A HIGH-ORDER DISCONTINUOUS GALERKIN SOLVER FOR 3D AERODYNAMIC TURBULENT FLOWS

F. Bassi⋆, A. Crivellini⋆, D. A. Di Pietro†, S. Rebay‡

⋆ Dipartimento di Ingegneria Industriale - Università di Bergamo
Viale Marconi 5, 24044 Dalmine (BG), Italy
e-mail: francesco.bassi@unibg.it, andrea.crivellini@unibg.it

† ENPC–CERMICS,
6–8 av. Blaise Pascal, Cité Descartes–Champs sur Marne, 77455 Marne La Vallée, France
e-mail: dipietro@cermics.enpc.fr

‡ Dipartimento di Ingegneria Meccanica - Università di Brescia
via Branze 38, 25123 Brescia, Italy
e-mail: stefano.rebay@bsing.unibs.it

Key words: Discontinuous Galerkin, Implicit Methods, Turbulent flows, Stanitz elbow, ONERA M6 wing

Abstract. This paper deals with the evaluation and validation of a recently developed parallel discontinuous Galerkin code for the numerical solution of the RANS and $k$-$\omega$ turbulence model equations. The main features of the code can be summarized as follows: a) high-order spatial accuracy on hybrid grids, b) fully coupled, implicit time discretization, c) non-standard, realizable $k$-$\omega$ model implementation, d) efficient parallel execution using METIS package for grid partitioning and PETSc library for the linear algebra. The paper reports the numerical results of second- and third-order accurate computations of the subsonic turbulent flow in the Stanitz elbow and of the transonic turbulent flow around the ONERA M6 wing.

1 INTRODUCTION

The growing interest that the discontinuous Galerkin (DG) method has been receiving in the recent past is due to various attractive features of the method. DG methods are in fact finite element methods which account for the physics of wave propagation by means of Riemann solvers as in upwind finite volume methods but, unlike the latter, can achieve higher-order accuracy on general unstructured grids using high degree polynomials as is customary in the classical (continuous) finite element method. Moreover, the implementation of high-order accurate boundary conditions is straightforward. In addition, DG methods lead to very compact space discretization formulae for both the Euler and the
Navier-Stokes equations. The compactness of the scheme is particularly advantageous when an implicit time advancement scheme is employed and/or for a parallel implementation of the method.

In the process of evaluating and validating a Reynolds-Averaged Navier-Stokes (RANS) DG solver developed in the latest years, this paper presents 3D high-order computational results for two complex turbulent flows, namely the subsonic flow in the Stanitz elbow and the transonic flow around the ONERA M6 wing.

For turbulent flows, at present, the code solves the RANS equations closed by the high- or low-Reynolds number $k$-$\omega$ turbulence model of Wilcox. As reported in the stability of the coupled RANS and $k$-$\omega$ computations takes advantage of a non-standard implementation of the $k$-$\omega$ model, whereby $\tilde{\omega} = \log \omega$ rather than $\omega$ itself is used as unknown. Moreover, robustness and efficiency of the code have been further improved by satisfying realizability constraints and Schwarz inequality on turbulent stresses as described.

The capabilities of the DG code have been recently extended in several directions. First of all, it can now handle arbitrary hybrid three dimensional grids containing hexahedra, pyramids, prisms and tetrahedra. Being based on a modal polynomial expansion in the physical space, the DG approximation of the code can use, for a given degree of polynomial approximation, the same number of degrees of freedom in each element, irrespective of its geometrical shape. The viscous flux discretization scheme, which is the straightforward extension to three dimensions of the scheme proposed in the code, can also be used on hybrid grids without any modification. Secondly, the code has been fully parallelized to run on any parallel system supporting the MPI standard for all message-passing communication, by using the Portable Extensible Toolkit for Scientific Computation (PETSc) software.

As regards time integration, the DG code features first- and second-order fully implicit time integration schemes, introduced in for the Euler and Navier-Stokes equations and in for the coupled RANS and $k$-$\omega$ equations. The code relies on the PETSc library for the parallel solution of linear systems resulting from the implicit time discretization and can thus access several direct and iterative linear system solvers available in PETSc.
2 GOVERNING EQUATIONS

The complete set of RANS and $k$-$\omega$ equations can be written as:

\[
\begin{align*}
\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j} (\rho u_j) &= 0, \\
\frac{\partial}{\partial t} (\rho u_i) + \frac{\partial}{\partial x_j} (\rho u_j u_i) &= -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ji}}{\partial x_j}, \\
\frac{\partial}{\partial t} (\rho e_0) + \frac{\partial}{\partial x_j} (\rho u_j h_0) &= \frac{\partial}{\partial x_j} \left[ u_i \tau_{ij} - q_j \right] - \tau_{ij} \frac{\partial u_i}{\partial x_j} + \beta^* \rho \bar{e} \bar{\omega}, \\
\frac{\partial}{\partial t} (\rho k) + \frac{\partial}{\partial x_j} (\rho u_j k) &= \frac{\partial}{\partial x_j} \left[ (\mu + \sigma^* \mu_t) \frac{\partial k}{\partial x_j} \right] + \tau_{ij} \frac{\partial u_i}{\partial x_j} - \beta^* \rho \bar{e} \bar{\omega}, \\
\frac{\partial}{\partial t} (\rho \bar{\omega}) + \frac{\partial}{\partial x_j} (\rho u_j \bar{\omega}) &= \frac{\partial}{\partial x_j} \left[ (\mu + \sigma^* \mu_t) \frac{\partial \bar{\omega}}{\partial x_j} \right] + \frac{\alpha}{k} \tau_{ij} \frac{\partial u_i}{\partial x_j} - \beta^* \rho \bar{e} \bar{\omega} + (\mu + \sigma^* \mu_t) \frac{\partial \bar{\omega}}{\partial x_k} \frac{\partial \bar{\omega}}{\partial x_k},
\end{align*}
\]

where the pressure, the turbulent and total stress tensors, the heat flux vector and the eddy viscosity are given by:

\[
\begin{align*}
p &= (\gamma - 1) \rho \left( e_0 - \frac{1}{2} u_k u_k \right), \\
\tau_{ij} &= 2\mu_t \left[ S_{ij} - \frac{1}{3} \frac{\partial u_k}{\partial x_k} \delta_{ij} \right] - \frac{2}{3} \rho \bar{k} \delta_{ij}, \\
\bar{\tau}_{ij} &= 2\mu \left[ S_{ij} - \frac{1}{3} \frac{\partial u_k}{\partial x_k} \delta_{ij} \right] + \tau_{ij}, \\
q_j &= -\left( \frac{\mu}{Pr} + \frac{\mu_t}{Pr_t} \right) \frac{\partial h}{\partial x_j}, \\
\bar{\mu}_t &= \alpha^* \rho \bar{e} \bar{\omega}, \quad \bar{k} = \max(0, k).
\end{align*}
\]

Here $\gamma$ is the ratio of gas specific heats, $Pr$ and $Pr_t$ are the molecular and turbulent Prandtl numbers and

\[
S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)
\]

is the mean strain-rate tensor. The closure parameters $\alpha$, $\alpha^*$, $\beta$, $\beta^*$, $\sigma$, $\sigma^*$ are those of the high- or low-Reynolds number $k$-$\omega$ model of Wilcox\(^1\).

Notice that the RANS and $k$-$\omega$ equations above are not in standard form since the variable $\bar{\omega} = \log \omega$ appears instead of $\omega$ in Eqs. (3), (4), (5). The use of the logarithm of turbulence variables has been introduced by Ilinca and Pelletier\(^7\) in the context of the $k$-$\epsilon$ model using wall functions and it is attractive because the positivity of the turbulence variables is guaranteed and because the distribution of the logarithmic turbulence variables is much more smooth than that of the turbulence variables themselves. However, applying the idea of logarithmic variables to the $k$-$\omega$ model, which is integrated down to
the wall without using wall functions, we have found useful to introduce the logarithm of $\omega$ but not the logarithm of $k$. In fact, the near wall distribution of $\log \omega$ is much better behaved than that of $\omega$, whilst the solid wall boundary condition for $k$ ($k = 0$) leads to an infinite value of $\log k$.

To deal with possible negative values of $k$ we have used the limited value $\overline{k}$ and the related eddy viscosity $\overline{\mu}_t$, given by Eq. (10), in the source terms of Eqs. (3), (4), (5). Notice that $\overline{k}$ has been limited exactly to zero (and not to an arbitrary small value) because, after the appropriate substitutions, no term in the above equations is divided by $\overline{k}$. Notice that the solution of Eq. (4) is not limited and that, as a consequence, $k$ could also take negative values. In practice we have noticed that negative values of $k$ may occur during the time evolution of the solution or may even be present in a converged steady state solution. However the occurrence of negative $k$ values can be eliminated by refining the computational grid and/or by increasing the degree of the polynomial approximation. As a final comment on the governing equations, notice that a source term is present in the mean-flow energy equation because the total energy $e_0$ and the total enthalpy $h_0$ do not include the turbulent kinetic energy, see e.g.\textsuperscript{8}.

Besides the limited value $\overline{k}$, we also notice that in the source terms of Eqs. (3), (4), (5) and in the eddy viscosity defined by Eq. (10) the variable $\overline{\omega}_r$ is employed instead of $\overline{\omega}$. This is to indicate that $\overline{\omega}_r$ fulfills suitably defined “realizability” conditions which in practice put a lower limit on $\overline{\omega}$ in Eqs. (3), (4), (5) and (10). This limitation substantially improves the stability and robustness of turbulent flow computations because there is numerical evidence that too small, though positive, values of $\omega = e^{\overline{\omega}}$ can lead to sudden breakdown of the computations. Very small values of $\omega$ could easily appear when approaching a steady state solution through large unphysical variations of the solution which are typical of implicit time integration methods. In nearly all cases the breakdown of computations could be avoided by reducing the time step size but, obviously, this is not a satisfactory way to address this issue since, doing so, implicit methods could not exploit their superior stability characteristics and would be very inefficient.

A better alternative is to consider realizability conditions which guarantee that the turbulence model predicts positive normal turbulent stresses and satisfies the Schwarz inequality for shear turbulent stresses. In the case of the $k$-$\omega$ model both requirements can be satisfied by imposing a lower bound on $\overline{\omega}$. In fact, the realizability conditions

$$\overline{\rho u_i u_i} \geq 0, \quad (11)$$

$$\left(\overline{\rho u_i u_j}ight)^2 \leq \overline{\rho u_i u_i} \overline{\rho u_j u_j} \quad (12)$$

imply, in terms of the modeled turbulent stresses,

$$\frac{2}{3} \rho \overline{k} - 2 \overline{\mu}_t \left( S_{ii} - \frac{1}{3} \frac{\partial u_k}{\partial x_k} \right) \geq 0, \quad i = 1, 2, 3, \quad (13)$$
\[-2\bar{\mu}_t S_{ij} \leq \left[ \frac{2}{3} \rho \bar{k} - 2\bar{\mu}_t \left( S_{ii} - \frac{1}{3} \frac{\partial u_k}{\partial x_k} \right) \right] \left[ \frac{2}{3} \rho \bar{k} - 2\bar{\mu}_t \left( S_{jj} - \frac{1}{3} \frac{\partial u_k}{\partial x_k} \right) \right], \]

\[i, j = 1, 2, 3, \quad i \neq j. \quad (14)\]

From Eqs. (13) and (14), recalling Eq. (10), we obtain

\[\frac{e^{\tilde{\omega}}}{\alpha^*} - 3 \left( S_{ii} - \frac{1}{3} \frac{\partial u_k}{\partial x_k} \right) \geq 0, \quad i = 1, 2, 3, \quad (15)\]

\[\left( \frac{e^{\tilde{\omega}}}{\alpha^*} \right)^2 - 3 \left( S_{ii} + S_{jj} - \frac{1}{3} \frac{\partial u_k}{\partial x_k} \right) \frac{e^{\tilde{\omega}}}{\alpha^*} + 9 \left[ \left( S_{ii} - \frac{1}{3} \frac{\partial u_k}{\partial x_k} \right) \left( S_{jj} - \frac{1}{3} \frac{\partial u_k}{\partial x_k} \right) - S_{ij}^2 \right] \geq 0, \]

\[i, j = 1, 2, 3, \quad i \neq j. \quad (16)\]

Let’s denote with \(a\) the maximum value of the unknown \(e^{\tilde{\omega}}/\alpha^*\) corresponding to the zeros of Eqs. (15) and (16). Then the lower bound \(\tilde{\omega}_{r0}\) that guarantees realizable turbulent stresses is given by

\[\frac{e^{\tilde{\omega}_{r0}}}{\alpha^*} = a. \quad (17)\]

The solution of Eq. (17) is trivial for the high-Reynolds number \(k-\omega\) model because in this case \(\alpha^*\) is constant. For the low-Reynolds number \(k-\omega\) model \(\alpha^*\) depends on the turbulent Reynolds number according to the equation

\[\alpha^* = \alpha_t^* \frac{\alpha_0^* + \text{Re}_t/R_k}{1 + \text{Re}_t/R_k^*}, \quad (18)\]

where \(\alpha_t^*, \alpha_0^*\) and \(R_k\) are constants and \(\text{Re}_t\) is the turbulent Reynolds number given by \(\text{Re}_t = k/(e^{\tilde{\omega}} \nu)\). Combining Eqs. (17) and (18) we find \(\tilde{\omega}_{r0}\) from the following second degree equation for the unknown \(e^{\tilde{\omega}_{r0}}\)

\[e^{2\tilde{\omega}_{r0}} - \left( \alpha_t^* \alpha_0^* a - \frac{k}{R_k \nu} \right) e^{\tilde{\omega}_{r0}} - \alpha_t^* \frac{k}{R_k \nu} a = 0, \quad (19)\]

and, finally, we set \(\tilde{\omega}_r\) in Eqs. (3), (4), (5) and (10) as

\[\tilde{\omega}_r = \max \left( \tilde{\omega}, \tilde{\omega}_{r0} \right). \quad (20)\]
3 DG SPACE DISCRETIZATION

The complete set of RANS and \( k-\omega \) turbulence model equations can be written in compact form as

\[
\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{F}_c(\mathbf{u}) + \nabla \cdot \mathbf{F}_v(\mathbf{u}, \nabla \mathbf{u}) + \mathbf{s}(\mathbf{u}, \nabla \mathbf{u}) = 0, \tag{21}
\]

where \( \mathbf{u}, \mathbf{s} \in \mathbb{R}^M \) denote the vectors of the \( M \) conservative variables and source terms, \( \mathbf{F}_c, \mathbf{F}_v \in \mathbb{R}^M \otimes \mathbb{R}^N \) denote the inviscid and viscous flux functions, respectively, and \( N \) is the space dimension. The entries of \( \mathbf{u}, \mathbf{s}, \mathbf{F}_c \) and \( \mathbf{F}_v \) can be found by comparison with Eqs. (19).

Multiplying Eq. (21) by an arbitrary test function \( \phi \), integrating over the domain \( \Omega \) and integrating by parts the divergence terms, the weak form of Eq. (21) reads:

\[
\int_{\Omega} \phi \frac{\partial \mathbf{u}}{\partial t} \, dx - \int_{\Omega} \nabla \phi \cdot \mathbf{F}(\mathbf{u}, \nabla \mathbf{u}) \, dx + \int_{\partial \Omega} \phi \mathbf{F}(\mathbf{u}, \nabla \mathbf{u}) \cdot \mathbf{n} \, d\sigma + \int_{\Omega} \phi \mathbf{s}(\mathbf{u}, \nabla \mathbf{u}) \, dx = 0, \tag{22}
\]

where \( \mathbf{F} \) is the sum of the inviscid and viscous fluxes.

In order to construct a DG discretization of Eq. (22), we consider a triangulation \( T_h = \{ K \} \) of an approximation \( \Omega_h \) of \( \Omega \), that is we partition \( \Omega_h \) into a set of non-overlapping elements \( K \) (not necessarily simplices). We denote with \( \mathcal{E}_h^0 \) the set of internal element faces, with \( \mathcal{E}_h^\partial \) the set of boundary element faces and with \( \mathcal{E}_h = \mathcal{E}_h^0 \cup \mathcal{E}_h^\partial \) their union. We moreover set

\[
\Gamma_h = \bigcup_{e \in \mathcal{E}_h^0} e, \quad \Gamma_h^\partial = \bigcup_{e \in \mathcal{E}_h^\partial} e, \quad \Gamma_h = \Gamma_h^0 \cup \Gamma_h^\partial. \tag{23}
\]

The solution is approximated on \( T_h \) as a piecewise polynomial function possibly discontinuous on element interfaces, i.e. we assume the following space setting for each component \( u_{h,i} = u_{h,1}, \ldots, u_{h,M} \) of the numerical solution \( \mathbf{u}_h \):

\[
u_{h,i} \in \Phi_h \overset{\text{def}}{=} \left\{ \phi_h \in L^2(\Omega) : \phi_h|_K \in \mathbb{P}_k(K) \ \forall K \in T_h \right\} \tag{24}
\]

for some polynomial degree \( k \geq 0 \), being \( \mathbb{P}_k(K) \) the space of polynomials of global degree at most \( k \) on the element \( K \). Following Brezzi et al.\(^9\), it is also convenient to introduce the jump and average trace operators, which on a generic internal face \( e \in \mathcal{E}_h^0 \), see Figure 1, are defined as:

\[
\left[ q \right] = q^+ n^+ + q^- n^-, \quad \left\{ \cdot \right\} = \frac{(\cdot)^+ + (\cdot)^-}{2}, \tag{25}
\]

where \( q \) denotes a generic scalar quantity and the average operator applies to scalars and vector quantities. By definition, \( \left[ q \right] \) is a vector quantity. These definitions can be suitably extended to faces intersecting \( \partial \Omega \) accounting for the weak imposition of boundary conditions.
Figure 1: Normals at point $P$ on face $e$.

The discrete counterpart of Eq. (22) for a generic element $K \in T_h$ then reads:

$$
\int_K \phi \frac{\partial \mathbf{u}_h}{\partial t} \, d\mathbf{x} - \int_K \nabla \phi \cdot \mathbf{F}(\mathbf{u}_h, \nabla \mathbf{u}_h) \, d\mathbf{x} + \int_{\partial K} \left[ \phi \mathbf{F}(\mathbf{u}_h|_K, \nabla \mathbf{u}_h|_K) \right] \cdot \mathbf{n} \, d\sigma + \int_K \phi \mathbf{s}(\mathbf{u}_h, \nabla \mathbf{u}_h) \, d\mathbf{x} = 0. \tag{26}
$$

Due to the discontinuous function approximation, the flux function in the boundary integral of Eq. (26) is not uniquely defined. Moreover, a consistent and stable discretization of viscous fluxes must account for the effect of interface discontinuities on the gradient $\nabla \mathbf{u}_h$. Uniqueness of interface fluxes is achieved by introducing suitably defined numerical fluxes which couple the solution in adjacent elements and ensure local conservation. As regards the viscous flux discretization we use the BR2 scheme presented in $^2,^4$ and theoretically analyzed in $^9,^{10}$ (where it is referred to as BRMPS).

Summing Eq. (26) over the elements and accounting for the above considerations, we obtain the DG formulation of problem (22), which then requires to find $\mathbf{u}_h^1, \ldots, \mathbf{u}_h^M \in \Phi_h$ such that:

$$
\int_{\Omega} \phi \frac{\partial \mathbf{u}_h}{\partial t} \, d\mathbf{x} - \int_{\Omega} \nabla \phi \cdot \mathbf{F}(\mathbf{u}_h, \nabla \mathbf{u}_h + \mathbf{r}([\mathbf{u}_h])) \, d\mathbf{x} + \int_{\Gamma_h} \left[ \phi \right] \cdot \mathbf{F}(\mathbf{u}_h^+, \nabla \mathbf{u}_h + \eta_e \mathbf{r}_e([\mathbf{u}_h]))^+ \, d\sigma + \int_{\Omega} \phi \mathbf{s}(\mathbf{u}_h, \nabla \mathbf{u}_h + \mathbf{r}([\mathbf{u}_h])) \, d\mathbf{x} = 0, \tag{27}
$$

for all $\phi \in \Phi_h$. The lifting operator $\mathbf{r}_e$, which is assumed to act on the jumps of $\mathbf{u}_h$ componentwise, is defined as the solution of the following problem:

$$
\int_{\Omega} \phi \cdot \mathbf{r}_e(\mathbf{v}) \, d\mathbf{x} = - \int_{\Gamma} \phi \cdot \mathbf{v} \, d\sigma, \quad \forall \phi \in [\Phi_h]^N, \, \mathbf{v} \in \left[ L^1(e) \right]^N, \tag{28}
$$

and the function $\mathbf{r}$ is related to $\mathbf{r}_e$ by the equation:

$$
\mathbf{r}(\mathbf{v}) \overset{\text{def}}{=} \sum_{e \in \mathcal{E}_h} \mathbf{r}_e(\mathbf{v}). \tag{29}
$$
The inviscid and viscous parts of the numerical flux $\hat{F}$ are treated independently. For the former we have used the Godunov flux (alternatively, our code can use the van Leer vector split flux as modified by Hänel\textsuperscript{11}). The numerical viscous flux is given by:

$$\hat{F}(\mathbf{u}_h^\pm, (\nabla_h \mathbf{u}_h + \eta_e \mathbf{r}_e(\|\mathbf{u}_h\|)))^\pm \overset{\text{def}}{=} \{ F(\mathbf{u}_h, \nabla_h \mathbf{u}_h + \eta_e \mathbf{r}_e(\|\mathbf{u}_h\|)) \},$$

where, according to\textsuperscript{9,10}, the penalty factor $\eta_e$ must be greater than the number of faces of the elements. A very interesting feature of the outlined viscous flux discretization scheme is that it couples only the unknowns already coupled by the inviscid flux discretization scheme, irrespective of the degree of polynomial approximation of the solution. This feature is obviously very attractive for an implicit implementation of the method.

At the boundary of the domain, the numerical flux function appearing in Eq. (27) must be consistent with the boundary conditions of the problem. In practice, this is accomplished by properly defining a boundary state which accounts for the boundary data and, together with the internal state, allows to compute the numerical fluxes and the lifting operator on the portion $\Gamma_h^0$ of the boundary $\Gamma_h$, see\textsuperscript{4,12}.

### 3.1 Time discretization and linear system solution

All integrals appearing in Eq. (27) are computed by means of Gauss integration rules with a number of integration points suited for the required accuracy. Cheaper non-product formulae are preferred to tensor product ones when available. The quadrature formulae are taken from the encyclopædia of cubature formulae developed and maintained by Cools\textsuperscript{13}.

The discrete problem corresponding to Eq. (27) can be written as:

$$M \frac{dU}{dt} + R(U) = 0,$$

where $U$ is the global vector of unknown degrees of freedom and $M$ is the global block diagonal mass matrix. Eq. (31) defines a system of (nonlinear) ODEs that for steady solutions on highly stretched grids cannot be solved efficiently using simple explicit schemes. The search for accurate and efficient time integration schemes is the subject of ongoing work and, at present, our DG code implements the following two-stage, second-order implicit Runge-Kutta scheme proposed by Iannelli and Baker\textsuperscript{14}:

$$U^{n+1} - U^n = Y_1 K_1 + Y_2 K_2,$$

$$\begin{bmatrix}
M 
\frac{\Delta t}{\alpha} + \frac{\partial R(U^n)}{\partial U}
\end{bmatrix} K_1 = -R(U^n),$$

$$\begin{bmatrix}
M 
\frac{\Delta t}{\alpha} + \frac{\partial R(U^n)}{\partial U}
\end{bmatrix} K_2 = -R(U^n + \beta K_1),$$

8
\[ \partial \mathbf{R}(\mathbf{U}^n)/\partial \mathbf{U} \] is the Jacobian matrix of the DG space discretization and the constants \( \alpha, \beta, Y_1, Y_2 \) are given by
\[
\alpha = \frac{2 - \sqrt{2}}{2}, \quad \beta = 8\alpha \left( \frac{1}{2} - \alpha \right), \quad Y_1 = 1 - \frac{1}{8\alpha}, \quad Y_2 = 1 - Y_1.
\]

Notice that the one-step backward Euler and Crank-Nicolson schemes can be recovered by setting \( Y_1 = 1, Y_2 = 0, \alpha = 1 \) and \( Y_1 = 1, Y_2 = 0, \alpha = 1/2 \), respectively. Notice also that each step in Eq. (32) requires to solve a linear system of the form \( \mathbf{A} \mathbf{x} = \mathbf{b} \).

To solve Eq. (32) our code can use one of the numerous methods (direct or iterative, sequential or parallel) available in the PETSc\textsuperscript{5} library (Portable Extensible Toolkit for Scientific Computations), the software upon which our DG code relies for the purpose of parallelization. The parallelization is based on grid partitioning accomplished by means of the METIS\textsuperscript{15} package. In this context each processor owns the data related to its local portion of the grid and the data on remote processors are accessed through MPI, the standard for message-passing communication. Thanks to the compactness of our DG method only the data owned by the near neighbor elements at partition boundaries need to be shared among different processes.

4 NUMERICAL RESULTS

To validate the DG solver presented in this work we have considered two complex turbulent flows, namely the internal subsonic flow in the Stanitz\textsuperscript{16} elbow and the external transonic flow around the ONERA M6 wing\textsuperscript{17}. Both test cases have been widely used in the literature to assess the physical and numerical models of CFD codes.

Some computational details concerning the test cases are as follows. All DG solutions have been computed in sequence, starting higher-order solutions from the lower-order ones. Solutions have been advanced in time with the backward Euler scheme and the linear system (32) has been solved using, whenever possible, the default solver available.
in PETSc, i.e. restarted GMRES preconditioned with the block Jacobi method with one block per process, each of which is solved with ILU(0). This solver could be used for the $P_0$ and $P_1$ computations of both test cases. Due to RAM limitations, $P_2$ solutions were computed using less efficient but cheaper alternatives: for the Stanitz elbow the block (per process) Jacobi preconditioner was replaced with the block (per element) diagonal preconditioner, while for the ONERA M6 wing the linear system was solved using the block diagonal preconditioned Richardson iterative method, whereby only the block diagonal portion of the matrix was explicitly stored. All computations have been run on a Linux cluster with 20 AMD Opteron processors and 60 Gbyte RAM.

As regards higher-order computations, two remarks are of order here. First, we found that the transonic computations of the ONERA M6 wing were stable without any control of oscillations or limiting procedure around discontinuities and, in this case, it was therefore possible to avoid resorting to such procedures. Secondly, the geometric representation of solid wall boundaries did not account for wall curvature. This was due to the grids at our disposal and not to limitations of our DG solver. In spite of this flaw the results presented below seem rather accurate and a thorough comparison with results obtained on modified grids accounting for curved boundary representation is presently underway.

4.1 Stanitz elbow

The Stanitz elbow\cite{16} is a duct of rectangular cross-section that was studied at NASA GRC in the early 1950s to examine secondary flow structures in low-speed flow ducts with turning. Figure 2 shows the shape of the elbow and the surface grid. The streamwise cross-section of the elbow is rectangular and the inner (suction) and outer (pressure) surfaces turn the flow approximately 90°. The flow domain includes the constant area “tunnel” section, which in the experimental setup connected the elbow to the plenum chamber, and the elbow extension at the exit. The flow enters the elbow from the bottom in Figure 2(a) and is subsonic throughout the elbow with a maximum Mach number of about 0.3. As the flow accelerates through the elbow, a passage vortex forms near the intersection of the inner and sidewall boundaries and moves towards the mid-span approaching the exit, Figure 2(b). The computational grid is structured and contains $49 \times 60 \times 33$ hexahedra with those subdivisions in the suction-to-pressure, streamwise and spanwise directions, respectively. The entrance tunnel and the elbow extension are discretized with 7 and 4 layers of elements in the streamwise direction, respectively. Elements are clustered near solid walls and the height of the first layer of elements is $h_{\text{min}}/w = 1.21 \times 10^{-4}$, where $w$ is the channel width (i.e. the distance between sidewall walls), and the maximum aspect ratio (maximum to minimum edge length in a given element) is equal to 2.332.

Boundary conditions set total pressure, entropy and tangential velocity components at inlet and static pressure at outlet. According to the experiments, the outlet pressure corresponds to an isentropic Mach number $M_{is} = 0.26$ and the Reynolds number, based on the channel width and the isentropic exit conditions, is $Re_{w,is} = 2,738,800$. Inlet values of $k$ and $\tilde{\omega}$ are computed from prescribed values of turbulence intensity $Tu = 0.01$
and turbulent to laminar viscosity ratio $\mu_t/\mu = 0.1$.

The $P_0$ solution started from uniform conditions (pressure and temperature equal to exit values, velocity in $y$ direction equal to 1% of exit velocity, $Tu = 0.5$ and $\mu_t/\mu = 0.1$) and took 200 time steps to converge to machine precision with a CFL number increasing from 10 up to $2.2 \times 10^{12}$. The restarted GMRES($m$) algorithm used $m = 30$ and a maximum of 60 iterations. Starting from the $P_0$ solution the residuals of the $P_1$ approximation drop to $1 \times 10^{-9}$ in 200 time steps while the CFL number raised up to $1 \times 10^8$ using the above GMRES parameters. Finally, running the $P_2$ solution for 200 more time steps with $CFL = 100$ and using the block diagonal preconditioned GMRES($m$) algorithm (with $m = 30$ and 90 iterations instead of 60), the residuals drop to $2 \times 10^{-6}$.

Figure 3 presents the comparison of experimental and computed pressure ratio distributions for different spanwise sections intersecting the inner and outer walls. The agreement is excellent and the improvement of the more accurate $P_2$ solution on the inner surface can be clearly appreciated.

The prediction of onset and development of the passage vortex, and of the total pressure loss associated with it, has often been considered a challenging problem to test the physical models and the numerical accuracy of CFD codes. Figure 4 presents the comparison of the experimental and computed total pressure loss distribution at the exit plane. The comparison is very favourable and the $P_2$ solution appears clearly capable to predict flow details missing in the more diffusive $P_1$ solution.
Figure 3: Stanitz elbow, pressure coefficient on suction and pressure walls for different spanwise sections.
Figure 4: Stanitz elbow, total pressure loss contours at exit. DG solutions (top) and experimental data (bottom).
4.2 ONERA M6 wing

This case concerns the transonic flow over the ONERA M6 wing. The wing was tested in a wind tunnel at Mach numbers of 0.7, 0.84, 0.88, 0.92, for angles-of-attack up to 6° and Reynolds numbers of about 12 million based on the mean aerodynamic chord. The wind tunnel tests are documented by Schmitt and Charpin\textsuperscript{17}. The ONERA M6 wing is a classic CFD validation case for external flows because of its simple geometry combined with complexities of transonic flow (i.e. local supersonic flow, shocks, and turbulent boundary layers separation). The case here considered uses the flow conditions of Test 2308 of Ref.\textsuperscript{17}, i.e. $M_\infty = 0.8395$, $\alpha_\infty = 3.06^\circ$ and $Re_{c,\infty} = 11.72 \times 10^6$, based on the mean aerodynamic chord $c = 0.64607\text{m}$.

Figure 5(a) shows the structured surface grid on the wing and on the symmetry plane. The grid was taken from the NPARC Alliance Verification and Validation Archive\textsuperscript{18} and is the same grid used by Slater et al.\textsuperscript{19} for the computational study of the transonic, turbulent flow over the ONERA M6 wing by means of the WIND code. The grid consists of four zones containing 294,912 hexahedral elements. The minimum grid spacing normal to the wing surface is $h_{\text{min}}/c = 2.4 \times 10^{-5}$.

Following a local 1D characteristic approach, far-field boundary conditions define the state on the exterior of the domain by combining prescribed values of the Riemann invariants, associated with ingoing characteristics, with those of outgoing waves. Far-field values of $k$ and $\tilde{\omega}$ are computed from prescribed values of turbulence intensity $Tu = 0.001$ and turbulent to laminar viscosity ratio $\mu_t/\mu = 0.1$. 

Figure 5: ONERA M6 wing.
Figure 6: ONERA M6 wing, pressure coefficient on the wing surface at different spanwise sections.
Figure 6: ONERA M6 wing, pressure coefficient on the wing surface at different spanwise sections (cont’d).
The $P_0$ solution started from uniform free-stream conditions and took 300 time steps to converge the residuals to $2 \times 10^{-11}$ with a CFL number increasing from 1 up to $5 \times 10^6$. The restarted GMRES(m) algorithm used $m = 30$ and a maximum of 60 iterations. Starting from the $P_0$ solution the residuals of the $P_1$ approximation drop to $3 \times 10^{-5}$ in 400 time steps while the CFL number raised up to $1 \times 10^2$ using the above GMRES parameters. Finally, running the $P_2$ solution for 800 more time steps with CFL = 15 and using the block diagonal preconditioned Richardson iteration to solve the linear system (with a maximum of 15 iterations), the residuals drop to $3 \times 10^{-4}$.

<table>
<thead>
<tr>
<th></th>
<th>$C_L$</th>
<th>$C_D$</th>
<th>$C_{D_p}$</th>
<th>$C_{D_f}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DG-$P_0$</td>
<td>0.246189</td>
<td>0.0362603</td>
<td>0.0335215</td>
<td>0.00273874</td>
</tr>
<tr>
<td>DG-$P_1$</td>
<td>0.261385</td>
<td>0.0157074</td>
<td>0.0126860</td>
<td>0.00302138</td>
</tr>
<tr>
<td>DG-$P_2$</td>
<td>0.260949</td>
<td>0.0170111</td>
<td>0.0118300</td>
<td>0.00518113</td>
</tr>
</tbody>
</table>

Table 1: Lift and drag coefficients of ONERA M6 wing.

Figure 6 compares the computed pressure coefficient distributions with the experimental data obtained by Schmitt and Charpin\textsuperscript{17} at seven sections along the span of the wing. Both $P_1$ and $P_2$ solutions are capable of reproducing accurately the lambda-shock structure on the suction surface and, unlike many results presented in the literature, the shocks can be clearly distinguished at section $y/b = 0.8$. The lift and drag coefficients of the wing, reported in Table 1, indicate that the $P_1$ and $P_2$ solutions are reasonably close, but
a further refinement (of either the grid or the polynomial approximation) would be useful to be more confident with such results.

The structure and development of the wing tip vortex is presented in Figure 7 by displaying the turbulence intensity contours of the $P_2$ solution. Moving downstream the tip vortex dominates ever larger parts of the wake as can be seen in Figure 7(b).

5 CONCLUSIONS

In this paper we have presented the main features of a high-order implicit DG solver for the RANS and $k$-$\omega$ turbulence model equations. The code has been validated by computing second- and third-order accurate solutions of the 3D turbulent flow in the Stanitz elbow at low Mach number and around the ONERA M6 wing at transonic conditions. The results here presented clearly show the reliability, robustness and accuracy of the DG solver for the simulation of complex 3D high-Reynolds number turbulent flows.

On-going work on the DG code concerns i) the improvement of RAM and CPU requirements of the current implementation of the implicit DG method, ii) the analysis and numerical evaluation of shock capturing techniques applied to high-Reynolds number turbulent flows such those here considered and iii) the assessment of parallel efficiency of the code.

REFERENCES


