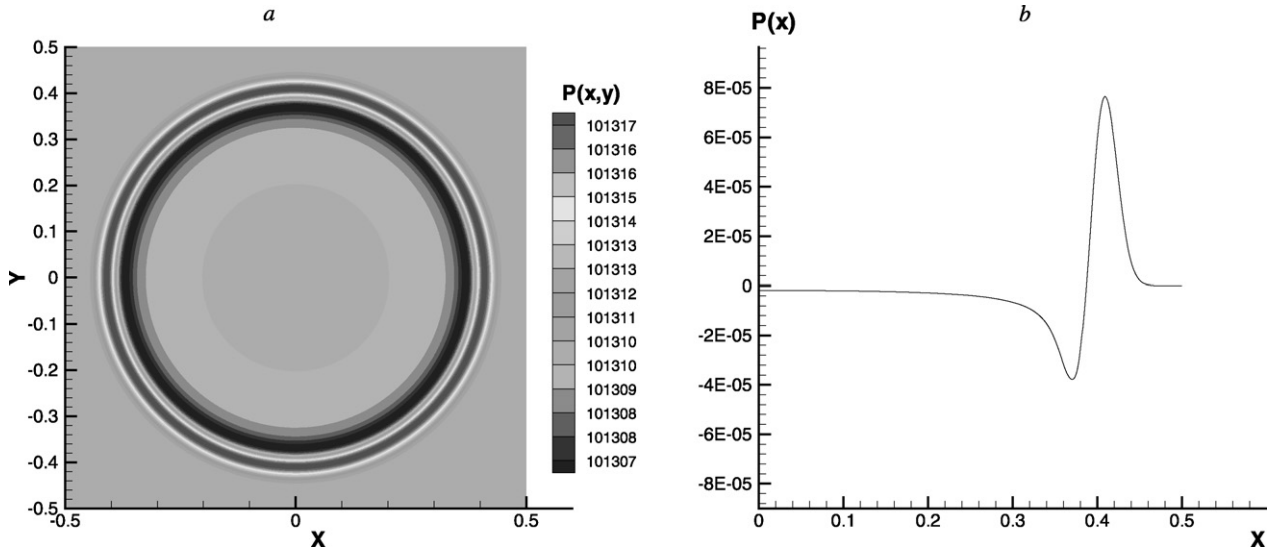

Fig. 2. (a) The exact solution $p(x, y, t)$ over the domain D at time $t = T_2$. (b) The pressure distribution along line $y = 0$ at time $t = T_2$.

We notice that

$$\int_0^\infty \xi J_0(\xi) 2^{-(\frac{\xi}{kr_0})^2} d\xi = \int_0^\infty \xi J_0(\xi) \exp\left(-\alpha \xi^2/k^2\right) d\xi = \frac{k^2}{2\alpha} \exp\left(-k^2/4\alpha\right),$$

where $\alpha = \ln(2)/r_0^2$. Substituting the above result into (13) we obtain the exact solution $p(x, y, t)$ as follows (cf. [10]):

$$p(x, y, t) = p_\infty \left(1 + \frac{A}{2\alpha} \int_0^\infty \exp\left(-k^2/4\alpha\right) \cos(ckt) J_0(kR) k\, dk\right). \tag{14}$$

Let us emphasize here that the function (12) is also a solution to the linearized Euler equations, as for small perturbations ($\delta p(x, y, t) \sim \epsilon p_\infty$, $\epsilon \ll 1$, e.g., see [9]) the solution to the full Euler equations is the same as the solution to the linearized equations. At the same time, in consideration of the above test case we have used a DG discretization of the full Euler equations and have not made any simplifications to the scheme that can be made when one has to solve linearized Euler equations. That has been done deliberately in order to verify the convergence properties of a higher order DG discretization for the full Euler equations in the case that an analytical solution and therefore the solution error is available in the problem. Apparently, a more detailed study of the Euler equations should include test cases where nonlinear effects present. However, at his stage our main goal is to investigate if a higher order DG scheme has advantages over finite-volume schemes in terms of the efficiency of computations.

The integral $p_I(R, t)$ for given location R and time t is computed by the DQDAWO subroutine available in the IMSL Fortran Library [19,20]. The subroutine is designed to integrate a function over an infinite or semi-infinite interval if the integrand contains a sine or a cosine function. The algorithm is to transform the original interval into the finite interval $[0, 1]$ where a uniform grid is generated. The length of each grid subinterval is then taken into account in relation to the size of a period of the cosine integrand to decide either a modified Gauss–Kronrod rule [21] or a Clenshaw–Curtis quadrature rule [22] will be employed to approximate the integral on each subinterval and to evaluate the error. Any subinterval with an unacceptable error estimate is bisected and the integration over new subintervals continues until the required accuracy is achieved.

Most of our computational tests require generation of a uniform grid in the domain D. A quasi-structured grid shown in Fig. 1b is obtained from a Cartesian grid by cutting each grid cell by a diagonal. The Cartesian grid is generated with a regular distribution of grid nodes given by $x_{ij} = ih_x$, $y_{ij} = jh_y$, where $i = 0, \ldots, N_x$, $j = 1, \ldots, N_y$, and h_x and h_y are grid step sizes in the x-direction and y-direction, respectively. In case that a sequence of nested grids is required for computations, it will be generated by doubling the number of grid nodes in the both directions for every Cartesian grid in the sequence.

In our computations the exact solution has been reconstructed at given time T on a very fine uniform grid G_f of $N_f = 22\,201$ nodes. The location R_n has been defined as $R_n^2 = x_n^2 + y_n^2$, where (x_n, y_n) are coordinates of node n, $n = 1, \ldots, N_f$. Computation of the exact solution on a very fine mesh is necessary to guarantee that the exact solution is always available at several points inside every grid cell on coarser meshes where the numerical solution is computed in our test cases. Let the exact solution be known at points $P_1, P_2, \ldots, P_l$ inside the grid cell e_l. The numerical solution is reconstructed at every point P_i, $i = 1, \ldots, l$ using the expansion (2). The reconstruction is performed at every grid cell, so that the numerical solution becomes available at N_f points over the grid. We then compute the solution error at each point P_j, $j = 1, \ldots, N_f$, as

$$e(P_j) = \left|p(x_j, y_j) - p_h(x_j, y_j)\right|, \tag{15}$$

where $p(x_j, y_j)$ and $p_h(x_j, y_j)$ are the exact and approximate solutions at point P_j, respectively.

For our test cases we need the exact solution at a time T when the perturbation front has yet not reached a boundary of the computation domain to avoid the impact of boundary conditions on the accuracy of the results. The time T is evaluated as $T = R_f/c$, where $R_f < 0.5$ is a chosen location of the perturbation front. Thus we consider times T_1 and T_2 that correspond to the radius $R_{f_1} = 0.2$ and $R_{f_2} = 0.4$, respectively. The exact solution (12) over the domain D is shown in Fig. 2a at time T_2. The pressure along line $y = 0$ at time T_2 is displayed in Fig. 2b.

Once the exact solution has been computed, various numerical tests can be carried out to validate the accuracy of a DG scheme. We first compute a numerical solution on a uniform grid G with $N_x = N_y = 50$ at time T_2. A graph of the exact solution and a numerical solution along the line $y = 0$ at time T_2 is presented in

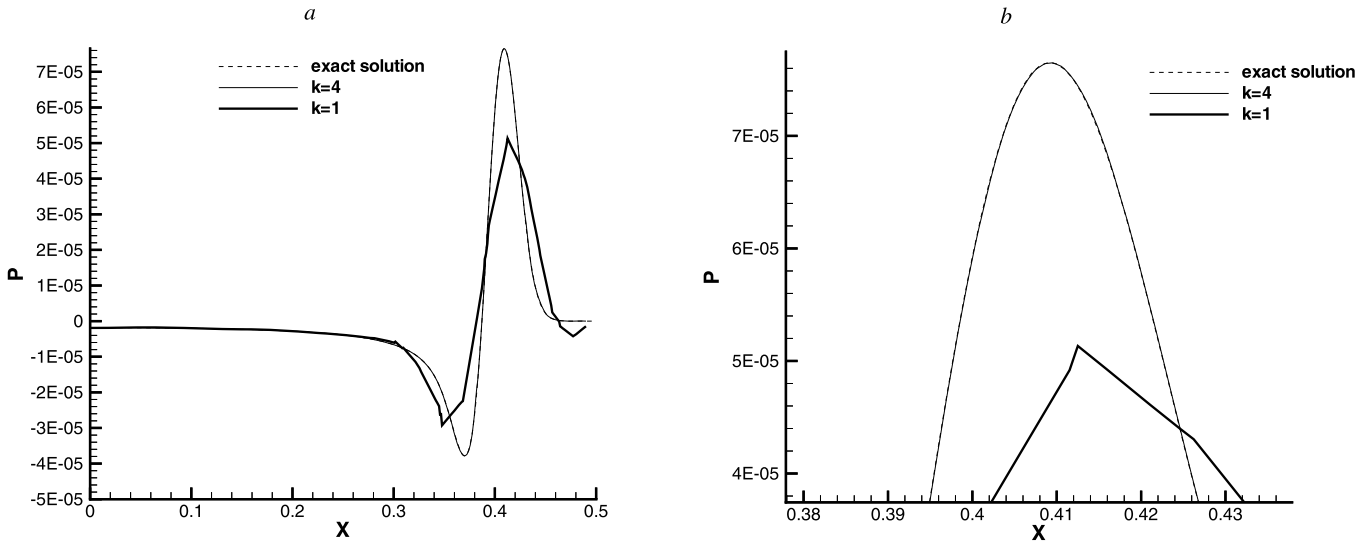

Fig. 3. (a) The solution $p(x, y)$ to the problem (1), (10) at time $t = T_2$. The exact solution is shown as a dashed line. A numerical solution is obtained by a DG method for polynomial degree $K = 1$ (bold line) and $K = 4$. (b) A close-up of the solution in (a) near the pressure peak. A higher order DG solution is visually the same as the exact solution (12).

Table 1
The number of degrees of freedom (N_{DOF}) per grid cell required for a higher order DG discretization. N is the number of grid nodes in each direction on a structured grid that provides the equivalent number of the degrees of freedom.

K	1	2	3	4
N_{DOF}	3	6	10	15
N	112	79	61	50

Fig. 3 where the numerical solution $p(x, y)$ is shown for polynomial degree $K = 1$ and $K = 4$. It can be seen from the figure that a higher order approximation $K = 4$ results in a well resolved solution function, while a piecewise linear approximation $K = 1$ does not resolve a region of a high pressure gradient. This expected result is confirmed by the solution error that is further computed for various polynomial degrees K in order to obtain the convergence history on a sequence of uniform grids.

The evaluation of accuracy of a higher order scheme should be linked to computational efficiency of the scheme. In other words, it should be taken into account that a higher order DG scheme requires computing the higher number of expansion coefficients (degrees of freedom) in (5) in comparison with a lower order scheme. The number of degrees of freedom (N_{DOF}) required to approximate a solution on a single grid cell is shown in Table 1 for polynomial degrees $K = 1, \ldots, 4$. Hence, if we want compare a solution error for different polynomial degrees K in a DG scheme, we need to generate grids with different number of grid cells and to compare the solution error for DG approximations that will have the same number of degrees of freedom on such grids. An example of grids used in our test is given in Table 1. In the table, a sequence of structured grids is shown where $N_x = N_y = N$ for any next grid in the sequence. The grids are generated as to have the total number of degrees of freedom approximately the same for different values of K. For instance, a grid with $N = 112$ will have $N^2 = 12\,544$ nodes and $N^2 \times N_{DOF} = 37\,362$ degrees of freedom for $K = 1$. The DG $K = 1$ approximation on this grid should be compared with a DG solution on a much coarser grid with $N = 50$, if we want to compare the $K = 1$ and $K = 4$ discretizations. The DG $K = 4$ approximation on the latter grid has $N^2 \times N_{DOF} = 37\,500$ degrees of freedom.

The convergence history on a sequence of grids that have the similar number N_{DOF} is shown in Fig. 4. We compute the L_2-norm of the solution error over the grid as

$$\|e\|_{L_2} = \sqrt{\sum_{j=1,\ldots,N_f} e(P_j)^2}, \tag{16}$$

where the error $e(x, y)$ is computed in points P_j determined by the algorithm (15).

The convergence graphs are plotted for polynomial degrees $K = 1, \ldots, 4$ at time $t = T_1$ and $t = T_2$ in Figs. 4a and 4b, respectively. It can be seen from the figure that a higher order discretization is more efficient that a lower order scheme, as the former scheme has a smaller error even if a coarser grid is used for computations.

Let us also mention here that a DG discretization with piecewise constant solution approximation ($K = 0$) is equivalent to a finite volume scheme in terms of the number of degrees of freedom, as one degree of freedom per each variable in the Euler equations is used in a finite volume method on each grid cell. A comparison of a finite volume scheme and a DG scheme has been discussed in [23], where a finite volume scheme has been considered on a uniform grid with $N = 200$. Consequently, a piecewise linear DG discretization ($K = 1$) has been considered on a grid with $N = 100$, and a grid with $N = 64$ has been generated for a piecewise quadratic DG scheme ($K = 2$). It has been shown in [23] that the accuracy of a finite volume scheme was similar to a piecewise linear DG discretization, as expected. However, a DG $K = 2$ scheme has provided a more accurate solution on a coarse grid $N = 64$ in comparison with both a finite volume scheme and a DG $K = 1$ scheme applied on finer meshes. This result is further confirmed by our computations, as the solution error displayed in Fig. 4 for the DG $K = 3$ and $K = 4$ discretizations reveals higher accuracy of these schemes on coarse meshes. At the same time, it has already been discussed in the paper that the implementation of a higher order DG discretization in the problem is a simpler and a more straightforward programming task in comparison with a higher order finite volume discretization, as the DG scheme does not require an expanded stencil to approximate higher order derivatives.

Speaking about the accuracy of a higher order scheme, it is instructive to look at the performance of a DG discretization on non-uniform grids. The propagation of a cylindric acoustic wave (12) does not require generation of an anisotropic grid for the numeri-

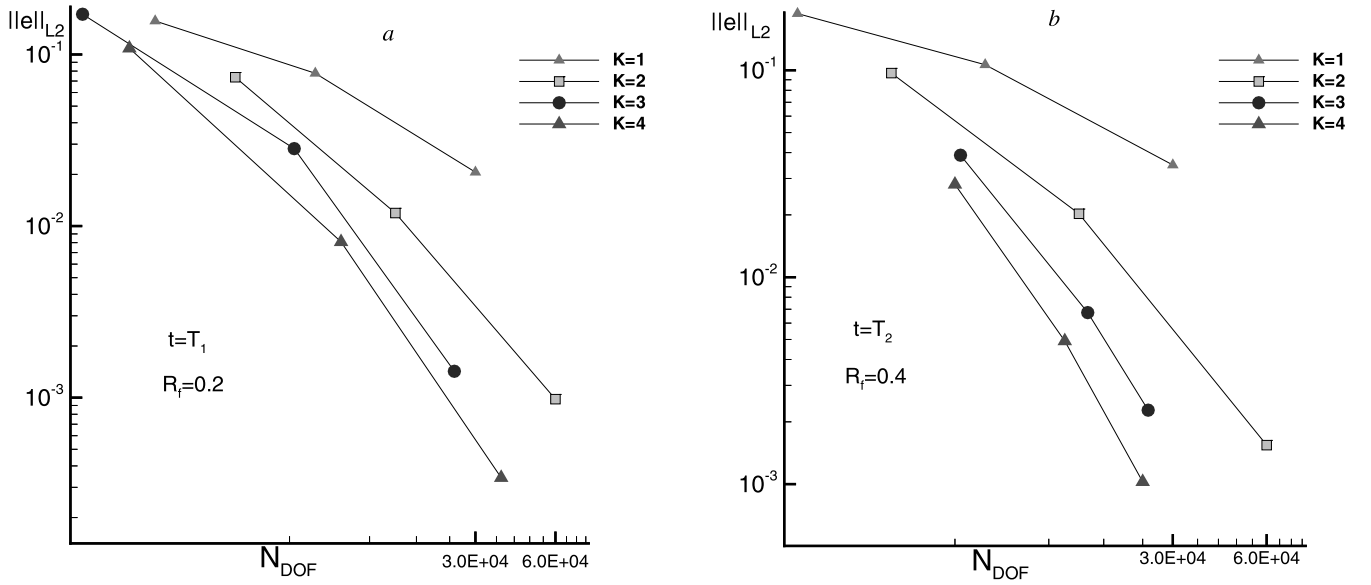

Fig. 4. The convergence history on a sequence of grids that have the similar number of degrees of freedom for various polynomial degrees K. The L_2-norm of the solution error (16) is shown as a function of the number N_{DOF}. (a) The L_2-norm of the solution error at time $t = T_1$. (b) The L_2-norm of the solution error at time $t = T_2$.

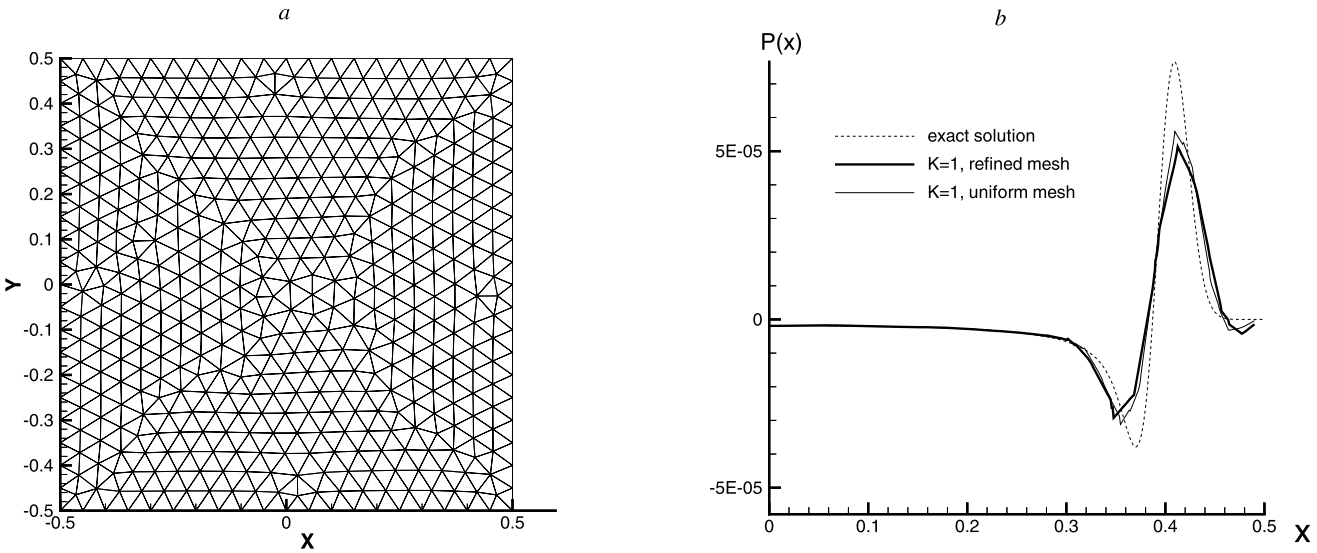

Fig. 5. (a) Example of a slightly perturbed grid generated in domain D. (b) Comparison of a piecewise linear ($K = 1$) DG approximation on a refined grid with that on a uniform grid. The exact solution is shown as a dotted line in the figure.

cal solution of the problem. However, in more complex problems a uniform grid may turn to be unsuitable for adequate resolution of solution features, so that generation of a non-uniform mesh may be required. The theoretical results state that a DG discretization provides an optimal order of convergence on grids with regular geometries. It has been discussed in [2] that the L_2-norm of the solution error can be estimated as $O(h^{K+1})$, where h is a diameter of grid cells. On the other hand, it has been demonstrated in [24, 25] that the convergence estimates that are true on regular meshes fail for a DG discretization on arbitrary unstructured grids. Thus our next test case is to slightly perturb a uniform quasi-structured grid used in our computations and to investigate the accuracy of a DG scheme in order to check how the grid geometry impacts on the accuracy of the DG approximation.

A slightly distorted quasi-structured grid of 228 cells generated in the domain D is shown in Fig. 5a. This grid is then considered as an initial grid for the convergence test. All nested grids that we need to verify the convergence rate are obtained by uniform refinement of the initial grid. The standard uniform refinement procedure is to cut each grid cell into four by connecting the edge midpoints. Such an approach should eventually improve the grid quality as it does not increase the maximum angle in each triangle but at the same time decreases the diameter of each grid cell.

An example of a numerical solution obtained on a non-uniform mesh is shown in Fig. 5b. A piecewise linear DG solution shown in the figure is computed on a uniform quasi-structured mesh of 2706 nodes and also on a refined mesh that has the same number of grid nodes. The latter solution appears to be more dissipative at the region of the pressure peak. The above result is further illustrated by the convergence history shown in Fig. 6 at time $t = T_2$. In the figure, the L_2-norm of a DG solution obtained on a sequence of refined grids is compared with the error on a sequence of uniform quasi-structured grids that has been used in our previous tests. The convergence history for polynomial degrees $K = 1$ and $K = 2$ is shown in Fig. 6a, while Fig. 6b displays the convergence history for polynomial degrees $K = 3$ and $K = 4$. It can be seen from the figure that a DG discretization is sensitive to the grid quality as even slight distortion of a uniform mesh results in the loss of

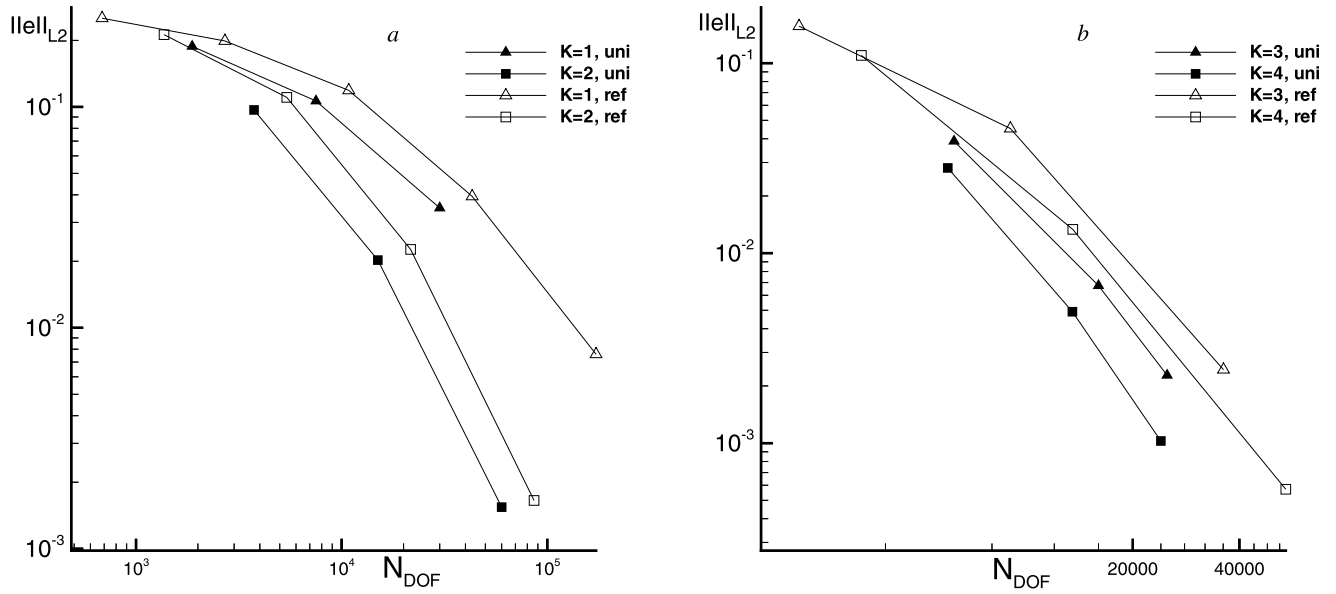

Fig. 6. The comparison of the convergence rate of a DG scheme on uniform and refined grids. The L_2-norm of the solution error is computed as a function of the number of degrees of freedom at time $t = T_2$. (a) The convergence history for a DG $K = 1$ scheme and a DG $K = 2$ scheme on uniform (black symbols) and refined (empty symbols) grids. (b) The convergence history for a DG $K = 3$ scheme and a DG $K = 4$ scheme on uniform (black symbols) and refined (empty symbols) grids.

accuracy of the scheme. The DG solution obtained for a fixed number of the degrees of freedom is always better resolved on uniform quasi-structured meshes as the error on a uniform mesh is always smaller than the error on a refined distorted mesh. Thus the convergence rate of a DG scheme on arbitrary unstructured grids still remains an open question, especially in case that coarse meshes are considered in the problem.

Finally, we are interested in validation of our time marching scheme for a higher order DG discretization. The previous numerical results were obtained with the 5-stage R–K method that does not have a TVD (total variation diminishing) property. We now compare it with a TVD R–K scheme where the total variation of the solution does not increase as the time progresses. The TVD property of a time discretization is important as it maintains the stability of the scheme, and it is well known that a non-TVD R–K time discretization can generate oscillations even for a TVD spatial discretization (e.g., see [26]).

The general TVD R–K method was formulated in [26] as follows

$$\mathbf{U}^{(i)} = \sum_{p=0}^{i-1} \alpha_{ip}\mathbf{U}^{(p)} + \beta_{ip}\Delta t\mathcal{L}\big(\mathbf{U}^{p}\big), \quad i = 1, 2, \ldots, m,$$
$$\mathbf{U}^{(0)} = \mathbf{U}^{n}, \qquad \mathbf{U}^{(m)} = \mathbf{U}^{n+1}, \tag{17}$$

where the operator $\mathcal{L}$ is defined by (9). For non-negative α_{ip} and β_{ip} the expansion (17) represents a combination of Euler forward operators and computations are similar to a non-TVD R–K scheme. The problem, however, is that for a higher-order TVD R–K method the coefficients β_{ip} might be negative, in which case the computation of an adjoint operator $\tilde{\mathcal{L}}$ is required to meet the TVD condition for the scheme (17). The operator $\tilde{\mathcal{L}}$ approximates the same space problem (1) as the operator $\mathcal{L}$ does, but the backward time integration should be applied in the problem

$$\mathbf{U}^{n+1} - \mathbf{U}^{n} = -\Delta t\tilde{\mathcal{L}}\mathbf{U}^{n}.$$

Since the computation of $\tilde{\mathcal{L}}$ comes at the same cost as computation of operator $\mathcal{L}$, the implementation of a higher-order TVD R–K scheme doubles the CPU time required for the discretization as well as storage requirement. Hence, an alternative is to combine a higher order space discretization with lower order TVD R–K schemes where all coefficients β_{in} are non-negative [4,7,8]. At the same time, a lower order R–K scheme may not provide the accurate time integration in the problem. Thus in our work we compare the 4th order $m = 3$ TVD R–K scheme previously combined with a higher-order DG discretization in [7,8] with the 5th order $m = 6$ scheme (17) that will be used along with a DG $K = 4$ discretization.

The coefficients α_{ip} and β_{ip} required for the both TVD R–K schemes that are exploited in our work can be found in paper [26]. Let us notice here again that we need to compute the adjoint operator $\mathcal{L}$ in order to implement the TVD $N = 6$ R–K method. For instance, the solution $\mathbf{U}^{(4)}$ is computed as

$$\mathbf{U}^{(4)} = \frac{1}{4}\mathbf{U}^{(0)} - \frac{5}{64}\Delta t\tilde{\mathcal{L}}\big(\mathbf{U}^{(0)}\big) + \frac{1}{8}\mathbf{U}^{(1)} - \frac{13}{64}\Delta t\tilde{\mathcal{L}}\big(\mathbf{U}^{(1)}\big) + \frac{1}{8}\mathbf{U}^{(2)} + \frac{1}{8}\Delta t\mathcal{L}\big(\mathbf{U}^{(2)}\big) + \frac{1}{2}\mathbf{U}^{(3)} + \frac{9}{16}\Delta t\mathcal{L}\big(\mathbf{U}^{(3)}\big).$$

The results of the implementation of a TVD R–K scheme in the problem are presented in Fig. 7 where the convergence history on a sequence of uniform grids is shown at time $t = T_2$. In the figure the L_2-norm of the solution error is displayed as a function of the number N_c of grid nodes and the solution is integrated in time up to $t = T_2$ by using the 5-stage R–K scheme, the 3-stage TVD R–K scheme and the 6-stage TVD R–K scheme.

The convergence on coarse grids ($N_c < 625$) is shown in Fig. 7a, while the error on finer grids ($N_c < 2500$) is demonstrated in Fig. 7b. The performance of the 3-stage TVD R–K scheme on coarse meshes is identical to the convergence history obtained with a 5-stage R–K method. Meanwhile, the 6-stage TVD R–K method appears to be slightly less accurate in comparison with the 5-stage and the 3-stage R–K method on coarse meshes. However, once the grid has been refined, the accuracy of the 6-stage method is getting better. The implementation of the 6-stage TVD R–K method on a grid of 2500 nodes results in a more accurate solution in comparison with the 5-stage non-TVD and the 3-stage TVD R–K method.

Since every convergence graph appears as a broken line in the figure, it is difficult to evaluate the order of convergence from the slope of a graph. The results of the convergence order evaluation made on two consequently refined grids are displayed in Table 2, where the order q has been computed from the solution error on grids with the number of nodes $N_c = 225, 900$ and 2500. Namely,

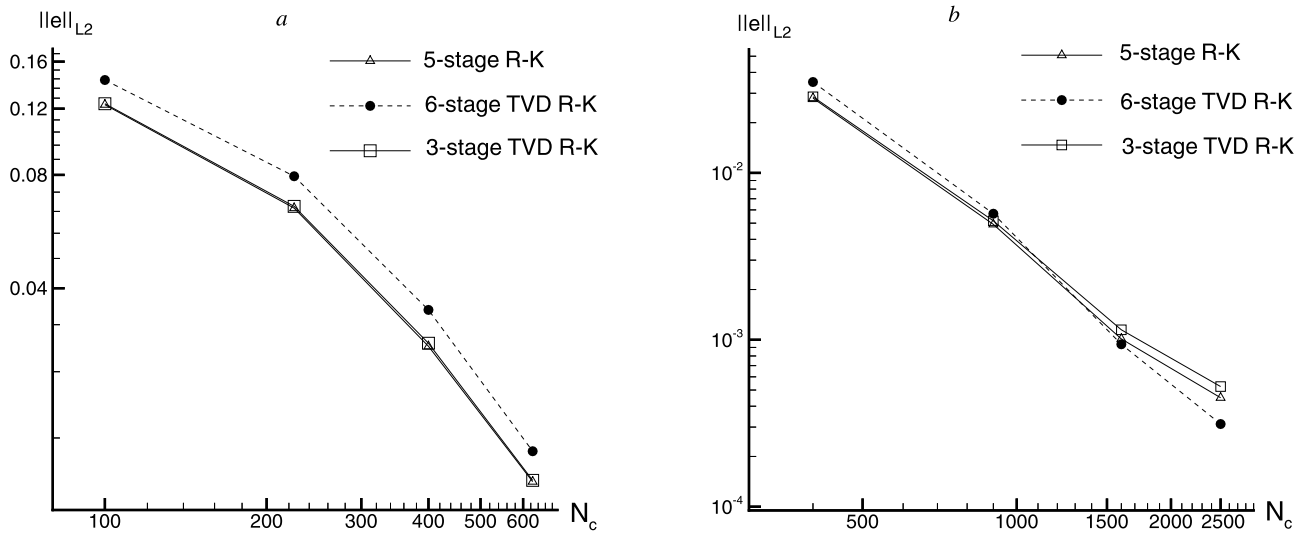

Fig. 7. Implementation of a TVD R–K schemes for a DG $K = 4$ discretization. The L_2-norm of the solution error is shown at time $t = T_2$. (a) A 5-stage R–K scheme, a 3-stage TVD R–K scheme and a 6-stage R–K scheme on coarse meshes. (b) The same time-marching schemes on fine meshes.

Table 2
The convergence order q of a DG discretization for polynomial degree $K = 4$. The solution error is computed at time $t = T_2$, where the solution is integrated in time by the 5-stage R–K scheme (RK_5), the 3-stage TVD R–K scheme (RK_3^{tvd}) and the 6-stage TVD R–K scheme (RK_6^{tvd}).

N_c	225	900	2500
q (RK_5)	1.55	4.74	3.51
q (RK_3^{tvd})	1.56	5.00	3.63
q (RK_6^{tvd})	1.45	5.24	4.93

let us introduce the error ratio r_e as $r_e = \|e\|_{i-1}/\|e\|_i$, where $\|e\|_i$ is the L_2-norm (16) computed over the ith grid in the sequence of grids used in our computations. The convergence order is then evaluated as

$$q = \frac{\log(r_e)}{\log(r_N)},$$

where $r_N = \sqrt{N_{DOF_i}/N_{DOF_{i-1}}}$, and N_{DOF_i} is the number of degrees of freedom on the ith grid.

It can be seen from the table that using a more accurate TVD R–K method improves the convergence rate as the grid is refined. However, further conclusions about the accuracy and efficiency of a TVD R–K method that is consistent with the order of a spatial DG discretization should be based on integration over a longer time interval where we need to take into account the impact of boundary conditions on the accuracy of the scheme. That should consist a topic of future work.

4. Concluding remarks

In the present work we have considered the problem of acoustic wave propagation where a numerical solution to the problem has been obtained by a higher order discontinuous Galerkin method. The aim of our work was to validate a higher order DG discretization for a simple problem where the analytical solution was available. The convergence tests were made to compare the accuracy of piecewise polynomial approximation with the degree at most $K = 4$ with that for a lower order discretization. The convergence tests demonstrated that a higher order scheme has better accuracy than a lower order scheme with the same number of degrees of freedom. Hence, a higher order discretization makes it possible to compute an accurate solution to the Euler equations on a coarse mesh, which is an issue of utmost importance in CAA applications where very fine grids are usually considered in large computational domains in order to provide accurate calculations in the near and the far acoustic field.

The results of our paper demonstrate that a higher order DG discretization can be considered as a reliable alternative to finite volume schemes currently used for numerical solution of the Euler equations on unstructured grids. At the same time, regarding the scheme efficiency, the implementation of a DG method should be further investigated to reduce the computational cost of the algorithm. Besides the scheme parallelization one of the future work targets is to understand whether a technique similar to the quadrature-free method developed in [27] for solving the linearized Euler equations can be applied to the computation of nonlinear fluxes in the problem.

References

[1] C.K.W. Tam, Computational aeroacoustics: Issues and methods, AIAA J. 33 (10) (1995) 1788–1796.
[2] B. Cockburn, Discontinuous Galerkin methods for convection-dominated problems, in: T. Barth, H. Deconinck (Eds.), High-Order Discretization Methods in Computational Physics, in: Lecture Notes in Comput. Sci. Engrg., vol. 9, Springer-Verlag, Heidelberg, 1999, pp. 69–224.
[3] N. Chevaugeon, J.-F. Remacle, X. Gallez, P. Ploumhans, S. Caro, Efficient discontinuous Galerkin methods for solving acoustic problems, AIAA 2005-2823, 2005.
[4] P. Delorme, P. Mazet, C. Peyret, Y. Ventribout, Computational aeroacoustics applications based on a discontinuous Galerkin method, C. R. Mecanique 333 (2005) 676–682.
[5] M. Dumbser, C.-D. Munz, ADER discontinuous Galerkin schemes for aeroacoustics, C. R. Mecanique 333 (2005) 683–687.
[6] R. Leneveu, B. Schiltz, J. Manera, S. Caro, Parallel DGM scheme for LEE applied to exhaust and bypass problems, AIAA 2007-3510, 2007.
[7] I. Toulopoulos, J. Ekaterinaris, High-order discontinuous-Galerkin discretizations for computational aeroacoustics in complex domains, AIAA-2004-522, 2004.
[8] A. Richter, J. Stiller, R. Grundmann, Stabilized discontinuous Galerkin methods for flow–sound interaction, J. Comput. Acoust. 15 (1) (2007) 123–143.
[9] C.K.W. Tam, J.C. Hardin (Eds.), Proceedings of the Second Computational Aeroacoustics (CAA) Workshop on Benchmark Problems, NASA Langley Research Center, 1997 (NASA Conference Publication 3352).
[10] C.K.W. Tam, J.C. Webb, Dispersion-relation-preserving schemes for computational acoustics, J. Comput. Phys. 107 (2) (1993) 262–281.
[11] F.Q. Hu, M.Y. Hussaini, J. Manthey, Low-dissipation and -dispersion Runge–Kutta schemes for computational acoustics, NASA Technical Report, 1994.
[12] D.J. Mavriplis, Unstructured mesh discretizations and solvers for computational aerodynamics, AIAA 2007-3955, 2007.
[13] N.B. Petrovskaya, Discontinuous weighted least-squares approximation on irregular grids, CMES: Computer Modelling in Engineering & Sciences 32 (2) (2008) 69–84.

[14] Ch. Hirsch, Numerical Computation of Internal and External Flows, John Wiley & Sons, 1988.
[15] R.J. LeVeque, Numerical Methods for Conservation Laws, Birkhäuser Verlag, Basel, Switzerland, 1992.
[16] N.B. Petrovskaya, Two types of solution overshoots in discontinuous Galerkin discretization schemes, Commun. Math. Sci. 3 (2) (2005) 233–247.
[17] J.J.W. van der Vegt, H. van der Ven, Space–time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flow, in: 33rd Computational Fluid Dynamics Course "Novel Methods for Solving Convection Dominated Systems", the von Karman Institute, Rhode-St-Genese, Belgium, March 24–28, 2003.
[18] L.D. Landau, E.M. Lifshits, Course of Theoretical Physics VI: Hydrodynamics, Nauka Publishing Inc., Moscow, 1988 (in Russian).
[19] The IMSL FORTRAN Library, http://www.vni.com/products/imsl/fortran/overview.php.
[20] R. Piessens, E. deDoncker-Kapenga, C. Uberhuber, D. Kahaner, QUADPACK: A Subroutine Package for Automatic Integration, Springer-Verlag, Berlin, 1983.
[21] D.P. Laurie, Calculation of Gauss–Kronrod quadrature rules, Math. Comp. 66 (1997) 1133–1145.
[22] C.W. Clenshaw, A.R. Curtis, A method for numerical integration on an automatic computer, Numer. Math. 2 (1960) 197–205.
[23] A. Wolkov, Ch. Hirsch, B. Leonard, Discontinuous Galerkin method on unstructured hexahedral grids for 3D Euler and Navier–Stokes equations, AIAA 2007-4078, 2007.
[24] F.Q. Hu, M.Y. Hussaini, P. Rasetarinera, An analysis of the discontinuous Galerkin method for wave propagation problems, J. Comput. Phys. 151 (2) (1999) 921–946.
[25] N.B. Petrovskaya, A.V. Wolkov, S.V. Lyapunov, Modification of basis functions in high order discontinuous Galerkin schemes for advection equation, Appl. Math. Model. 32 (5) (2008) 826–835.
[26] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, J. Comput. Phys. 77 (1988) 439–471.
[27] H.L. Atkins, C.W. Shu, Quadrature-free implementation of discontinuous Galerkin method for hyperbolic equations, AIAA 96-1683, 1996.