V European Conference on Computational Fluid Dynamics
ECCOMAS CFD 2010
J. C. F. Pereira and A. Sequeira (Eds)
Lisbon, Portugal,14-17 June 2010

HIGH-ORDER ACCURATE *P*-MULTIGRID DISCONTINUOUS GALERKIN SOLUTION OF THE RANS AND *K*-ω TURBULENCE MODEL EQUATIONS

F. Bassi*, A. Colombo*, N. Franchina*, A. Ghidoni††, S. Rebay††

*Department of Industrial Engineering, University of Bergamo,
Viale Marconi 5, 24044 Dalmine (BG), Italy
e-mail: {francesco.bassi,alessandro.colombo,nicoletta.franchina}@unibg.it
††Department of Mechanical and Industrial Engineering, University of Brescia
Via Branze 38, 25100, Brescia, Italy
e-mail: {antonio.ghidoni,stefano.rebay}@ing.unibs.it

Key words: Discontinuous Galerkin, high-order, turbulent flow, *p*-multigrid, semi-implicit Runge-Kutta

Abstract. *Discontinuous Galerkin (DG) methods have proven to be perfectly suited for the construction of very high-order accurate numerical schemes on arbitrary unstructured and possibly non conforming grids for a wide variety of applications, but are rather demanding in terms of computational resources. In the last years, in order to improve the computational efficiency of DG methods, multigrid solution strategies, both in the h- and p- variants, have been considered. This work will focus on the solution of the Reynolds averaged Navier-Stokes and k-*ω *turbulence model equations using a p-multigrid algorithm, which is based on a semi-implicit Runge-Kutta smoother for high-order polynomial approximations and the implicit Backward Euler scheme for piecewise constant approximations. The effectiveness of the proposed approach is demonstrated for two 3D test cases.*

1 INTRODUCTION

During the last two decades, the considerable advances in algorithm development and the huge increase of computer power have made CFD a key discipline for the industry. However the numerical technology used in standard industrial codes is still mainly based on formally second-order accurate finite volume or finite element schemes. In practice the accuracy provided by these methods is inadequate in applications such as large eddy simulation, direct numerical simulation, computational aeroacoustic, unless using prohibitively large computational resources, that are beyond the available capabilities. Higher-order accurate methods, such as discontinuous Galerkin (DG) methods, are therefore needed to cope with this class of flow simulations.

The DG method was first introduced by Reed and Hill[1] for the numerical solution of the neutron transport equation. A major step forward in the development of this class of methods is however due to the work of Cockburn, Shu and several collaborators who, in a series of paper starting from the late 80's,[2–5] developed total variation stable DG schemes for the numerical solution of nonlinear conservation laws. DG methods have been successively extended to advection-diffusion[6–8] and to purely elliptic problems,[9,10] and are nowadays adopted for the numerical solution of a wide variety of applications.

DG methods are finite element methods (FEM) in which the solution of the weak or variational form of a problem is approximated by means of piecewise continuous functions inside elements which are in general discontinuous at the interface between neighbouring elements. The lack of a global continuity constraint opens the way to the treatment of the solution at element interfaces by the technique developed in the context of upwind finite volume method, which is in fact a very effective manner to introduce the stabilization required by any FEM for the solution of purely advective or advection dominated problems. DG methods are very popular for their great geometrical flexibility, providing higher-order accurate approximations on general unstructured hybrid grids also in the presence of strongly distorted elements and for grids characterized by a very irregular variation in element size. They are also compact schemes, which is an advantageous feature for implicit time integration and for the parallel implementation.

The price to pay for the robustness, accuracy and flexibility offered by DG methods is their relatively high computational cost and storage requirement. In order to improve the computational efficiency of DG methods, a considerable research effort has been recently devoted to devise more efficient computational strategies, both for the construction of DG space discretization operators and for the integration in time of the space discretized DG equations.

The most commonly considered time integration schemes used with DG space discretization are the explicit multistage Runge-Kutta (RK) methods and the implicit schemes.[11] The former shows extremely slow convergence rate for large scale simulations and/or for high-order polynomial approximations. The latter, even reducing the number of iterations needed to reach the steady state solution, is characterized by an high computational demand, both in terms of the CPU time and of the memory required to store the Jacobian matrix, which may be prohibitive

for a large scale problem and high-order solutions.

In the last years, in order to improve the computational efficiency of DG methods, multigrid (MG) solution strategies,[13–16] both in the h- and p- variants, have been considered. Multigrid methods offer an alternative and efficient approach to steady state solution. For high-order accurate approximations, h- and p-MG solution strategies (or a combination of both) can be considered. While in the classical h-MG method the discrete equations are solved on a series of recursively coarsened grids, in the p-MG algorithm the equations are solved by considering a series of progressively lower order approximations on the same grid. Several p-MG algorithms for DG space approximations have been recently proposed in the literature for both inviscid[17,18] and viscous flows,[16,19] showing that this solution approach is perfectly suited to the DG method.

This work will focus on the solution of the Reynolds averaged Navier-Stokes and k-ω turbulence model equations using p-MG algorithm. We propose an improvement of the p-multigrid scheme originally introduced in,[18] which employs a semi-implicit RK smoother for $\mathbb{P}^k$ polynomial approximation if $k > 0$, and the implicit backward Euler smoother for $\mathbb{P}^0$ polynomial approximation. This strategy allows a significant saving in computer memory, storing only the diagonal part of the Jacobian with respect to the implicit approach. The effectiveness of the proposed approach is demonstrated by computing two 3D test cases.

The organization of this paper is as follows: in Section 2 the DG space discretization is briefly presented, Section 3 describes the p-multigrid algorithm, and in Section 4 the computed results for the subsonic turbulent flow around the ADIGMA[20] BTC0 three dimensional body and DLR F6[21] aeronautical configuration are presented.

2 DG DISCRETIZATION OF RANS AND k-ω EQUATIONS

We will first briefly present the DG space discretization of the RANS and k-ω turbulence model equations. The governing equations can be written as

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_j) = 0, \tag{1}$$

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_j u_i) = -\frac{\partial p}{\partial x_i} + \frac{\partial \widehat{\tau}_{ji}}{\partial x_j}, \tag{2}$$

$$\frac{\partial}{\partial t}(\rho e_0) + \frac{\partial}{\partial x_j}(\rho u_j h_0) = \frac{\partial}{\partial x_j}\left[u_i \widehat{\tau}_{ij} - q_j\right] - \tau_{ij}\frac{\partial u_i}{\partial x_j} + \beta^* \rho \overline{k} e^{\widetilde{\omega}_r}, \tag{3}$$

$$\frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_j}(\rho u_j k) = \frac{\partial}{\partial x_j}\left[(\mu + \sigma^* \overline{\mu}_t)\frac{\partial k}{\partial x_j}\right] + \tau_{ij}\frac{\partial u_i}{\partial x_j} - \beta^* \rho \overline{k} e^{\widetilde{\omega}_r}, \tag{4}$$

$$\frac{\partial}{\partial t}(\rho \widetilde{\omega}) + \frac{\partial}{\partial x_j}(\rho u_j \widetilde{\omega}) = \frac{\partial}{\partial x_j}\left[(\mu + \sigma \overline{\mu}_t)\frac{\partial \widetilde{\omega}}{\partial x_j}\right] + \frac{\alpha}{\overline{k}}\tau_{ij}\frac{\partial u_i}{\partial x_j} - \beta \rho e^{\widetilde{\omega}_r} + (\mu + \sigma \overline{\mu}_t)\frac{\partial \widetilde{\omega}}{\partial x_k}\frac{\partial \widetilde{\omega}}{\partial x_k}, \tag{5}$$

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

$$p = (\gamma - 1)\rho\left(e_0 - u_k u_k/2\right), \tag{6}$$

$$\tau_{ij} = 2\overline{\mu}_t\left[S_{ij} - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right] - \frac{2}{3}\rho\overline{k}\delta_{ij}, \tag{7}$$

$$\widehat{\tau}_{ij} = 2\mu\left[S_{ij} - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right] + \tau_{ij}, \tag{8}$$

$$q_j = -\left(\frac{\mu}{\mathrm{Pr}} + \frac{\overline{\mu}_t}{\mathrm{Pr}_t}\right)\frac{\partial h}{\partial x_j}, \tag{9}$$

$$\overline{\mu}_t = \alpha^*\rho\overline{k}e^{-\widetilde{\omega}_r}, \quad \overline{k} = \max(0,k). \tag{10}$$

Here γ 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 α, α^*, β, β^*, σ, σ^* are those of the high- or low-Reynolds number k-ω model of Wilcox.[23]

Notice that the RANS and k-ω equations here employed are not in standard form,[8] since the variable $\widetilde{\omega} = \log\omega$, instead of ω itself, is used in eqs. (3), (4), (5). The variable $\widetilde{\omega}_r$ in the source terms of eqs. (3), (4), (5) and in the eddy viscosity defined by eq. (10) is introduced to indicate that $\widetilde{\omega}_r$ fulfills suitably defined "realizability" conditions, which in practice put a lower limit on $\widetilde{\omega}$ in such equations. This limitation substantially improves the stability and robustness of turbulent flow computations because there is numerical evidence that too small, though positive, values of $\omega = e^{\widetilde{\omega}}$ can lead to sudden breakdown of computations.

Realizability conditions, which guarantee that the turbulence model predicts positive normal turbulent stresses and satisfies the Schwarz inequality for shear turbulent stresses, lead to the following constraints

$$\frac{e^{\widetilde{\omega}}}{\alpha^*} - 3\left(S_{ii} - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\right) \geq 0, \quad i = 1,2,3, \tag{11}$$

$$\left(\frac{e^{\widetilde{\omega}}}{\alpha^*}\right)^2 - 3\left(S_{ii} + S_{jj} - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\right)\frac{e^{\widetilde{\omega}}}{\alpha^*} + 9\left[\left(S_{ii} - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\right)\left(S_{jj} - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\right) - S_{ij}^2\right] \geq 0, \quad i,j = 1,2,3, \quad i \neq j. \tag{12}$$

Denoting with a the maximum value of the unknown $e^{\widetilde{\omega}}/\alpha^*$ corresponding to the zeros of

eqs. (11) and (12), the lower bound $\widetilde{\omega}_{r0}$ that guarantees realizable turbulent stresses is given by

$$\frac{e^{\widetilde{\omega}_{r0}}}{\alpha^*} = a, \tag{13}$$

and $\widetilde{\omega}_r$ in eqs. (3), (4), (5) and (10) is then set as

$$\widetilde{\omega}_r = \max\left(\widetilde{\omega}, \widetilde{\omega}_{r0}\right). \tag{14}$$

ω values set at solid walls are dependent on the degree of polynomial approximation. For this purpose we define

$$\int_{y_1} \phi\, \widetilde{\omega}\, \mathrm{d}y = \int_{y_1} \phi\, \widetilde{\omega}_{ex}\, \mathrm{d}y, \tag{15}$$

where $\widetilde{\omega}_{ex}$ is the near-wall analytical behavior of $\widetilde{\omega}$, *i.e.*,

$$\widetilde{\omega}_{ex} = \log\left(\frac{6\nu}{\beta}\right) - 2\log y, \tag{16}$$

and ϕ is the one-dimensional polynomial basis adopted to define $\widetilde{\omega}$. From eq. (15) we can then compute $\widetilde{\omega}_w^k = \widetilde{\omega}(0)$ for any desired polynomial degree k.

DG space discretization of the RANS and k-ω turbulence model equations can be written in compact form as

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla\cdot\mathbf{F}_c(\mathbf{u}) + \nabla\cdot\mathbf{F}_v(\mathbf{u},\nabla\mathbf{u}) + \mathbf{s}(\mathbf{u},\nabla\mathbf{u}) = \mathbf{0}, \tag{17}$$

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 weak form of the system (17) reads

$$\int_\Omega \phi\frac{\partial \mathbf{u}}{\partial t}\,\mathrm{d}\mathbf{x} - \int_\Omega \nabla\phi\cdot\mathbf{F}(\mathbf{u},\nabla\mathbf{u})\,\mathrm{d}\mathbf{x} + \int_{\partial\Omega}\phi\mathbf{F}(\mathbf{u},\nabla\mathbf{u})\cdot\mathbf{n}\,\mathrm{d}\sigma + \int_\Omega \phi\mathbf{s}(\mathbf{u},\nabla\mathbf{u})\,\mathrm{d}\mathbf{x} = \mathbf{0}, \tag{18}$$

for any arbitrary, sufficiently smooth, test function ϕ ($\mathbf{F}$ is the sum of the inviscid and viscous fluxes). The DG discretization of eq. (18) is defined on a triangulation $\mathscr{T}_h = \{K\}$ of an approximation Ω_h of Ω and consists of a set of non-overlapping elements K not necessarily simplexes. The following space setting of discontinuous piecewise polynomial functions for each component $u_{h_i} = u_{h_1},\ldots,u_{h_M}$ of the numerical solution $\mathbf{u}_h$ is assumed:

$$u_{h_i} \in \Phi_h \stackrel{\text{def}}{=} \left\{\phi_h \in L^2(\Omega) :\, \phi_h|_K \in \mathrm{P}^k(K)\ \forall K\in\mathscr{T}_h\right\} \tag{19}$$

for some polynomial degree $k\geq 0$, being $\mathrm{P}^k(K)$ the space of polynomials of global degree at most k on the element K.

The discrete counterpart of the eq. (18) is therefore obtained by replacing $\phi \leftarrow \phi_h$ and $\mathbf{u} \leftarrow \mathbf{u}_h$, for each element $K \in \mathcal{T}_h$,

$$\int_K \phi_h \frac{\partial \mathbf{u}_h}{\partial t}\,\mathrm{d}\mathbf{x} - \int_K \nabla_h \phi_h \cdot \mathbf{F}(\mathbf{u}_h, \nabla_h \mathbf{u}_h)\,\mathrm{d}\mathbf{x} + \int_{\partial K} \phi_h \mathbf{F}(\mathbf{u}_h|_K, \nabla_h \mathbf{u}_h|_K)\cdot \mathbf{n}\,\mathrm{d}\sigma + \int_K \phi_h \mathbf{s}(\mathbf{u}_h, \nabla \mathbf{u}_h)\,\mathrm{d}\mathbf{x} = \mathbf{0}. \quad (20)$$

The discontinuous approximation of the numerical solution requires introducing a special treatment of the inviscid interface flux and of the viscous flux. For the former it is common practice to use suitably defined numerical flux functions (the exact Godunov flux function has been used) which ensure conservation and account for wave propagation. For the latter we employ the BR2 scheme[12,22] to obtain a consistent, stable and accurate discretization of the viscous flux. Summing eq. (20) over the elements and accounting for these aspects, the DG formulation of problem (18) then requires to find $u_{h_1}, \ldots, u_{h_M} \in \Phi_h$ such that

$$\int_{\Omega_h} \phi_h \frac{\partial \mathbf{u}_h}{\partial t}\,\mathrm{d}\mathbf{x} - \int_{\Omega_h} \nabla_h \phi_h \cdot \mathbf{F}\left(\mathbf{u}_h, \nabla_h \mathbf{u}_h + \mathbf{r}([\![\mathbf{u}_h]\!])\right)\,\mathrm{d}\mathbf{x} + \int_{\Gamma_h} [\![\phi_h]\!] \cdot \widehat{\mathbf{F}}\left(\mathbf{u}_h^{\pm}, (\nabla_h \mathbf{u}_h + \eta_e \mathbf{r}_e([\![\mathbf{u}_h]\!]))^{\pm}\right)\,\mathrm{d}\sigma + \int_{\Omega_h} \phi_h \mathbf{s}\left(\mathbf{u}_h, \nabla_h \mathbf{u}_h + \mathbf{r}([\![\mathbf{u}_h]\!])\right)\,\mathrm{d}\mathbf{x} = \mathbf{0}, \quad (21)$$

for all $\phi_h \in \Phi_h$. In eq. (21) we have introduced the following jump and average trace operators

$$[\![q]\!] \stackrel{\text{def}}{=} q^+ \mathbf{n}^+ + q^- \mathbf{n}^-, \qquad \{\cdot\} \stackrel{\text{def}}{=} \frac{(\cdot)^+ + (\cdot)^-}{2}, \quad (22)$$

where q denotes a generic scalar quantity and the average operator applies to scalars and vector quantities. These definitions can be suitably extended to faces intersecting $\partial\Omega$ accounting for the weak imposition of boundary conditions. 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:

$$\int_{\Omega_h} \phi_h \cdot \mathbf{r}_e(\mathbf{v})\,\mathrm{d}\mathbf{x} = -\int_e \{\phi_h\} \cdot \mathbf{v}\,\mathrm{d}\sigma, \forall \phi_h \in [\Phi_h]^N,\ \mathbf{v} \in \left[L^1(e)\right]^N, \quad (23)$$

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

$$\mathbf{r}(\mathbf{v}) \stackrel{\text{def}}{=} \sum_{e \in \mathcal{T}_h} \mathbf{r}_e(\mathbf{v}). \quad (24)$$

η_e is the penalty factor and must be greater than the number of faces of the elements.

By assembling together all the elemental contributions, the system of ordinary differential equations governing the evolution in time of the discrete solution can be written as

$$\mathbf{M}\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}t} + \mathbf{R}(\mathbf{U}) = \mathbf{0}, \quad (25)$$

where **U** is the global vector of unknown degrees of freedom, **M** is the global block diagonal mass matrix and **R** the residual vector.

In this work modal expansion bases defined in the physical space have been used and all integrals appearing in eq. (21) are computed by means of Gauss quadrature formulae with a number of points consistent with the accuracy required.

3 THE p-MULTIGRID SOLUTION STRATEGY

The standard iterative solvers are very effective at eliminating the high-frequency or oscillatory components of the error, while leaving the low-frequency or smooth components relatively unchanged. In order to remove all the error components, in the h-MG method a sequence of progressively coarsened grids is adopted. This procedure greatly increases the scheme convergence rate, since a smooth error mode that is hard to remove on a given grid becomes an oscillatory mode that is easy to remove on a coarser grid. In the p-MG method, the same idea is applied on a sequence of progressively lower order approximations.

The p-MG algorithm is particularly attractive for DG approximations because the restriction and prolongation operators required to transfer the solution and the error between the various approximation levels are simply L_2 projection operators. This means that the transfer operators are block diagonal matrices, and that the transfer operations can be performed in a very efficient manner.

The various levels can be visited following different paths. In the commonly considered V-cycle and W-cycle, the algorithm visits the various levels, as depicted in Figure 1. At each level, a number ν_1 of pre-smoothing iterations is performed prior to restricting the solution to the next coarser level (bullets), while, on the way back to "finer" levels, a number ν_2 of post-smoothing iterations is performed after prolongation (circles).

As a further improvement of the method, in the full multigrid (FMG) algorithm the coarser level solutions are exploited to obtain good initial guess to initialize the computation of the finer grids, as depicted in Figure 3. Notice that it is not necessary to compute a fully converged solution on each level before passing to the next finer level, since the discretization error on coarser levels can be relatively large and there is no point in achieving a convergence tolerance that is better than the approximation error. In the proposed algorithm this issue is addressed by deciding to prolongate the solution at the current iterate to the next finer level if a residual-based criterion is met, as described in.[16]

3.1 The p-Multigrid algorithm

The entire multigrid strategy is based on a recursive application of the so called two-level algorithm, in which the "exact" solution on the coarser grid is used to accelerate the solution on the finer grid. To avoid the prohibitively expensive exact solution on the coarse grid, the two-level algorithm is recursively applied to progressively coarser grids thus arriving at the previously described V-cycle, W-cycle, and full multigrid algorithm. The two level scheme suited for nonlinear problems is called Full Approximation Storage scheme (FAS).

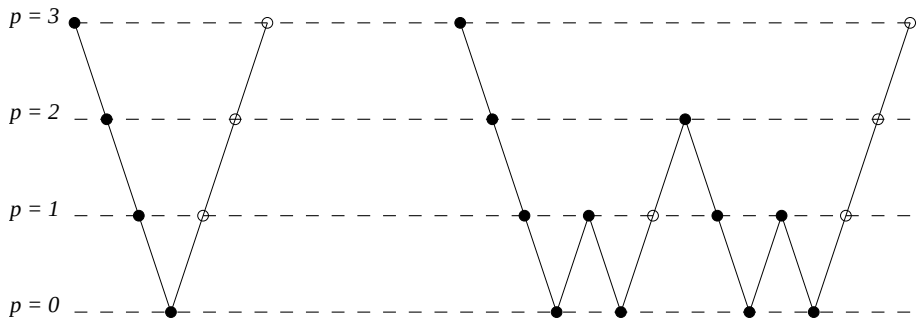

Figure 1: V-cycle and W-cycle for $p = 3$ (•: pre-smoothing; ◦: post-smoothing)

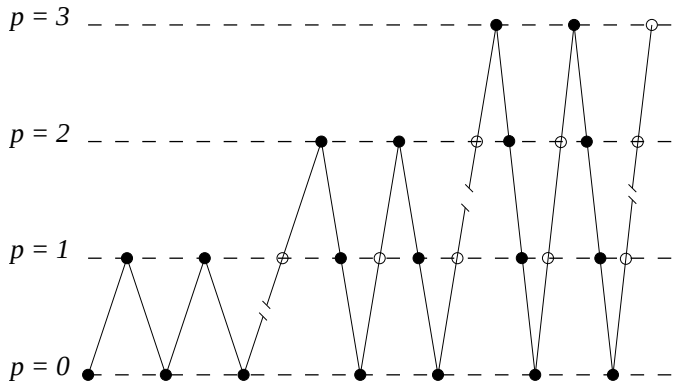

Figure 2: V cycle full multigrid for $p = 3$ (•: pre-smoothing; ◦: post-smoothing)

In order to illustrate the FAS scheme, let

$$\mathbf{A}^p(\mathbf{u}^p) = \mathbf{f}^p \tag{26}$$

denote a generic nonlinear problem, where $\mathbf{u}^p \in \mathbb{P}^p$ is the discrete solution and $\mathbf{A}^p(\mathbf{u}^p)$ is the associated non linear algebraic operator. Notice that the superscript p indicates both the level in the p-MG algorithm and the degree of the polynomial approximation employed in the DG approximation. Let $\mathbf{v}^p$ be the approximate solution of problem (26) during the iterative solution process, and let

$$\mathbf{r}^p(\mathbf{v}^p) = \mathbf{f}^p - \mathbf{A}^p(\mathbf{v}^p)$$

denote the residual vector.

In the basic two-level multigrid method, the exact solution on the coarse level is used to correct the solution on the fine level. The correction is performed according to the following steps:

- restrict the solution and the residual to the coarse level,

$$\mathbf{v}_0^{p-1} = \widetilde{\mathbf{I}}_p^{p-1}\mathbf{v}^p, \quad \mathbf{r}^{p-1} = \mathbf{I}_p^{p-1}\mathbf{r}^p(\mathbf{v}^p), \tag{27}$$

where $\widetilde{\mathbf{I}}_p^{p-1}$ and $\mathbf{I}_p^{p-1}$ are the solution and the residual restriction operators from level p to level $p-1$, respectively;

- compute the forcing term for the coarse level:

$$\mathbf{s}^{p-1} = \mathbf{A}^{p-1}(\mathbf{v}_0^{p-1}) - \mathbf{r}^{p-1}; \tag{28}$$

- solve the coarse level problem:

$$\mathbf{A}^{p-1}(\mathbf{v}^{p-1}) = \mathbf{f}^{p-1} + \mathbf{s}^{p-1}; \tag{29}$$

- calculate the coarse grid error:

$$\mathbf{e}^{p-1} = \mathbf{v}^{p-1} - \mathbf{v}_0^{p-1}; \tag{30}$$

- prolongate the coarse grid error and correct the fine level approximation:

$$\mathbf{v}^{p} = \mathbf{v}^{p} + \widetilde{\mathbf{I}}_{p-1}^{p}\mathbf{e}^{p-1} \tag{31}$$

where $\widetilde{\mathbf{I}}_{p-1}^{p}$ is the error prolongation operator.

The solution restriction and prolongation operators, $\widetilde{\mathbf{I}}_p^{p-1}$ and $\widetilde{\mathbf{I}}_{p-1}^{p}$, are simply L^2 projections onto the low-order and high-order spaces. An explicit expression of the residual restriction operator $\mathbf{I}_p^{p-1}$ can be obtained following the approach proposed by Fidkowski, see e.g.,[16] which shows that in fact $\mathbf{I}_p^{p-1} = (\widetilde{\mathbf{I}}_{p-1}^{p})^T$.

3.2 Smoothers

At all the levels p with the exception of the coarsest level p_{min}, i.e. for $p_{min}+1 \leq p \leq p_{max}$, the smoother used in the p-MG scheme is the m-stage semi-implicit RK scheme introduced in,[15,18] which can be written as

$$\begin{array}{l}
\mathbf{u}^0 = \mathbf{u}^n \\
\texttt{DO}\ k = 1,\ m \\
\qquad \left[\mathbf{M} + \alpha_k \Delta t D(\mathbf{u}^0)\right] \delta\mathbf{u}^k = -\mathbf{M}(\mathbf{u}^{k-1} - \mathbf{u}^0) - \alpha_k \Delta t \mathbf{r}(\mathbf{u}^{k-1}) \\
\qquad \mathbf{u}^k = \mathbf{u}^{k-1} + \delta\mathbf{u}^k \\
\texttt{END DO} \\
\mathbf{u}^{n+1} = \mathbf{u}^m.
\end{array} \tag{32}$$

$D(\mathbf{u}^0)$ is the block diagonal part of the full Jacobian matrix, which is computed only for the first stage.

At the coarsest level $p = p_{min}$, an implicit iterative smoother based on the linearized Backward Euler scheme is instead used. With reference to Eq. (25), the smoother can be written as

$$\left(\frac{\mathbf{M}^{p_{min}}}{\Delta t} + \frac{\partial \mathbf{r}(\mathbf{u}^{p_{min}})}{\partial \mathbf{u}}\right) \Delta \mathbf{u}^{p_{min}} + \mathbf{r}(\mathbf{u}^{p_{min}}) = 0, \tag{33}$$

where $\mathbf{M}^{p_{min}}$ denotes the mass matrix, $\mathbf{u}^{p_{min}}$ the vector of the unknowns and $\mathbf{r}(\mathbf{u}^{p_{min}})$ the residual vector at level $p = p_{min}$. The fully coupled linear system is solved by means of the PGMRES[11] method with the incomplete LU factorization as preconditioner.

4 NUMERICAL RESULTS

This section presents the results for two shockless testcases, the turbulent flows around BTC0 three dimensional body and around the DLR F6 aeronautical configuration. These tests have been computed with the FMG V-cycle and the solution is prolongated to the finer level when a residual-based criterion is met (see Seciton 3).

The following smoothing strategy has been considered:

- the five stage semi-implicit Runge-Kutta (SIRK5) smoother described in Section 3.2 with the coefficients values $\alpha_1 = 0.2$, $\alpha_2 = 0.25$, $\alpha_3 = 0.3333$, $\alpha_4 = 0.5$ and $\alpha_5 = 1.0$ for all levels $p_{min} < p \leq p_{max}$;
- the implicit Backward Euler (BE) smoother for the coarsest level p_{min}.

Table 1 presents the number of pre and post iterations adopted at each level. ν_0 represents the number of iteration at the coarsest level p_{min}, ν_1 and ν_2 the number of pre/post iteration at level $p > p_{min}$. These values have been empirically determined in order to minimize the CPU time needed to reach a converged solution. .

Level	ν_0	ν_1	ν_2
$p = p_{min}$	1	-	-
$p_{min} < p < p_{max}$	-	2	1
$p = p_{max}$	-	1	1

Table 1: Number of pre/post iterations at each level p

4.1 BTC0

In the first test case the turbulent flow around the BTC0 three dimensional body is computed for a farfield Mach number $M_\infty = 0.5$, an angle of attack $\alpha = 5^\circ$ and a Reynolds number $Re = 10^7$. The geometry has been represented with bi-quadratic faces and two grids have been considered. The coarse grid is composed of 832 hexahedral elements with a maximum stretching factor value equal to 8850, while the fine grid has 6656 hexahedral elements with a maximum stretching factor value equal to 20000. $\mathbb{P}^0$ has been considered as coarsest level.

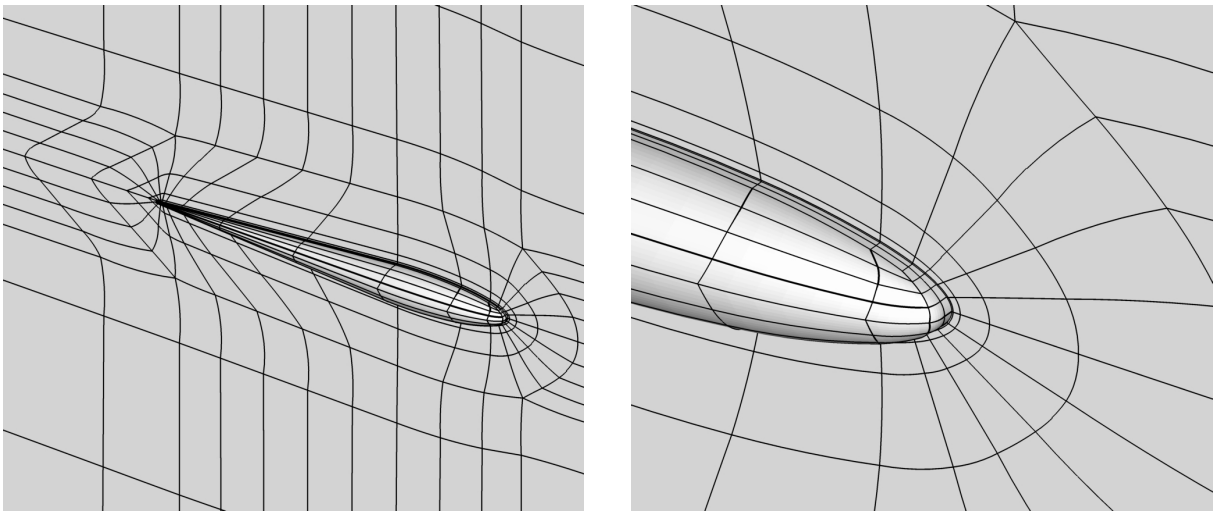

Figure 3: Turbulent BTC0: 6656 hexahedral elements mesh

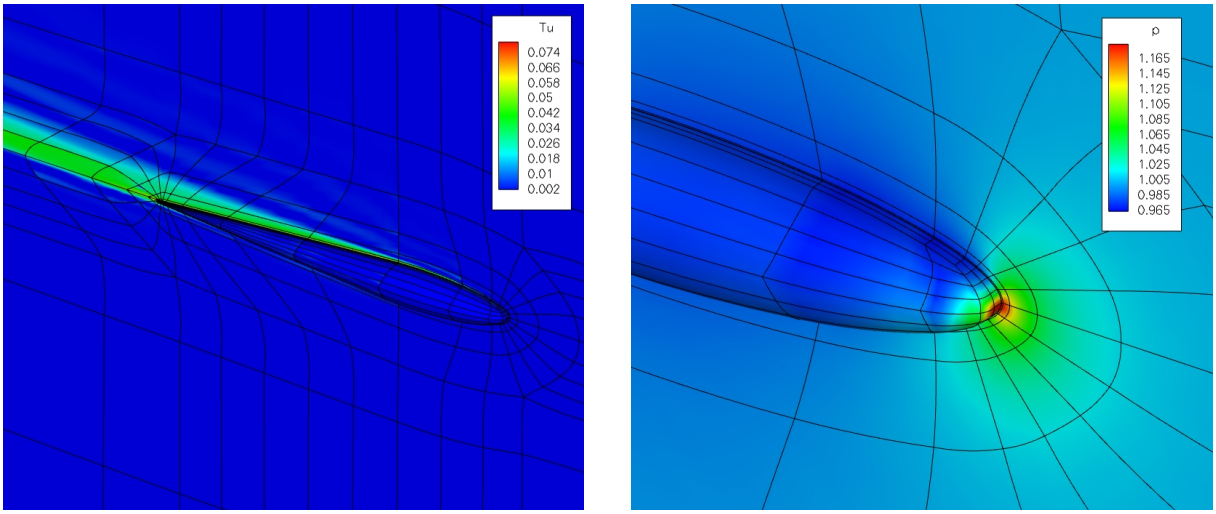

Figure 4: Turbulent BTC0: turbulence intensity (left) and pressure (right) contours of $\mathbb{P}^3$ solution on the fine mesh

Figure 3 shows details of the fine grid, while Figure 4 illustrates the corresponding pressure (right) and turbulence intensity (left) contours for $\mathbb{P}^3$ spatial discretization.

Figure 5 (left) illustrates the $log(\omega)$ residual L_2 norm convergence history as a function of the MG iterations of the SIRK5+BE scheme for different solution approximation $\mathbb{P}^k$ (k=1,...,3) on the coarse grid, while Table 2 presents the number of MG cycles needed to converge and the convergence curve slope σ for every level. Notice that the polynomial order independent property is not satisfied for $\mathbb{P}^3$ solution approximation, due to a deterioration of the convergence rate. It is also shown that the memory requirement of the p-MG algorithm is $\approx 50\%$ of that needed by an implicit scheme. Figure 5 (right) shows, instead, the residuals L_2 norm convergence history in term of CPU time for $\mathbb{P}^3$ solution approximation, computed on the coarse grid.

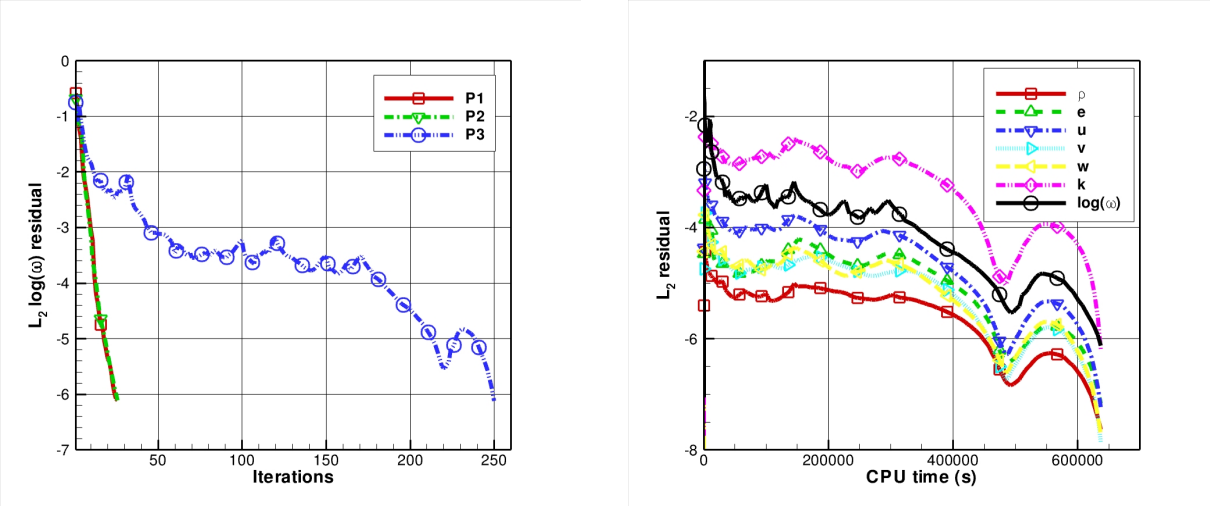

Figure 5: Turbulent BTC0: $log(\omega)$ residual L_2 norm convergence history of the SIRK5+BE as a function of MG cycles on the coarse mesh for different solution approximation $\mathbb{P}^k$ (left). Residuals L_2 norm convergence history versus CPU time on the coarse mesh for $\mathbb{P}^3$ solution approximation (right)

$\mathbb{P}^k$	1	2	3
N_{MG}	23	23	248
σ	-0.215	-0.214	-0.012
Mem_{MG}[Gb]	0.48	0.96	2.72
Mem_{IMP}[Gb]	0.64	1.92	6.08

Table 2: Turbulent BTC0: convergence summary and memory requirement for each polynomial approximation $\mathbb{P}^k$. N_{MG} is the number of MG cycles needed to converge, σ is the slope of the convergence curve, Mem_{MG}[Gb] the p-MG algorithm memory requirement, Mem_{IMP}[Gb] the memory requirement of an implicit scheme.

4.2 DLR-F6

The DLR-F6 aircraft has been the focus of several wind-tunnel tests and computational studies.[21,24] The design cruise Mach number is $M_\infty = 0.75$, the lift coefficient is $C_L = 0.5$ and the chord Reynolds is $Re = 5 \times 10^6$. Since the associated flow regime past the aircraft is transonic , the free stream Mach number has been reduced to 0.5 in order to avoid the presence of shocks. The computation has been carried out on a mesh of 50618 elements (Figure 6), consisting of hexahedral elements with a third order representation of the boundary. Computation has been performed up to $\mathbb{P}^3$ solution approximation and the coarsest level is taken equal to $\mathbb{P}^0$. Figure 7 show the pressure (left) and Mach (right) contours.

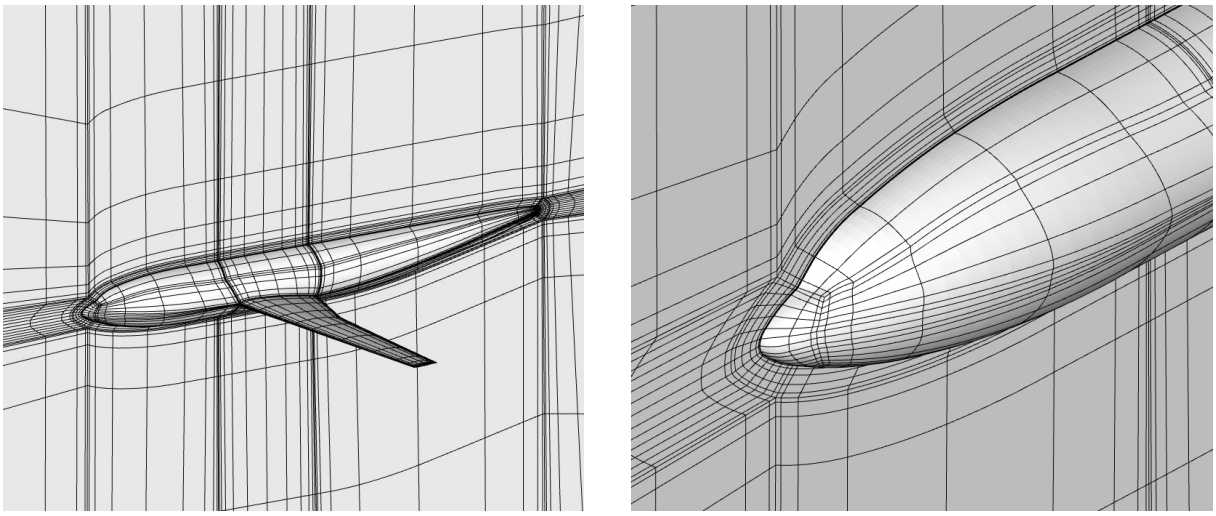

Figure 6: DLR-F6: 50618 hexahedral elements mesh

The convergence rate behaviour is similar to that observed for the BTC0 test case and the polynomial order independent property is not fulfilled. The memory saving with respect to implicit schemes is about 50%.

5 CONCLUSIONS

A p-multigrid discontinuous Galerkin algorithm for the solution of the steady state Reynolds averaged Navier-Stokes and k-ω turbulence model equations has been proposed. The performance of the proposed approach has been assessed by computing two different shockless turbulent test cases. The approach is characterized by a strong reduction of the memory requirement with respect to implicit solvers ($\approx 50\%$). At this time, however, the numerical results indicate that the proposed p-MG strategy displays a convergence rate that degrades for the highest degrees of the polynomial approximation here considered.

Future work will focus on the investigation of the proposed p-MG algorithm to enhance the convergence rate at all polynomial levels, and also on the implementation within this strategy of a robust shock-capturing technique, already developed for implicit schemes.

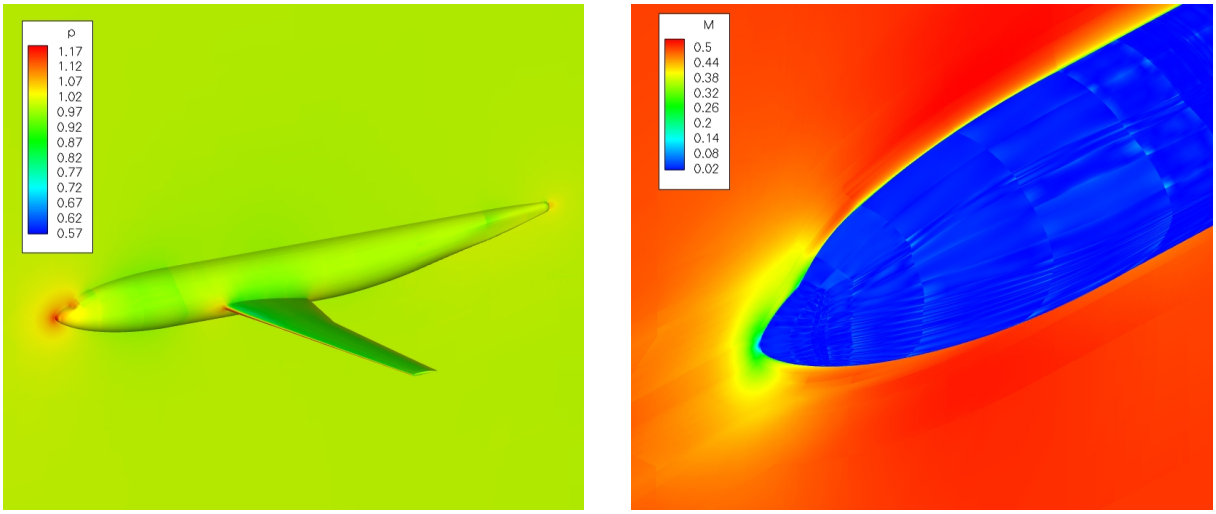

Figure 7: DLR-F6: pressure (left) and Mach (right) contours of $\mathbb{P}^3$ solution

REFERENCES

[1] W. H. Reed and T. R. Hill, Triangular mesh methods for the neutron transport equation, *Technical Report*, LA-UR-73-479, Los Alamos Scientific Laboratory (1973).

[2] B. Cockburn, C.W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: General framework, *Mathematics of Computation*, **52**, 411–435 (1989).

[3] B. Cockburn, C.W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One dimensional systems. *Journal of Computational Physics*, **84**, 90–113 (1989).

[4] B. Cockburn, S. Hou, C.W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case, *Mathematics of Computation*, **54**, 545–581 (1990).

[5] B. Cockburn, S.Y. Lin, C.W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element for conservation laws V: Multidimensional systems, *Journal of Computational Physics*, **141**, 199–224 (1998).

[6] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations, *Journal of Computational Physics*, **131**, 267–279 (1997).

[7] B. Cockburn, C.W. Shu, The local discontinuous Galerkin method for time-dependent convection-diffusion systems, *SIAM Journal on Numerical Analysis*, **35**, 2440–2463 (1998).

[8] F. Bassi, A. Crivellini, S. Rebay, M. Savini, Discontinuous Galerkin solution of the Reynolds averaged Navier-Stokes and k-ω turbulence model equations, *Computers & Fluids*, **34**, 507–540 (2005).

[9] D.N. Arnold, F. Brezzi, B. Cockburn, D. Marini, Discontinuous Galerkin methods for elliptic problems, *Lecture Notes in Computational Science and Engineering, vol. Discontinuous Galerkin Methods: Theory, Computation and Applications*, Springer-Verlag, New York (2000).

[10] D.N. Arnold, F. Brezzi, B. Cockburn, D. Marini, Unified analysis of discontinuous Galerkin method for elliptic problems, *SIAM Journal on Numerical Analysis*, **39**, 1749–1779 (2002).

[11] F. Bassi, S. Rebay, GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations, *Lecture Notes in Computational Science and Engineering, vol. Discontinuous Galerkin Methods: Theory, Computation and Applications*, Springer-Verlag, New York (2000).

[12] F. Bassi, S. Rebay, A High Order Discontinuous Galerkin Method for Compressible Turbulent Flows, *Lecture Notes in Computational Science and Engineering, vol. Discontinuous Galerkin Methods: Theory, Computation and Applications*, Springer-Verlag, New York (2000).

[13] B.T. Helenbrook, D.J. Mavriplis, H.A. Atkins, Analysis of p-multigrid for continuous and discontinuous finite element discretizations, In proceedings of the*16th AIAA Computational Fluid Dynamics Conference*, AIAA, Orlando, Florida (2003).

[14] J.J. van der Vegt, H. van der Ven, Space–time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows: I general formulation, *Journal of Computational Physics*, **182**, 546–585 (2002).

[15] F. Bassi, S. Rebay, Numerical solution of the Euler equations with a multiorder discontinuous finite element method. In proceedings of the *Proceedings of the Second International Conference on Computational Fluid Dynamics*, Computational Fluid Dynamics 2002, S. Armfield, P. Morgan, K. Srinivas (eds.), Springer Verlag, Sydney (2002).

[16] K.J. Fidkowski, T.A. Oliver, J. Lu, L. Darmofal, p-Multigrid solution of high-order discontinuous Galerkin discretizations of the compressible Navier-Stokes equations, *Journal of Computational Physics*, **207**, 92–113 (2005).

[17] H. Luo, J.D. Bauman, R. Lohner, A p-multigrid discontinuous Galerkin method for the Euler equations on unstructured grids, *Journal of Computational Physics*, **211**, 767–783 (2006).

[18] F. Bassi, A. Ghidoni, S. Rebay, P. Tesini, High-order accurate *p*-multigrid discontinuous Galerkin solution of the Euler equations, *International Journal for Numerical Methods in Fluids*, **60**, 847–865 (2009).

[19] K. Shahbazi, D.J. Mavriplis, N.K. Burgess, Multigrid algorithms for high-order discontinuous Galerkin discretizations of the compressible Navier-Stokes equations, *Journal of Computational Physics* **228**, 7917–7940 (2009).

[20] N. Kroll, ADIGMA-A European project on the development of adaptive higher-order variational methods for aerospace applications, In proceedings of the*47th AIAA Aerospace Sciences Meeting*, AIAA, Orlando, Florida (2009).

[21] O. Brodersen, A. Sturmer, Drag prediction of engine-airframe interface effects using unstructured Navier-Stokes calculations, In proceedings of the*19th AIAA Applied Aerodynamics Conference*, AIAA, Anaheim, California (2001).

[22] F. Bassi, S. Rebay, G. Mariotti, S. Pedinotti, M. Savini, A High-Order Accurate Discontinuous Finite Element Method for Inviscid and Viscous Turbomachinery Flows, In proceedings of the *2nd European Conference on Turbomachinery Fluid Dynamics and Thermodynamics*, R. Decuypere, G. Dibelius (eds.), Antwerpen, Belgium (1997).

[23] D.C. Wilcox, Turbulence Modelling for CFD, *DCW industries Inc.* (1993).

[24] E. Lee-Rausch, N. Frink, D. Mavriplis, R. Rausch, W. Milholen, Transonic drag prediction on a DLR-F6 transport configuration using ubstructured grid solvers, *Computer & Fluid*, **38**, 511–532 (2009).