A DISCONTINUOUS GALERKIN METHOD FOR LAMINAR LOW MACH NUMBER FLOWS

A. Nigro[1], S. Renda[1], C. De Bartolo[1], R. Hartmann[2], F. Bassi[3]

[1]Dipartimento di Meccanica, Università della Calabria, Ponte P. Bucci cubo 44/C, 87036 Rende, CS, Italy
[2]Institute of Aerodynamics and Flow Technology, German Aerospace Center (DLR), Lilienthalplatz 7, 38108 Braunschweig, Germany
[3]Dip. di Ingegneria Industriale, Università di Bergamo, viale Marconi 5, 24044 Dalmine, BG, Italy

SUMMARY

In this paper we present a Discontinuous Galerkin (DG) method designed to improve the accuracy and efficiency of laminar flow simulations at low Mach numbers using a fully implicit scheme. The algorithm is based on the flux preconditioning approach, which modifies only the dissipative terms of the numerical flux. This formulation is quite simple to implement in existing implicit DG codes, it overcomes the time-stepping restrictions of explicit multistage algorithms, is consistent in time and thus applicable to unsteady flows. The performance of the method is demonstrated by solving a laminar flow past a NACA0012 airfoil at different low Mach numbers using various degrees of polynomial approximations. Computations with and without flux preconditioning are performed on different grid topologies to analyze the influence of the spatial discretization on the accuracy of the DG solutions at low Mach numbers.

1. INTRODUCTION

Low Mach number flows or locally incompressible flows typologies are manifold and of great practical interest: high-speed flows with large embedded regions of low velocity, low-speed flows that are compressible due to density changes induced by heat sources, problems where compressible and incompressible flow at varying Mach numbers occur side by side. Typical industrial examples of compressible low speed flows can be found in natural convection in gas or liquid phase, subsonic combustion, heat engines or burners and heat transfer, in heat exchangers, in rocket motor flows where the Mach number is zero at the closed end and supersonic at the divergent nozzle exit, in high speed flows with large embedded recirculation zones, in flow over a wing at high angle of attack and others.

Nevertheless, algorithms used for compressible flows suffer from a lack of accuracy and slow convergence to solve low Mach number flows in which the density is almost constant [1]. The reason for the bad convergence is the large disparity between acoustic and convective wave speeds that causes the governing equations to be ill-conditioned (stiffness problem). The decreasing accuracy results from a lack of artificial dissipation for small Mach number, as addressed by Guillard and Viozat [2] for upwind schemes with their asymptotic analysis for the Euler equations. The most general approach to overcome the stiffness problem is based on the preconditioning strategy. This technique artificially modifies the acoustic wave speeds of the governing equations drastically reducing the condition number and improving the convergence process. However, the time derivative preconditioning destroys the time accuracy and it can be applied to steady-state simulations only. Furthermore, the accuracy for nearly incompressible inviscid and viscous flows can be improved by preconditioning, modifying the inviscid dissipation term of the numerical flux function. In particular, the preconditioned governing equations preserve the accuracy of numerical solutions. Some of the most recognized local preconditioners for laminar flows were proposed by Choi and Merkle [3], Turkel [4,5], Lee and van Leer [6] and Weiss and Smith [7], respectively.

In this paper we extend the DG low Mach number preconditioning technique already implemented by the authors for Euler equations [8] to the viscous flows. The Bassi and Rebay's BR2 scheme is employed for the dicretization of the diffusive terms [9]. The conservative Navier-Stokes equations are written in terms of primitive variables, more suitable for low Mach number computations, and iterated to the numerical solution using an implicit scheme. According to [10], explicit integration schemes suffer from severe time stepping restrictions computing low speed flows, hence an implicit scheme is more appropriate for this kind of computations. In particular, we found [8] that preconditioning only needs to be applied to the numerical flux function (flux preconditioning technique) for guaranteeing accuracy and convergence rate improvements. Furthermore, this formulation is consistent in time and could directly be used to compute unsteady low Mach number flows without resorting to dual time-stepping techniques [7].

The aim of this paper is to assess the capability of the DG method in solving laminar low Mach number flows. The method yielding a minimal amount of dissipation is well suited for the solution of the Navier-Stokes equations in the low Mach number limit. The results show that accurate solutions on relatively coarse quadrangular and triangular meshes can be computed using a high-order representation of the unknowns, highlighting the influence of the spatial discretization on the lack of accuracy exhibit by the upwind schemes at low Mach numbers. Furthermore, we demonstrate that the flux preconditioning approach clearly improves both the accuracy and efficiency of the viscous DG solver.

The outline of the paper is as follows. In section 2 we present the compressible Navier Stokes equations in conservative formulation and primitive variables. In Section 3 we present the DG BR2 discretization of the governing equations, the boundary conditions and the preconditioned numerical flux function. In Section 4 we give some detail on the implicit time stepping schemes employed. The performance of the numerical scheme is then demonstrated in Section 5 by computing a viscous flow around a NACA0012 airfoil for different low Mach numbers, grid topologies and degrees of polynomial approximation. Finally, a few conclusions are drawn in Section 6.

2. GOVERNING EQUATIONS

The conservative variables are commonly used in compressible flow computation because they are more accurate to capture shocks. Nevertheless, as Mach number tends to zero, the density becomes constant and the choice of pressure p as a dependent variable proved to be more adequate [11]. Moreover, for viscous flow, temperature gradients have to be computed for the thermal diffusion terms, so it is also more convenient to work with temperature. Therefore we express the compressible Navier-Stokes equations in conservative form based on the set of primitive variables $\boldsymbol{q} = [p, u, v, T]^T$ as follows:

$$\boldsymbol{\Gamma}\frac{\partial \boldsymbol{q}}{\partial t} + \nabla\cdot\boldsymbol{F}_c(\boldsymbol{q}) = \nabla\cdot\boldsymbol{F}_v(\boldsymbol{q},\nabla\boldsymbol{q}), \qquad (1)$$

where $\boldsymbol{F}_c = (\boldsymbol{f}_c, \boldsymbol{g}_c)$ and $\boldsymbol{F}_v = (\boldsymbol{f}_v, \boldsymbol{g}_v)$ are the inviscid and viscous flux vectors respectively, given by

$$\boldsymbol{f}_c = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uH \end{bmatrix}, \qquad \boldsymbol{g}_c = \begin{bmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho vH \end{bmatrix},$$

$$\boldsymbol{f}_v = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{yx} \\ u\tau_{xx} + v\tau_{yx} - q_x \end{bmatrix}, \qquad \boldsymbol{g}_v = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ u\tau_{xy} + v\tau_{yy} - q_y \end{bmatrix},$$

ρ is the fluid density, u and v are the components of the velocity $\boldsymbol{v}$, p is the pressure and E is the total internal energy for unit mass. The total enthalpy for unit mass, H, is given by $H = E + p/\rho$. The shear stress tensor components τ_{ij} and the heat flux vector components q_k of viscous flux vectors can be calculated as

$$\tau_{xx} = \left(2\mu\frac{\partial u}{\partial x} + \lambda\nabla\cdot\boldsymbol{v}\right),\ \tau_{yy} = \left(2\mu\frac{\partial v}{\partial y} + \lambda\nabla\cdot\boldsymbol{v}\right),$$

$$\tau_{xy} = \tau_{yx} = \mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right),$$

$$q_x = -k\frac{\partial T}{\partial x},\ q_y = -k\frac{\partial T}{\partial y}\ .$$

The transformation matrix $\boldsymbol{\Gamma}$ from conservative to primitive variables is given by

$$\boldsymbol{\Gamma} = \begin{bmatrix} \rho_p & 0 & 0 & \rho_T \\ \rho_p u & \rho & 0 & \rho_T u \\ \rho_p v & 0 & \rho & \rho_T v \\ \rho_p H - 1 & \rho u & \rho v & \rho_T H + \rho C_p \end{bmatrix}, \qquad (2)$$

with

$$\rho_p = \frac{\partial\rho}{\partial p}\bigg|_{T=const.}, \qquad \rho_T = \frac{\partial\rho}{\partial T}\bigg|_{p=const.}.$$

In order to close the system of equations, the Navier-Stokes equations must be augmented by algebraic expressions which relate the internal energy E, the pressure p, the dynamic viscosity μ, the second viscosity coefficient λ and the conductivity coefficient k to the thermodynamic state of the fluid. For an ideal gas, assuming that the fluid satisfies the equation of state of perfect gas, the pressure is given by $p = \rho(\gamma - 1)[E - \frac{u^2+v^2}{2}]$, where γ is the ratio of specific heats of the fluid, given by $\gamma = \frac{C_p}{C_v}$. The dynamic viscosity coefficient μ can be approximated using the Sutherland formula $\frac{\mu}{\mu_0} = \frac{T_0+C}{T+C}\left(\frac{T}{T_0}\right)^{\frac{3}{2}}$, with T_0, μ_0 and C depending on the gas. Finally we have $\rho_p = \frac{1}{T}$ and $\rho_T = -\frac{\rho}{T}$.

3. DG DISCRETIZATION

The BR2 scheme [9] for the DG discretization of the Navier-Stokes equations can be obtained introducing the auxiliary variable $\boldsymbol{z} = \nabla\boldsymbol{q}$ and reformulating Eq. (1) as a system of two first order equations,

$$\boldsymbol{z} = \nabla\boldsymbol{q},$$
$$\boldsymbol{\Gamma}\frac{\partial \boldsymbol{q}}{\partial t} + \nabla\cdot\boldsymbol{F}_c(\boldsymbol{q}) = \nabla\cdot\boldsymbol{F}_v(\boldsymbol{q},\boldsymbol{z}),$$

equipped with suitable initial and boundary conditions. To discretize in space, we define $\boldsymbol{V}_h$ to be the space of discontinuous vector-valued polynomials of degree n, on a subdivision τ_h of the domain Ω into non-overlapping elements such that $\Omega = \bigcup_{K\in\tau_h} K$. Thus, the solution and test functions space is defined by

$$\boldsymbol{V}_h = \{\boldsymbol{v_h} \in L^2(\Omega_h) \colon \boldsymbol{v_h}|_K \in P_n(K)\ \forall K \in \tau_h\},$$

where $P_n(K)$ is the space of polynomial functions of degree at most n. By approximating the unknown $\boldsymbol{z}$ and $\boldsymbol{q}$ with the discrete versions $\boldsymbol{z}_h \in \boldsymbol{V}_h$ and $\boldsymbol{q}_h \in \boldsymbol{V}_h$, considering the discrete test functions $\boldsymbol{u}_h \in \boldsymbol{V}_h$ and $\boldsymbol{v}_h \in \boldsymbol{V}_h$, and finally applying the techniques described in [9], it is possible to express the auxiliary variable $\boldsymbol{z}_h$ as sum of two contributions:

$$\boldsymbol{z}_h|_{\Omega_h} = \nabla\boldsymbol{q}_h + \boldsymbol{R}([\![\boldsymbol{q}_h]\!]_0),$$
$$\boldsymbol{z}_h|_{\Gamma_h} = \nabla\boldsymbol{q}_h + \eta_e\boldsymbol{R}_e([\![\boldsymbol{q}_h]\!]),$$

where Ω_h is the discrete approximation of the domain, $\partial\Omega_h$ the boundary, Γ_h^0 the set of internal edges and $\Gamma_h = \Gamma_h^0 \cup \partial\Omega_h$. $\boldsymbol{R}_e([\![\boldsymbol{q}_h]\!])$ and $\boldsymbol{R}([\![\boldsymbol{q}_h]\!]_0)$ are, respectively, the local and global lifting operators, defined as

$$\int_{\Omega_h} \boldsymbol{u}_h{}^T \cdot \boldsymbol{R}_e([\![\boldsymbol{q}_h]\!])dx = -\int_{\mathrm{E}} \{\boldsymbol{u}_h{}^T\} \cdot [\![\boldsymbol{q}_h]\!]_0 d\sigma,$$

$$\boldsymbol{R}([\![\boldsymbol{q}_h]\!]_0) = \sum_{\mathrm{E}} \boldsymbol{R}_e([\![\boldsymbol{q}_h]\!]),$$

with

$$[\![\boldsymbol{q}_h]\!]_0 = \begin{cases} [\![\boldsymbol{q}_h]\!] & on\ \Gamma_h^0 \\ (\boldsymbol{q}_h - \boldsymbol{q}_b) & on\ \partial\Omega_h \end{cases},$$

$$[\![\boldsymbol{q}_h]\!] = (\boldsymbol{q}_h\, \boldsymbol{n})^+ + (\boldsymbol{q}_h\, \boldsymbol{n})^-,$$

and $\{(\cdot)\}$ denoting the average between left $(\cdot)^-$ and right state $(\cdot)^+$, see Fig. 1.

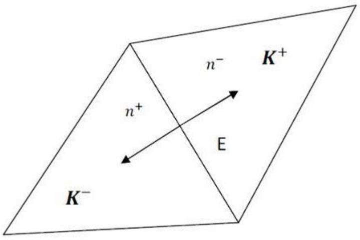

Fig. 1 Two neighboring elements K^+ and K^- sharing edge E.

Each local lifting operator is not null only on the one or two element $K^+$and K^- adjacent to the generic edge E. η_e is called "penalty" parameter and its lower bound is estabilished as the number of neighbours of the generic element K, to guarantee the stability of the method.
In this way it is possible to evaluate $\boldsymbol{F}_v(\boldsymbol{q}_h, \boldsymbol{z}_h)$ as

$$\boldsymbol{F}_v(\boldsymbol{q}_h, \boldsymbol{z}_h)|_{\Omega_h} = \boldsymbol{F}_v(\nabla\boldsymbol{q}_h + \boldsymbol{R}([\![\boldsymbol{q}_h]\!]_0)),$$

$$\boldsymbol{F}_v(\boldsymbol{q}_h, \boldsymbol{z}_h)|_{\Gamma_h} = \boldsymbol{F}_v(\nabla\boldsymbol{q}_h + \eta_e\boldsymbol{R}_e([\![\boldsymbol{q}_h]\!])).$$

Therefore, the BR2 formulation of the system (1) is given by,

$$\begin{aligned} &\int_{\Omega_h} \boldsymbol{v}_h{}^T \boldsymbol{\Gamma} \frac{\partial \boldsymbol{q}_h}{\partial t} dx + \\ &-\int_{\Omega_h} (\nabla\boldsymbol{v}_h{}^T) \cdot (\boldsymbol{F}_c(\boldsymbol{q}_h) - \boldsymbol{F}_v(\boldsymbol{q}_h, \nabla\boldsymbol{q}_h + \boldsymbol{R}([\![\boldsymbol{q}_h]\!]_0)))\, dx + \\ &+\int_{\Gamma_h^0} \boldsymbol{v}_h{}^{+T} \boldsymbol{H}(\boldsymbol{q}_h{}^+, \boldsymbol{q}_h{}^-, \boldsymbol{n}) d\sigma + \\ &-\int_{\Gamma_h^0} [\![\boldsymbol{v}_h]\!] \{\boldsymbol{F}_v(\boldsymbol{q}_h, \nabla\boldsymbol{q}_h + \eta_e\boldsymbol{R}_e([\![\boldsymbol{q}_h]\!]))\} d\sigma + \\ &+\int_{\partial\Omega_h} \boldsymbol{v}_h{}^{+T} (\boldsymbol{H}(\boldsymbol{q}_h{}^+, \boldsymbol{q}_h{}^b, \boldsymbol{n}) \\ &\quad - \boldsymbol{F}_v(\boldsymbol{q}_h\,, \nabla\boldsymbol{q}_h + \eta_e\boldsymbol{R}_e([\![\boldsymbol{q}_h]\!]_0), \boldsymbol{n})_b) d\sigma \\ &= 0 \end{aligned} \tag{4}$$

where $\boldsymbol{H}(\boldsymbol{q}_h{}^+, \boldsymbol{q}_h{}^-, \boldsymbol{n})$ and $\boldsymbol{H}(\boldsymbol{q}_h{}^+, \boldsymbol{q}_h{}^b, \boldsymbol{n})$ are the numerical flux functions at the interior and boundary faces, respectively, and $\boldsymbol{n}$ is the unit outward normal vector.

We underline that Γ is only the transformation matrix from conservative to primitive variables and it isn't involved in the preconditioning. In this way the numerical scheme maintains the temporal consistency and can be used for non-stationary problems.

The spatial DG discretization of Eq. (4) results in the following global system of equations:

$$\mathbf{M}_\Gamma \frac{\mathrm{dQ}}{\mathrm{dt}} + \mathbf{R} = 0, \tag{5}$$

where $\mathbf{Q}$ and $\mathbf{R}$ are the global vectors of degrees of freedom (dofs) and residuals, respectively, and $\mathbf{M}_\Gamma$ stands for the discretization of the first integral of Eq. (4).

3.1 Boundary Conditions

In the case of the Navier-Stokes equations boundary conditions must be imposed on the boundary data and on their gradients, for the computations of viscous terms. At far-field we employ the non-preconditioned boundary condition [12], because the flux preconditioning technique does not alter the time dependence of the equations. At the wall boundary we impose the no slip condition (on velocity components) plus the adiabatic condition (on heat flux). We underline that, as shown by Bassi and Rebay [13], high order DG methods are highly sensitive to geometry representation. Thus it is necessary to adopt a high order representation of the domain boundary. To this aim, in this work, the geometry is represented using a nodal Lagrange basis according to [14].

3.2 Numerical Flux Function

In this work we employ the Roe's approximate Riemann solver [15] to evaluate the numerical flux function at the interface between elements.

The standard formulation of the numerical flux, for the primitive variables $\boldsymbol{q}$ is

$$\begin{aligned} \boldsymbol{H}(\boldsymbol{q}^+, \boldsymbol{q}^-, \boldsymbol{n}) = \frac{1}{2}(&\boldsymbol{F}_c(\boldsymbol{q}^+) \cdot \boldsymbol{n} + \boldsymbol{F}_c(\boldsymbol{q}^-) \cdot \boldsymbol{n} \\ &-\boldsymbol{D}(\boldsymbol{q}^+, \boldsymbol{q}^-, \boldsymbol{n})\Delta\boldsymbol{q}), \end{aligned}$$

where the dissipative term $\boldsymbol{D}(\boldsymbol{q}^+, \boldsymbol{q}^-, \boldsymbol{n})$ can be evaluated as,

$$\boldsymbol{D}(\boldsymbol{q}^+, \boldsymbol{q}^-, \boldsymbol{n}) = \boldsymbol{\Gamma}(\boldsymbol{T}|\boldsymbol{\Lambda}|\boldsymbol{T}^{-1}) = \boldsymbol{\Gamma}|\boldsymbol{\Gamma}^{-1}\left(\frac{\partial \boldsymbol{F}_c}{\partial \boldsymbol{q}} \cdot \boldsymbol{n}\right)|.$$

$\boldsymbol{\Lambda} = diag\{\boldsymbol{v} \cdot \boldsymbol{n}, \boldsymbol{v} \cdot \boldsymbol{n}, \boldsymbol{v} \cdot \boldsymbol{n} + c, \boldsymbol{v} \cdot \boldsymbol{n} - c\}$ is the diagonal matrix of eigenvalues and $\boldsymbol{T}$ is the modal matrix that diagonalizes $\boldsymbol{\Gamma}^{-1}\left(\frac{\partial \boldsymbol{F}_c}{\partial \boldsymbol{q}} \cdot \boldsymbol{n}\right)$. The dissipation matrix is computed using Roe's average [16].

It was demonstrated in [2] that at low Mach number this numerical flux function produces an incorrect physical behaviour. In particular, the pressure fluctuations, that in the continuous case scale with squared Mach number, in the discrete case scale in order of Mach. To extend the validity of the Roe's numerical flux function at the incompressible limit, we use the low Mach number preconditioning technique originally proposed by Weiss & Smith in [7] for FV schemes, and latter extended to DG scheme, for inviscid flows by the authors [8]. To this purpose the transformation matrix $\mathbf{\Gamma}$ from conservative to primitive variables is replaced by the preconditioning matrix $\bar{\mathbf{\Gamma}}$ as follows

$$\boldsymbol{D}(\boldsymbol{q}^+,\boldsymbol{q}^-,\boldsymbol{n}) = \bar{\mathbf{\Gamma}}|\,\bar{\mathbf{\Gamma}}^{-1}\left(\frac{\partial \boldsymbol{F}_c}{\partial \boldsymbol{q}}\cdot \boldsymbol{n}\right)|.$$

The Weiss and Smith preconditioning matrix $\bar{\mathbf{\Gamma}}$ is obtained from (2), replacing ρ_p with θ:

$$\bar{\mathbf{\Gamma}} = \begin{bmatrix} \theta & 0 & 0 & \rho_T \\ \theta u & \rho & 0 & \rho_T u \\ \theta v & 0 & \rho & \rho_T v \\ \theta H - 1 & \rho u & \rho v & \rho_T H + \rho C_p \end{bmatrix},$$

where θ is given by

$$\theta = \left(\frac{1}{U_r^2} - \frac{\rho_T}{\rho C_p}\right).$$

Here, U_r is a reference velocity which, for an ideal gas, is defined as

$$U_r = \begin{cases} \varepsilon c & if\ |\boldsymbol{v}\cdot\boldsymbol{n}| < \varepsilon c \\ |\boldsymbol{v}\cdot\boldsymbol{n}| & if\ \ \varepsilon c < |\boldsymbol{v}\cdot\boldsymbol{n}| < c, \\ c & if\ |\boldsymbol{v}\cdot\boldsymbol{n}| > c \end{cases}$$

c is the acustic speed and ε is a small number included to prevent singularities at stagnation points. ε is chosen as $O(M)$ according to [8].

Weiss and Smith proposed for viscous computation to limit the reference velocity such that it does not become smaller than the local diffusion velocity [7]. Colin Y et al. proposed a more accurate limitation based on isentropic Mach number [17]. Nevertheless, DG computations showed that it is better not introducing the viscous limitation for aerodynamic simulations, like observed by Unrau and Zingg [18] for the finite volume discretization.

4. TIME DISCRETIZATION OF NAVIER STOKES EQUATIONS

As demonstrated in [10] by Birken and Meister the low Mach number preconditioning approach combined with an explicit time integration scheme suffers from stability problems if the time step does not satisfy the requirement to be $O(M^2)$ as the Mach number tends to zero. In order to overcome this severe disadvantage implicit method are usually employed. In this work we use the implicit backward Euler scheme:

$$\left[\frac{\mathbf{M}_\Gamma}{\Delta t} + \frac{\partial \mathbf{R}^\mathrm{n}}{\partial \mathbf{Q}}\right]\Delta\mathbf{Q}^\mathrm{n} = -\mathbf{R}^\mathrm{n}, \qquad (6)$$

where $\Delta\mathbf{Q}^\mathrm{n} = \mathbf{Q}^{\mathrm{n}+1} - \mathbf{Q}^\mathrm{n}$, $\frac{\partial \mathbf{R}^\mathrm{n}}{\partial \mathbf{Q}}$ is the Jacobian matrix of the DG space discretization and $\left[\frac{\mathbf{M}_\Gamma}{\Delta \mathrm{t}} + \frac{\partial \mathbf{R}^\mathrm{n}}{\partial \mathbf{Q}}\right]$ denotes the global system matrix. The matrix $\left[\frac{\mathbf{M}_\Gamma}{\Delta \mathrm{t}} + \frac{\partial \mathbf{R}^\mathrm{n}}{\partial \mathbf{Q}}\right]$ can be regarded as an $N_K \times N_K$ block sparse matrix where N_K is the number of elements in τ_h and the rank of each block is $m \times N_{dof}^K$, where N_{dof}^K is the number of dof for each of the m primitive variables in the generic element K. The Jacobian matrix of the DG discretization has been computed analytically (except for the computation of the dissipative part of the numerical flux that has been computed numerically) without any approximation and, using very large time steps, the method can therefore achieve quadratic convergence in the computation of steady state solutions. For the backward Euler scheme and in the limit $\Delta t \to \infty$ Eq. (6) is in fact identical to one iteration of the Newton method applied to the steady discrete problem. To solve Eq. (6) we have used the restarted GMRES algotithm with ILU(0) available in the PETSc [19] library.

5. NUMERICAL RESULTS

In order to demonstrate the capability of the DG method in solving laminar low Mach number flows we consider the flow past a NACA0012 airfoil at zero angle of attack and at a Reynolds number = 500. The simulations have been performed with and without the flux preconditioning approach at several Mach numbers, $M = 10^{-1}$, $M = 10^{-2}$ and $M = 10^{-3}$, using linear (P_1), quadratic (P_2) and cubic elements (P_3). Two grid topologies (quadrangular and triangular) are used in order to investigate the behaviour of both the standard and the preconditioned DG method for different element shapes. The C-type quadrangular mesh contains 1792 elements. The triangular mesh contains 3584 elements resulting from the triangulation of the quadrangular one (Fig. 2).

The computational results are organized in two sections, one focusing on the convergence of the residuals and the other on the accuracy of the converged solutions. As regards the convergence of the residuals we present plots of the normalized L_2 norm of residuals versus number of iterations and versus CPU time. As regards the accuracy of the converged solutions, the normalized pressure fields are presented for a qualitative comparison.

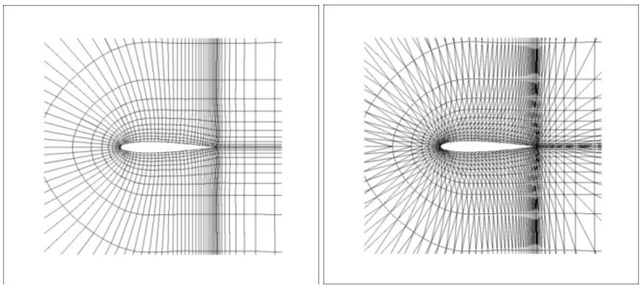

Fig. 2 Computational Grids.

5.1 Convergence

The convergence histories are shown only for the quadrangular grid as similar histories are obtained on the triangular one. The results are presented for fixed GMRES parameters (number of Krylov subspace vectors = 120, number of restarts = 1 and relative tolerance to stop iterative solution $= 10^{-6}$). Fig 3 compares the history of residuals versus the number of implicit iteration steps of the backward

Euler scheme with and without flux preconditioning. The plots show a deterioration in the convergence rate without preconditioning as the Mach number gets smaller whilst the preconditioned scheme always produces quadratic convergence. The effect is appreciable at $M = 10^{-2}$ and more evident at $M = 10^{-3}$. At $M = 10^{-1}$ both the non-preconditioned and preconditioned DG schemes converge at about the same convergence rate independently of the polynomial degree. Furthermore the corresponding non-preconditioned and preconditioned residual histories decrease about the same order of magnitude. In particular, the residual of velocity components are indistinguishable, whereas the non-preconditioned residuals of pressure and temperature present a slightly larger decrease than the corresponding preconditioned one. This behaviour becomes more evident at $M = 10^{-2}$, and at $M = 10^{-3}$ for linear elements. The major benefits of preconditioning technique are shown at $M = 10^{-3}$, for quadratic and cubic elements, where the convergence of the numerical solution is not reached without preconditioning. We conclude that, with the chosen GMRES parameters, the flux preconditioning improves the convergence rate of the numerical solution. This is due to the effect of low Mach number preconditioning on the linear system matrix, through the Jacobian of the residuals. In particular the full convergence of the residuals is reached quadratically in about 10 iterations, independently of both Mach number and polynomial degree, like for the inviscid case [8].

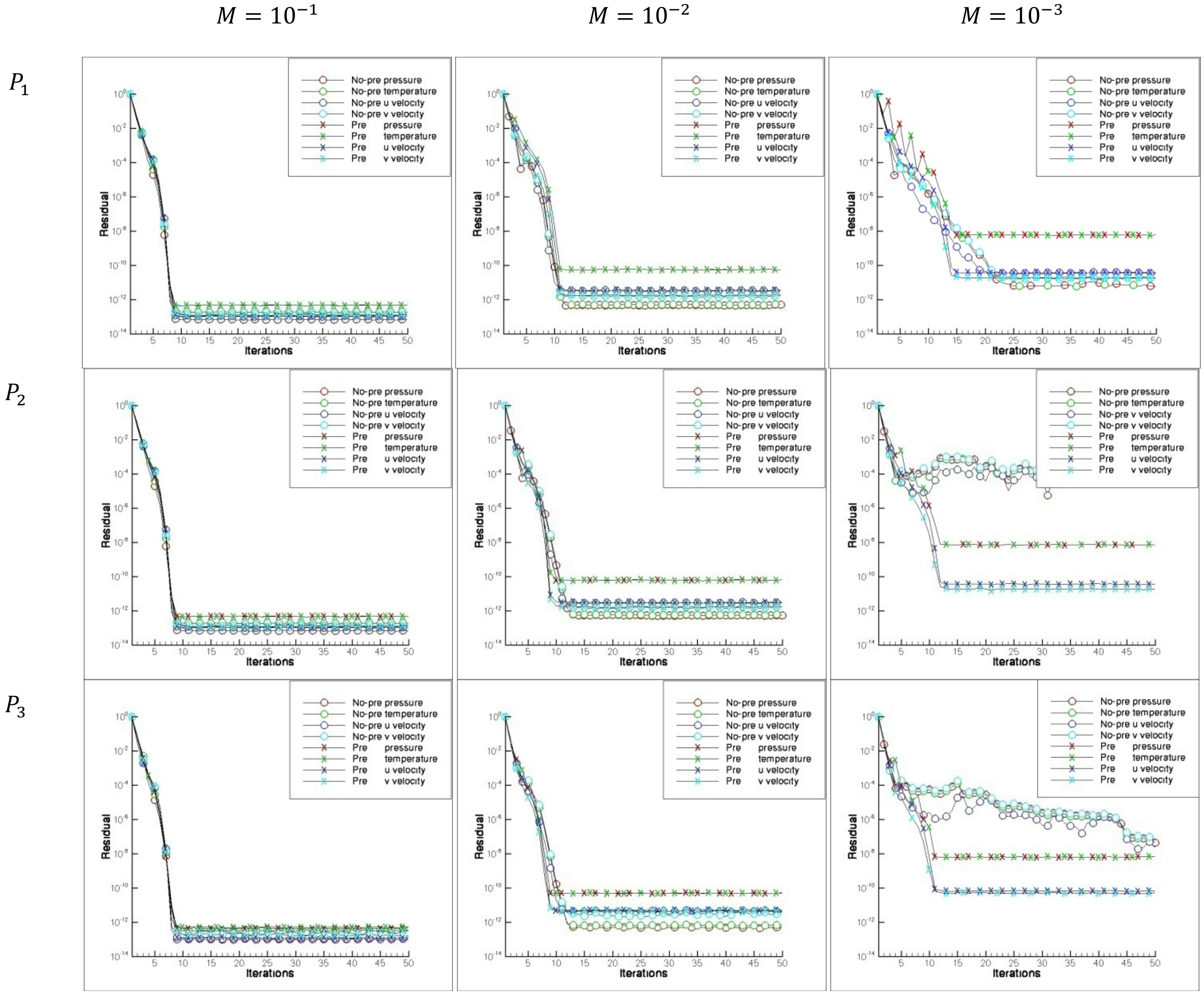

Fig. 3 History of residuals versus number of iterations for the quadrangular grid. $M = 10^{-1}$(left column), $M = 10^{-2}$ (middle column) and $M = 10^{-3}$ (right column). Linear (P_1 top row), quadratic (P_2 middle row) and cubic (P_3 bottom row) elements.

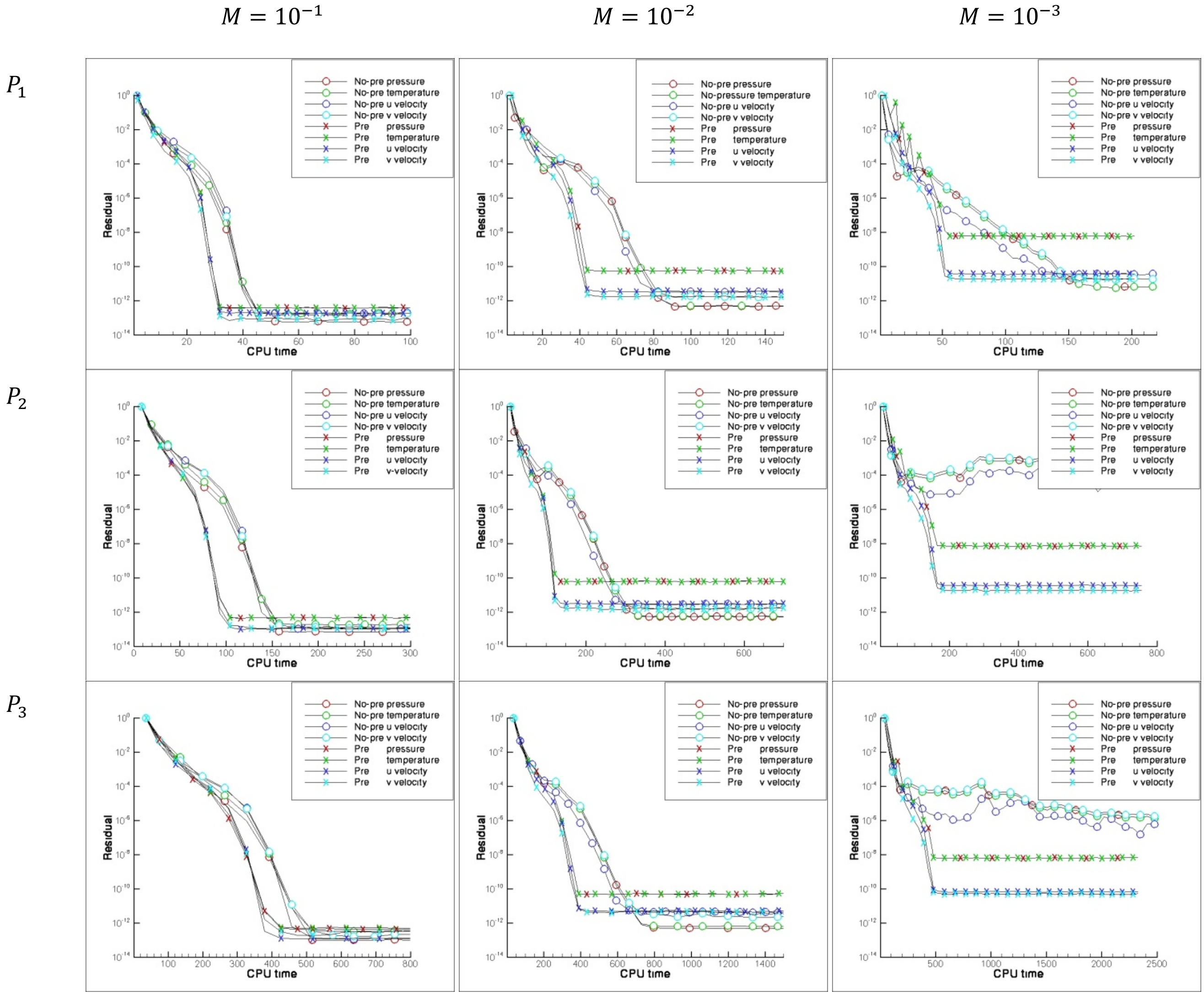

Fig. 4 History of residuals versus CPU time for the quadrangular grid. $M = 10^{-1}$ (left column), $M = 10^{-2}$ (middle column) and $M = 10^{-3}$ (right column). Linear (P_1 top row), quadratic (P_2 middle row) and cubic (P_3 bottom row) elements.

Fig. 4 compares the history of residuals versus CPU time (seconds), computed on the quadrangular grid with and without flux preconditioning. The plots show that, using the non-preconditioned Roe's flux, the overhead in terms of CPU time increases as the Mach number gets smaller and the polynomial degree increases, whilst it is almost independent of the Mach number with flux preconditioning. We conclude that the preconditioning improves the efficiency of the implicit solver, reducing the overhead in terms of CPU time.

Fig.5 summarizes the performance of the GMRES solver with (right column) and without (left column) low Mach number preconditioning. The graphs show the results for the P_1, P_2 and P_3 solutions at $M = 10^{-2}$. Similar results hold also for $M = 10^{-1}$ and $M = 10^{-3}$. The plots on the top row show the number of GMRES iterations (open symbols) and the logarithm of CFL number (solid symbols), while those on the bottom row show the ratio between the L_2 norms of the last and the first residual of the GMRES iterative solution. The quantity on the x-axis is the number of non-linear iterations. We can observe that increasing the CFL number the computations performed without low Mach number preconditioning rapidly use up the maximum number of GMRES iterations (240) without satisfying the required six-order drop of residuals. Instead the low Mach number preconditioning improves the efficiency of GMRES solver so that the preconditioned solutions require somewhat less than 240 GMRES iterations to solve the linear system within each time step, even for the highest CFL numbers: this can be explained with the improved conditioning of the linear system matrix. We notice that in comparison to the inviscid case [8] a higher number of Krylov subspace vectors was used (120 instead of 60), with the same number of restarts (1) and relative tolerance (10^{-6}). Finally we mention that, at $M = 10^{-2}$, the cost to compute the analytical Jacobian relative to the computational cost of a full time step using 240 GMRES iterations is around 31%, 41% and 60% for the P_1, P_2 and P_3 solutions, respectively.

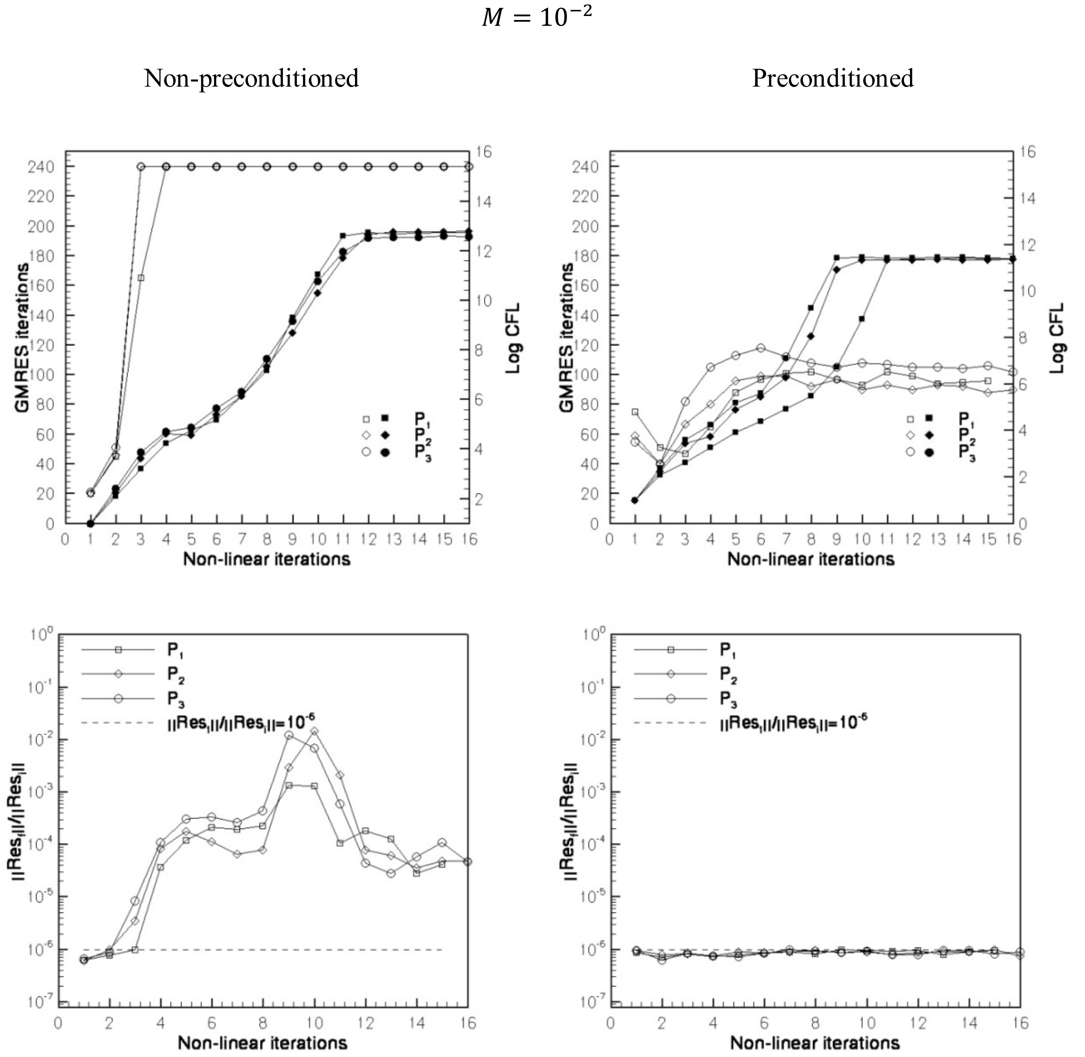

Fig. 5 Behaviour of GMRES solver with (right column) and without (left column) low Mach number preconditioning for $M = 10^{-2}$.

5.2 Accuracy

In the following we present the contour plots of the normalized pressure, defined as $p_{norm} = \frac{p - p_{\min}}{p_{\max} - p_{\min}}$, computed on the quadrangular and triangular grids. Fig. 6 shows the isolines of normalized pressure at $M = 10^{-2}$. We see that the preconditioned solution is more accurate than the non-preconditioned one. The loss of accuracy of the non-preconditioned solution is less evident increasing the polynomial degree. In particular we obtain an acceptable level of accuracy without preconditioning only using P_3 elements. Furthermore, for a given Mach number, the higher the polynomial degree, the lower is the difference between the preconditioned and the non-preconditioned solutions. In such cases the preconditioning allows to significantly reduce the computational effort. We note that for a given polynomial degree the non-preconditioned solution worses as the Mach number reduces.

In Fig. 7 we compare the contours of normalized pressure without (top row) and with (middle row) preconditioning, near the leading edge, computed on a quadrangular grid, at $M = 10^{-3}$, for linear P_1 (left column), quadratic P_2 (central column), and cubic P_3 (right column) elements. The non-preconditioned contours at $M = 0.4$ (bottom row) are also shown as reference. It is evident that approaching the stagnation point the solution degrades and that this effect reduces by increasing the degree of polynomial approximation.

Fig. 8 shows the same local analysis of Fig. 7, for the triangular grid. Overall, it is worth noting that also in the case of the Navier-Stokes equations, the DG discretization on triangular grid yields remarkably accurate solutions at low Mach number even without preconditioning, like found for the inviscid case [8]. In particular, the preconditioned and the non-preconditioned contours of normalized pressure are almost indistinguishable using P_3 elements, whereas some small differences can be seen in the P_1 and P_2 solutions. The marked influence of the geometrical shape of the elements on the accuracy of the Roe's flux in the low Mach number limit could be explained by the asymptotic analysis recently performed by Rieper and Bader [20,21] for the first-order Roe scheme. Our current work seems to indicate that low order DG schemes face the same problems as the standard finite volume upwind schemes: at low Mach number they only work on triangular elements.

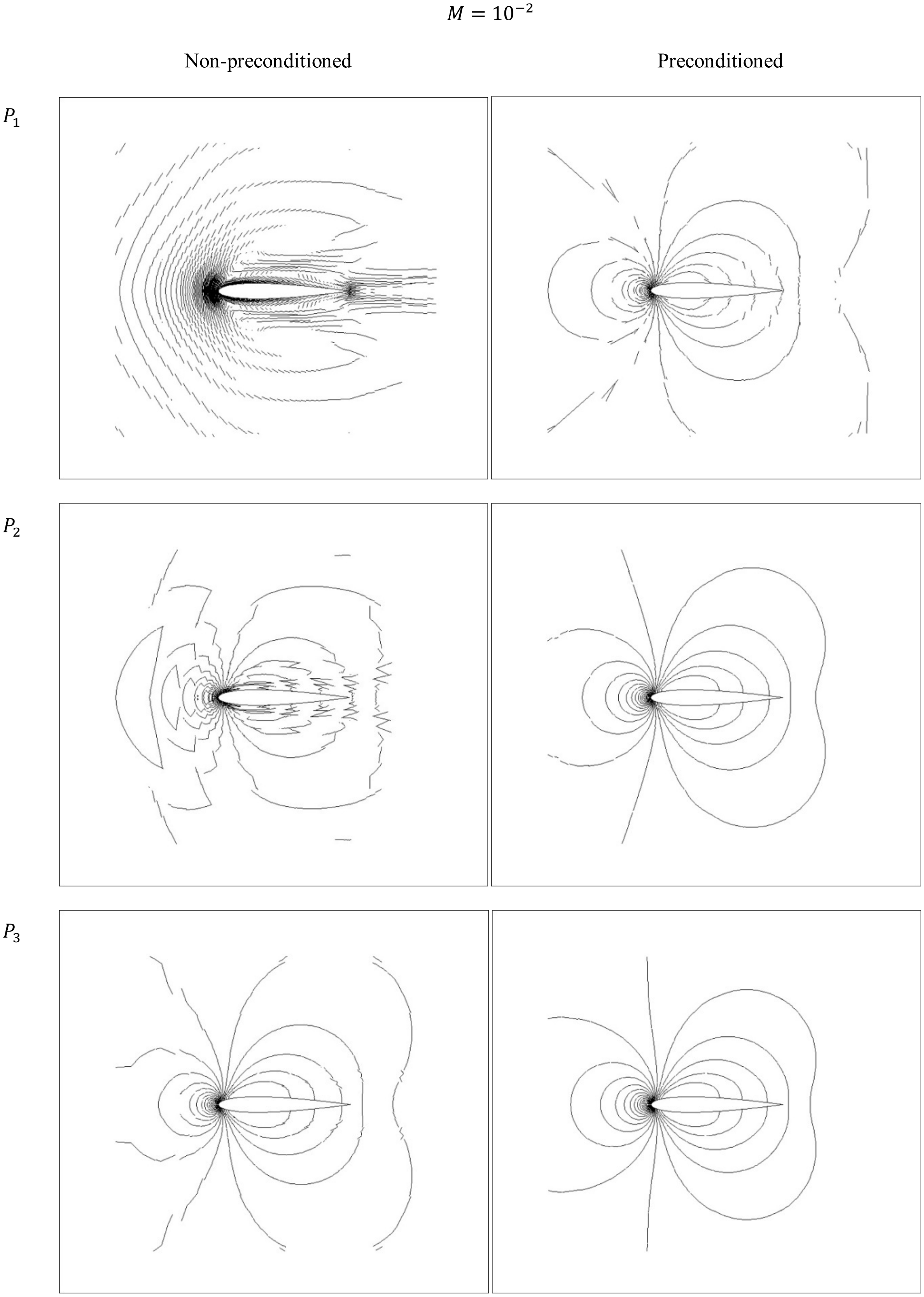

Fig. 6 Contours of normalized pressure with and without preconditioning for quadrangular grid. $M = 10^{-2}$. Linear (P_1 top row), quadratic,(P_2 middle row) and cubic (P_3 bottom row) elements.

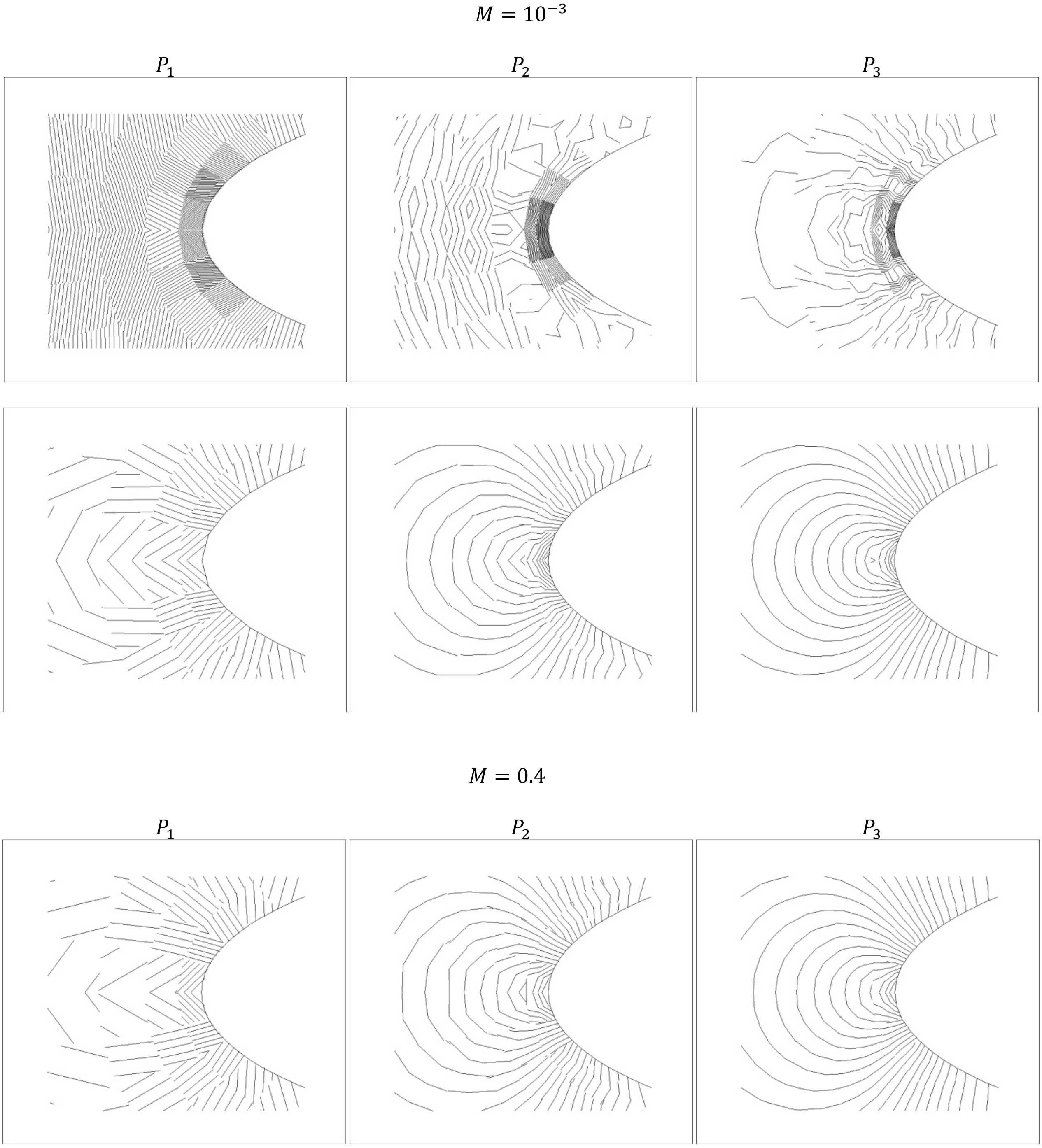

Fig. 7 Quadrangular grid: contours of normalized pressure near the leading edge for $M = 10^{-3}$, non-preconditioned (top row), preconditioned (middle row) and non-preconditioned at $M = 0.4$ (bottom row). Linear P_1 (left column), quadratic P_2 (central column), and cubic P_3 (right column) elements.

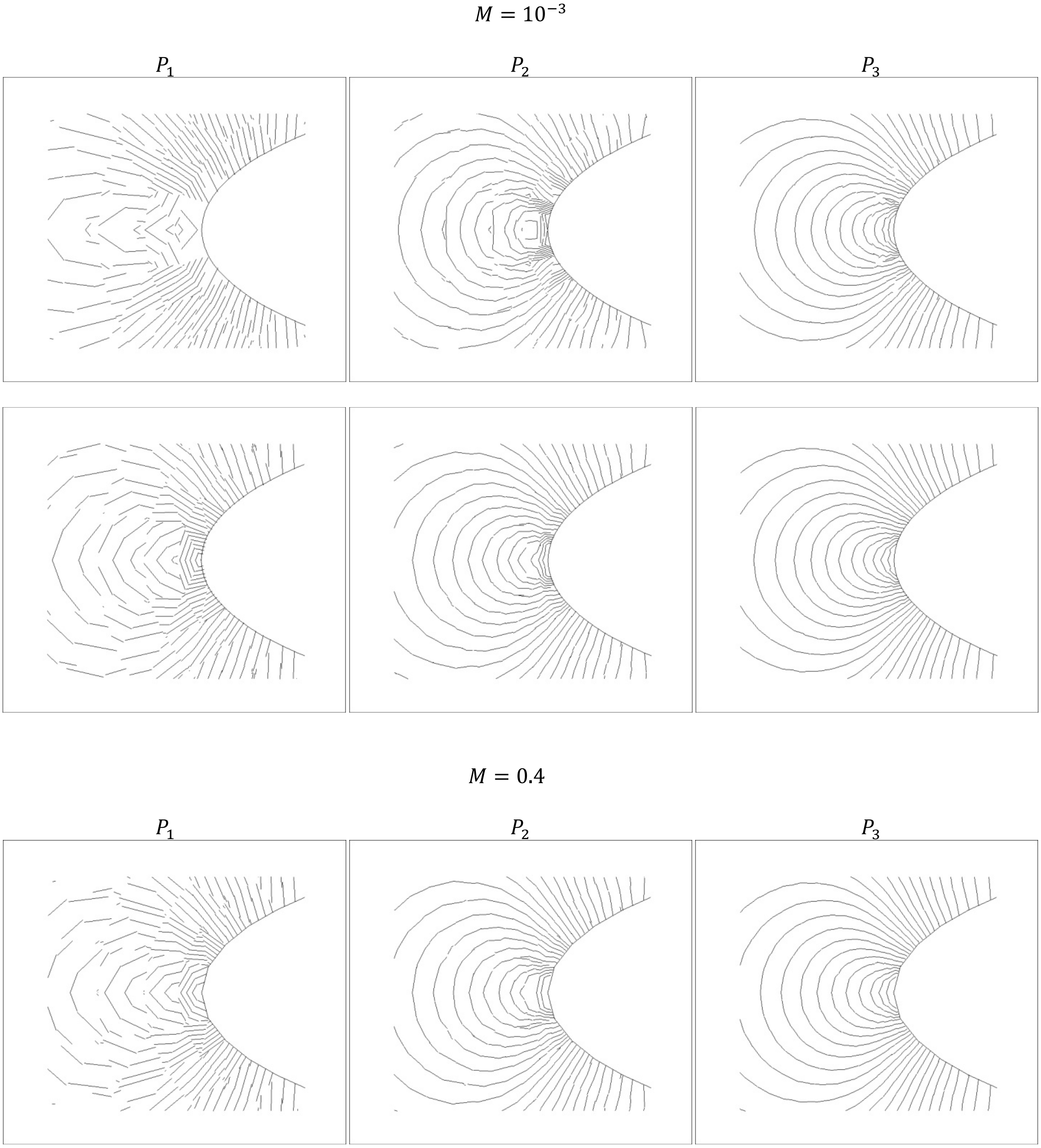

Fig. 8 Triangular mesh: contours of normalized pressure near the leading edge for $M = 10^{-3}$, non-preconditioned (top row), preconditioned (middle row) and non-preconditioned at $M = 0.4$ (bottom row). Linear P_1 (left column), quadratic P_2 (central column), and cubic P_3 (right column) elements.

6. CONCLUSIONS

We have presented the application of a preconditioned DG discretization and a fully implicit time integration method to the numerical solution of the Navier-Stokes equations at low Mach numbers. The algorithm employs the flux preconditioning approach which modifies only the dissipative terms of the numerical flux and thus preserves the time consistency of the governing equations. The performance of the method is demonstrated by considering a flow past a NACA0012 airfoil at different low Mach numbers. The simulations have been performed using piecewise linear, quadratic and cubic elements (corresponding to second-, third-order and fourth-order accurate space discretizations) on quadrangular and triangular grids. The system of equations has been iterated to the steady-state using the backward Euler scheme. In all the tests the implicit scheme turns out to be efficient using just the modified numerical flux function. Furthermore, as for the inviscid case, the laminar computations show that the flux preconditioning approach is mandatory to obtain accurate solutions on a relatively coarse quadrangular grid. In contrast to that, the DG discretization on the triangular grid yields remarkably accurate solutions even without preconditioning.

NOMENCLATURE

c acustic speed [-]
C Sutherland's constant [-]
$\mathrm{C_p}$ specific heat at constant pressure [-]
$\mathrm{C_v}$ specific heat at constant volume [-]
$\boldsymbol{D}$ dissipation matrix [-]
E generic edge of the element [-]
E total internal energy for unit mass [-]
$\boldsymbol{f}_c$ x-component of inviscid flux vector [-]
$\boldsymbol{F}_c$ inviscid flux vector [-]
$\boldsymbol{f}_v$ x-component of viscous flux vector [-]
$\boldsymbol{F}_v$ viscous flux vector [-]
$\boldsymbol{g}_c$ y-component of inviscid flux vector [-]
$\boldsymbol{g}_v$ y-component of viscous flux vector [-]
H total enthalpy for unit mass [-]
$\boldsymbol{H}$ numerical flux function vector [-]
k conductivity coefficient [-]
K generic element of the domain [-]
m number of equations (m = 4 in 2d) [-]
M Mach number [-]
$\mathbf{M}_\Gamma$ mass matrix including $\boldsymbol{\Gamma}$ [-]
n time level [-]
n degree of polynomial [-]
$\boldsymbol{n}$ unit outward normal vector [-]
N_K number of elements in τ_h [-]
N_{dof}^K number of dofs per equation and element [-]
p pressure [-]
$p_{\min}$ minimum pressure [-]
$p_{\max}$ maximum pressure [-]
p_{norm} normalized pressure [-]
P_n space of polynomial functions of degree at most n [-]
q_k heat flux vector component [-]
$\boldsymbol{q}$ primitive variables vector [-]
$\boldsymbol{q}_h$ discrete version of the primitive variables vector [-]
$\mathbf{Q}$ global vector of dofs [-]
$\boldsymbol{R}$ global lifting operator [-]
$\mathbf{R}$ global vector of residuals [-]
$\boldsymbol{R}_e$ local lifting operator [-]
t time [-]
T temperature [-]
$\mathrm{T_0}$ reference temperature [-]
$\boldsymbol{T}$ modal matrix [-]
u x-velocity component [-]
$\boldsymbol{u}_h$ discrete test functions vector [-]
U_r reference velocity [-]
v y-velocity component [-]
$\boldsymbol{v}$ velocity vector [-]
$\boldsymbol{v}_h$ discrete test functions vector [-]
$\boldsymbol{V}_h$ space of discontinuous vector-valued polynomials of degree n [-]
$\boldsymbol{z}$ auxiliar variables vector [-]
$\boldsymbol{z}_h$ discrete version of the auxiliar variables vector [-]

γ ratio of specific heats of the fluid [-]
Γ_h^0 set of internal edges [-]
Γ_h set of internal and boundary edges [-]
$\boldsymbol{\Gamma}$ transformation matrix from conservative to primitive variables [-]
$\bar{\boldsymbol{\Gamma}}$ preconditioning matrix [-]
ε preconditioning parameter [-]
η_e penalty parameter [-]
θ preconditioning parameter [-]
λ second viscosity coefficient [-]
$\boldsymbol{\Lambda}$ diagonal matrix of eigenvalues [-]
μ dynamic viscosity [-]
μ_0 reference viscosity at reference temperature [-]
ρ density [-]
ρ_p partial derivative of density with respect to pressure at constant temperature [-]
ρ_T partial derivative of density with respect to temperature at constant pressure [-]
τ_h subdivision of the domain [-]
τ_{ij} shear stress tensor component [-]
Ω domain [-]
Ω_h discrete approximation of the domain [-]
$\partial\Omega_h$external boundary of Ω_h [-]

+ right state [-]
- left state [-]
b boundary state [-]

BIBLIOGRAPHY

1. G. Volpe Performance of compressible flow codes at low mach numbers. AIAA Journal, vol.31, pp. 49-56, 1993.
2. H. Guillard and C. Viozat On the Behavior of Upwind Schemes in the Low Mach Number Limit. Computers & Fluids, vol. 28, pp. 63-86,1999.
3. Y. H. Choi and C. L. Merkle The Application of Preconditioning in Viscous Flows. J. Computational Physics, vol. 105, pp. 207-233, 1993.
4. E. Turkel Review of Preconditioning Methods for Fluid Dynamics. Applied Numerical Mathematics, vol. 12, pp. 257-284, 1993.

5. E. Turkel Preconditioning Techniques in Computational Fluid Dynamics. Annu. Re. Fluid Mech., vol. 31, pp. 385-416, 1999.
6. D. Lee and B. Van Leer Progress in Local Preconditioning of the Euler and Navier-Stokes Equations. AIAA Paper vol. 93, pp.3328, 1993.
7. J. Weiss and W. A. Smith Preconditioning Applied to Variable and Constant Density Flows. AIAA Journal, vol.33, pp. 2050-2057, 1995.
8. F. Bassi, C. De Bartolo, R. Hartmann, A. Nigro, A discontinuous Galerkin method for inviscid low Mach number flows, J. Comput. Phys., vol. 228 (11), pp. 3996–4011, 2009.
9. F. Bassi, S. Rebay High-order accurate discontinuous Galerkin methods in computational fluid dynamics: from model problems to complex turbulent flows-Part 1 35Th CFD/Adigma Course on Very high order discretization methods, edited by H. Deconinck, 2008.
10. Birken P, Meister A. Stability of preconditioned finite volume schemes at low Mach numbers. BIT vol. 45, pp. 463–480, 2005.
11. G. Hauke, T.J.R. Hughes, A comparative study of different sets of variables for solving compressible and incompressible flows, Comput. Methods Appl. Mech. Eng., vol. 153, pp.1–44, 1998.
12. D.L. Whitfield, J.M. Janus, Three-dimensional unsteady Euler equations solution using flux vector splitting, AIAA Paper, vol. 84, pp. 1552, 1984.
13. F. Bassi, S. Rebay. High order accurate discontinuous finite element solution of the 2d Euler equations. J. Comput. Phys., vol. 138, pp.251–285, 1997.
14. A. Nigro, Discontinuous Galerkin methods for inviscid low Mach number flows, DLR-IB 124-2008/1.
15. P.L. Roe, Characteristic based schemes for the Euler equations, Annu. Rev. Fluid Mech., vol. 18, pp. 337–365, 1986.
16. P.L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, J. Comput. Phys. vol. 43, pp. 352–357, 1981.
17. Colin Y et AL A robust low speed preconditioning formulation for viscous flow computations. Comput Fluids doi:10.1016/j.compfluid.2011.01.015, 2011.
18. D. Unrau and D.W. Zingg Viscous airfoil computations using local preconditioning, AIAA Paper, vol. 96, pp. 2088, 1996.
19. S. Balay, K. Buschelman, W. D. Gropp, D. Kaushik, M. G. Knepley, L. Curfman Mclnneses, B. F. Smith, and H. Zhang. PETSc Web page http://www.mcs.anl.gov/petsc, 2001.
20. F. Rieper, G. Bader The influence of cell geometry on the accuracy of upwind schemes in the low Mach number regime, J. Comput. Phys., vol.228, pp.2918–2933, 2009.
21. F. Rieper On the dissipation mechanism of upwind-schemes in the low Mach number regime: A comparison between Roe and HLL, J. Comput. Phys. vol. 229, pp. 221–232, 2010.

SOMMARIO

In questo lavoro estendiamo il metodo di elevato ordine agli elementi finiti discontinui di Galerkin (DG) all'analisi di flussi laminari a basso numero di Mach, impiegando uno schema implicito per la discretizzazione temporale delle equazioni. L'algoritmo è basato su di una tecnica di precondizionamento che modifica esclusivamente il termine dissipativo della funzione di flusso numerico. Questa formulazione è piuttosto semplice da implementare in un qualsiasi codice DG, supera le restrizioni sul time-step tipiche degli schemi espliciti multistadio, è consistente nel tempo e quindi applicabile a flussi non stazionari. Le prestazioni dello schema sono valutate risolvendo un flusso laminare attorno ad un profilo NACA0012 con angolo d'attacco nullo e per differenti numeri di Mach, utilizzando approssimazioni polinomiali della soluzione di diverso grado. Le simulazioni sono condotte su differenti tipologie di griglia di calcolo, al fine di valutare l'influenza della discretizzazione spaziale sulla accuratezza e la convergenza delle soluzioni DG a basso numero di Mach.