GMRES Discontinuous Galerkin Solution of the Compressible Navier-Stokes Equations

F. Bassi[1] and S. Rebay[2]

[1] Dipartimento di Energetica
Università degli Studi di Ancona
Via Brecce Bianche, 60100 Ancona, Italy
[2] Diparimento di Ingegneria Meccanica
Università degli Studi di Brescia
Via Branze 38, 25123 Brescia, Italy

Abstract

We present an implicit solution method for the compressible Navier-Stokes equations based on a Discontinuous Galerkin space discretization and on the implicit backward Euler time integration scheme. The linear system arising from the implicit time stepping scheme are solved with the preconditioned GMRES iterative method. Several preconditioners have been considered. We describe the features of the method and investigate its accuracy and performance by computing several classical 2-dimensional test cases.

1 Introduction

The Discontinuous Galerkin (DG) method is a recently developed higher-order accurate method which has been receiving great attention by several researchers, among others see e.g. [10,11,2,5,6,1,3], because of its several promising features. First of all, the DG method combines two key features which characterize the finite volume and the finite element method, the physics of wave propagation being accounted for by means of (approximate) Riemann solvers at element interfaces and accuracy being pursued by means of high-order polynomial approximations within elements. The method is therefore ideally suited to compute high-order accurate solution of the Navier–Stokes equations on general structured or unstructured grids. A second interesting feature of the DG method is the compactness of the scheme. The expansion coefficients of the numerical solution associated to a generic element are in fact coupled only with those associated to neighboring elements (that is the elements sharing a face). In the case of triangular (tetrahedral) elements, this means that coupling is introduced only among the unknowns associated to four (five) elements, respectively. This compactness results in sparse matrices which are very convenient for implicit time integration schemes (especially in 3D).

In this work we present a DG method which is a variant of that described in [4,3] exploiting the linear dependence of the viscous flux function

B. Cockburn et al. (eds.), *Discontinuous Galerkin Methods*

on the gradient of the conservation variables. The equations are integrated in time with the implicit backward Euler scheme. At each time step the coupled nonlinear equations resulting from the time stepping scheme are locally linearized and therefore reduced to a system of linear equations, which is iteratively solved with the preconditioned GMRES method. As preconditioner we have considered the standard incomplete LU factorization (ILU) with different levels of fill-in and a block diagonal preconditioner which turned out to be particularly well suited to the DG Galerkin discretization of the NS equations.

In the following we give a complete description of the method, with particular attention to the discretization of the viscous part of the NS equations. The performance of the method is displayed by computing the compressible laminar flow for two classical airfoil steady state calculations.

2 DG Space Discretization of the Navier-Stokes Equations

The compressible Navier-Stokes equations can be written in compact form as

$$\frac{\partial u}{\partial t} + \nabla \cdot \boldsymbol{f}_{\mathrm{c}}(u) + \nabla \cdot \boldsymbol{f}_{\mathrm{v}}(u, \nabla u) = 0,$$

where $u \in \mathbb{R}^{d+2}$ is the vector of the conservative variables, $\boldsymbol{f}_{\mathrm{c}}$ and $\boldsymbol{f}_{\mathrm{v}} \in \mathbb{R}^{d+2} \otimes \mathbb{R}^{d}$ are the inviscid and viscous flux functions, respectively, and d denotes the number of space dimensions. The viscous flux function is linear with respect to ∇u, i.e.

$$\boldsymbol{f}_{\mathrm{v}}(u, \nabla u) = \boldsymbol{\mathcal{A}}_{\mathrm{v}}(u) \nabla u, \quad \boldsymbol{\mathcal{A}}_{\mathrm{v}}(u) = \frac{\partial \boldsymbol{f}_{\mathrm{v}}(u, \nabla u)}{\partial \nabla u} \ . \tag{1}$$

A discrete version of the NS equations is obtained by subdividing Ω into a set of elements $\{e\}$ (triangles in this work), and by restricting u and ϕ to be polynomial functions inside each element. No global continuity is enforced on u and ϕ, which are therefore discontinuous at element interfaces. By splitting the integral over Ω into the sum of integrals over the elements e and by performing an integration by parts, we obtain the weak formulation

$$\sum_e \int_{\Omega_e} \phi \frac{\partial u}{\partial t} \, d\Omega + \sum_e \oint_{\partial \Omega_e} \phi \boldsymbol{n} \cdot \boldsymbol{f}(u, \nabla u) \, d\sigma - \sum_e \int_{\Omega_e} \nabla \phi \cdot \boldsymbol{f}(u, \nabla u) \, d\Omega = 0 \ . \tag{2}$$

Due to the discontinuous function approximation chosen, the flux functions appearing in the boundary integral of the previous equation is not uniquely defined and does not disappear for internal interfaces as in the standard

continuous finite element method. It is instead necessary to resort to an interface treatment which *weakly* enforces continuity at element interfaces (and the boundary conditions at boundary sides), thus providing a coupling between neighboring elements which would be otherwise completely missing. In general this is accomplished by replacing the physical normal flux $\boldsymbol{n} \cdot \boldsymbol{f}(u, \nabla u)$ with a numerical flux $h(u^-, \nabla u^-, u^+, \nabla u^+, \boldsymbol{n}^-)$. With the notation $(\cdot)^-$ and $(\cdot)^+$ we denote the interface value of any quantity associated to the two elements sharing the face, in which the normal unit vector $\boldsymbol{n}^-$ points outward from the element associated to the values $(\cdot)^-$.

To put in evidence the different role played by the contour integral for internal interfaces and for boundary sides (and to better reflect the way in which computations are actually performed), the sum of contour integrals appearing in Eq. (2) can be rearranged as the sums of internal interface integrals and of boundary side integrals

$$-\sum_e \int_{\Omega_e} \nabla\phi \cdot \boldsymbol{f}(u, \nabla u)\, d\Omega + \sum_f \int_{\sigma_f} (\phi^- h^- + \phi^+ h^+)\, d\sigma + \sum_b \int_{\sigma_b} \phi \boldsymbol{n}^- \cdot \boldsymbol{f}(u^*, \nabla u^*)\, d\sigma, \quad (3)$$

where the subscript f refer to internal interfaces, the subscript b to boundary sides, and

$$h^\pm = h(u^\pm, \nabla u^\pm, u^\mp, \nabla u^\mp, \boldsymbol{n}^\pm) \ .$$

Notice that for each interface integral of the second sum in Eq. (3) there are two numerical flux contribution which correspond to the two elements sharing a face. The flux function arguments u^* and ∇u^* appearing in the boundary integrals of Eq. (3) are defined as

$$u^* = u^b, \qquad \nabla u^* = \nabla u^-$$

to prescribe Dirichlet conditions u^b and as

$$u^* = u^-, \qquad \boldsymbol{n} \cdot \nabla u^* = \boldsymbol{n} \cdot \nabla u^b$$

to prescribe Neumann conditions $\boldsymbol{n} \cdot \nabla u^b$.

The formulation of the numerical flux function for the inviscid part of the NS equations is completely analogous to that commonly employed in upwind finite volume methods. In our computations we have used the van Leer flux difference splitting numerical flux as modified by Hänel [12]. The inviscid terms of the NS equations are therefore discretized as

$$\sum_e \int_{\Omega_e} \phi \frac{\partial u}{\partial t}\, d\Omega - \sum_e \int_{\Omega_e} \nabla\phi \cdot \boldsymbol{f}(u, \nabla u)\, d\Omega + \sum_f \int_{\sigma_f} (\phi^- h^- + \phi^+ h^+)\, d\sigma + \sum_b \int_{\sigma_b} \phi \boldsymbol{n} \cdot \boldsymbol{f}_c(u^b)\, d\sigma = 0 \ . \quad (4)$$

The formulation of a numerical flux function for the viscous part of the Navier-Stokes equations does not have a counterpart in the finite volume method, and it therefore will be described in greater detail. In order to accommodate the generalized Laplacian operator

$$\nabla \cdot \boldsymbol{f}_{\mathrm{v}} = \nabla \cdot (\mathcal{A}_{\mathrm{v}} \nabla u)$$

in weak variational form in a space of discontinuous functions, we reformulate the NS equations as the system

$$\begin{cases} \boldsymbol{f}_{\mathrm{v}} = \mathcal{A}_{\mathrm{v}} \nabla u \\ \partial_t u + \nabla \cdot \boldsymbol{f}_{\mathrm{c}} + \nabla \cdot \boldsymbol{f}_{\mathrm{v}} = 0, \end{cases} \tag{5}$$

in which the two coupled first order equations can be approximated with DG techniques similar to those already developed in the case of hyperbolic systems of conservation laws.

Let's begin from the first equation of the system. A "weakly continuized" viscous flux $\widetilde{\boldsymbol{f}_{\mathrm{v}}}$ is defined as

$$\begin{aligned} &\sum_e \int_{\Omega_e} \psi \cdot \widetilde{\boldsymbol{f}_{\mathrm{v}}}\, d\Omega = -\sum_e \int_{\Omega_e} u\, \nabla \cdot (\mathcal{A}_{\mathrm{v}}^T \psi)\, d\Omega \\ &+ \frac{1}{2} \sum_e \oint_{\partial\Omega_{e_i}} [\boldsymbol{n} \cdot (\mathcal{A}_{\mathrm{v}}^T \psi)]^- (u^+ + u^-)\, d\sigma + \sum_e \oint_{\partial\Omega_{e_b}} [\boldsymbol{n} \cdot (\mathcal{A}_{\mathrm{v}}^T \psi)]^- u^b\, d\sigma, \end{aligned} \tag{6}$$

where $\partial\Omega_{e_i}$ denotes the part of $\partial\Omega \notin \Gamma$ and $\partial\Omega_{e_b}$ denotes the part $\in \Gamma$. By summing and subtracting to the right hand side of Eq. (6) the expression

$$\frac{1}{2} \sum_e \oint_{\partial\Omega_{e_i}} [\boldsymbol{n} \cdot (\mathcal{A}_{\mathrm{v}}^T \psi)]^- u^-\, d\sigma + \sum_e \oint_{\partial\Omega_{e_b}} [\boldsymbol{n} \cdot (\mathcal{A}_{\mathrm{v}}^T \psi)]^- u^-\, d\sigma$$

and by "backward" integrating by parts, we can rewrite Eq. (6) as

$$\begin{aligned} \sum_e \int_{\Omega_e} \psi \cdot \boldsymbol{\delta}\, d\Omega = \sum_e \int_{\Omega_e} \psi \cdot \widetilde{\boldsymbol{f}_{\mathrm{v}}}\, d\Omega - \sum_e \int_{\Omega_e} \psi \cdot (\mathcal{A}_{\mathrm{v}} \nabla u)\, d\Omega = \\ \frac{1}{2} \sum_e \oint_{\partial\Omega_{e_i}} [\boldsymbol{n} \cdot (\mathcal{A}_{\mathrm{v}}^T \psi)]^- (u^+ - u^-)\, d\sigma \\ + \sum_e \oint_{\partial\Omega_{e_b}} [\boldsymbol{n} \cdot (\mathcal{A}_{\mathrm{v}}^T \psi)]^- (u^b - u^-)\, d\sigma\ . \end{aligned} \tag{7}$$

The above equation defines the auxiliary variable $\boldsymbol{\delta} = \widetilde{\boldsymbol{f}_{\mathrm{v}}} - \mathcal{A}_{\mathrm{v}} \nabla u$ as the difference between the weakly continuized viscous flux $\widetilde{\boldsymbol{f}_{\mathrm{v}}}$ — which takes into account the effect of the jump $u^+ - u^-$ at element interfaces and of the jump $u^b - u^-$ at the boundary — and the "internal" viscous flux $\mathcal{A}_{\mathrm{v}} \nabla u$. By

rearranging the sums of contour integrals as a sum over internal interfaces plus a sum over boundary faces, we can express $\boldsymbol{\delta}$ as

$$\sum_e \int_{\Omega_e} \boldsymbol{\psi} \cdot \boldsymbol{\delta}\, d\Omega = -\frac{1}{2} \sum_f \int_{\sigma_f} [(\boldsymbol{\mathcal{A}}_{\mathrm{v}}^T \boldsymbol{\psi})^- + (\boldsymbol{\mathcal{A}}_{\mathrm{v}}^T \boldsymbol{\psi})^+] \cdot [(u\boldsymbol{n})^- + (u\boldsymbol{n})^+]\, d\sigma + \sum_b \int_{\sigma_b} [\boldsymbol{n} \cdot (\boldsymbol{\mathcal{A}}_{\mathrm{v}}^T \boldsymbol{\psi})]^- \cdot (u^b - u^-)\, d\sigma \ . \quad (8)$$

We now consider the weak formulation of the last term of the second equation of the system, which, by substituting the physical viscous flux $\boldsymbol{f}_{\mathrm{v}}$ with the weakly continuized flux $\widetilde{\boldsymbol{f}_{\mathrm{v}}} = \boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u + \boldsymbol{\delta}$, and by rearranging the sums of contour integrals as sums of interface and boundary integrals, can be written in weak form as

$$\begin{aligned} &- \sum_e \int_{\Omega_e} \boldsymbol{\nabla}\phi \cdot (\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u)\, d\Omega - \sum_e \int_{\Omega_e} \boldsymbol{\nabla}\phi \cdot \boldsymbol{\delta}\, d\Omega \\ &\quad + \frac{1}{2} \sum_f \int_{\sigma_f} [(\phi \boldsymbol{n})^- + (\phi \boldsymbol{n})^+] \cdot [(\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u)^+ + (\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u)^-]\, d\sigma \\ &\quad + \frac{1}{2} \sum_f \int_{\sigma_f} [(\phi \boldsymbol{n})^- + (\phi \boldsymbol{n})^+] \cdot (\boldsymbol{\delta}^+ + \boldsymbol{\delta}^-)\, d\sigma \\ &\quad + \sum_b \int_{\sigma_b} (\phi \boldsymbol{n})^- \cdot (\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u)^b\, d\sigma + \sum_b \int_{\sigma_b} (\phi \boldsymbol{n})^- \cdot \boldsymbol{\delta}^b\, d\sigma \ . \end{aligned} \quad (9)$$

By virtue of Eq (8), in which we choose the test function $\boldsymbol{\psi} = \boldsymbol{\nabla}\phi$, the second volume integral sum of the above expression can be reformulated as a summation of interface and boundary integrals. The weak formulation of the second equations of system (5) can therefore be written as

$$\begin{aligned} &\mathbb{E} - \sum_e \int_{\Omega_e} \boldsymbol{\nabla}\phi \cdot (\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u)\, d\Omega \\ &\quad + \frac{1}{2} \sum_f \int_{\sigma_f} [(\boldsymbol{\mathcal{A}}_{\mathrm{v}}^T \boldsymbol{\nabla}\phi)^- + (\boldsymbol{\mathcal{A}}_{\mathrm{v}}^T \boldsymbol{\nabla}\phi)^+] \cdot [(u\boldsymbol{n})^- + (u\boldsymbol{n})^+]\, d\sigma \\ &\quad + \frac{1}{2} \sum_f \int_{\sigma_f} [(\phi \boldsymbol{n})^- + (\phi \boldsymbol{n})^+] \cdot [(\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u)^+ + (\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u)^-]\, d\sigma \\ &\quad + \frac{1}{2} \sum_f \int_{\sigma_f} [(\phi \boldsymbol{n})^- + (\phi \boldsymbol{n})^+] \cdot (\boldsymbol{\delta}^+ + \boldsymbol{\delta}^-)\, d\sigma \\ &\quad - \sum_b \int_{\sigma_b} [\boldsymbol{n} \cdot (\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla}\phi)]^- \cdot (u^b - u^-)\, d\sigma \\ &\quad + \sum_b \int_{\sigma_b} (\phi \boldsymbol{n})^- \cdot (\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u)^b\, d\sigma + \sum_b \int_{\sigma_b} (\phi \boldsymbol{n})^- \cdot \boldsymbol{\delta}^b\, d\sigma = 0, \end{aligned} \quad (10)$$

where $\mathbb{E}$ denotes the DG discretization of the inviscid part of the NS equations as given in Eq. (4). The previous equations and Eq. (8) can be regarded as a system of two equations in the unknowns u and $\boldsymbol{\delta}$ which discretize the NS equations in mixed form (see e.g. [1]). Unfortunately a DG Galerkin method for the solution of the NS equations very similar to the one described so far displays an unsatisfactory convergence rate for polynomial approximations of odd order. A cure to this convergence problem is suggested in [4,3], and has been thoroughly investigated theoretically in a series of papers of Brezzi and coworkers [9,8,7].

Following the ideas introduced in [4,3], we therefore replace the variable $\boldsymbol{\delta}$ appearing in the third interface integral and in the third boundary integral of Eq. (10) with a locally defined variable $\boldsymbol{\delta}_f$ given by

$$\int_{\Omega_e{}^{\pm}} \boldsymbol{\psi} \cdot \boldsymbol{\delta}_f^{\pm} \, d\Omega = \frac{1}{2} \int_{\sigma_f} [\boldsymbol{n} \cdot (\boldsymbol{\mathcal{A}}_{\mathrm{v}}^T \boldsymbol{\psi})]^{\pm} (u^{\mp} - u^{\pm}) \, d\sigma \tag{11}$$

for internal interfaces and by

$$\int_{\Omega_e{}^{-}} \boldsymbol{\psi} \cdot \boldsymbol{\delta}_f^{-} \, d\Omega = \int_{\sigma_b} [\boldsymbol{n} \cdot (\boldsymbol{\mathcal{A}}_{\mathrm{v}}^T \boldsymbol{\psi})]^{-} \cdot (u^b - u^{-}) \, d\sigma \tag{12}$$

for boundary sides. The previously defined variable $\boldsymbol{\delta}$ and the newly defined variables $\boldsymbol{\delta}_f$ are related for each element e through the relation

$$\boldsymbol{\delta}|_e = \sum_{f \in e} \boldsymbol{\delta}_f,$$

where the summation includes the faces on the boundary of element e. The modified DG formulation of the NS equations can therefore be written as

$$\begin{aligned}
&\mathbb{E} - \sum_e \int_{\Omega_e} \boldsymbol{\nabla} \phi \cdot (\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u) \, d\Omega \\
&\quad + \frac{1}{2} \sum_f \int_{\sigma_f} [(\boldsymbol{\mathcal{A}}_{\mathrm{v}}^T \boldsymbol{\nabla} \phi)^{-} + (\boldsymbol{\mathcal{A}}_{\mathrm{v}}^T \boldsymbol{\nabla} \phi)^{+}] \cdot [(u\boldsymbol{n})^{-} + (u\boldsymbol{n})^{+}] \, d\sigma \\
&\quad + \frac{1}{2} \sum_f \int_{\sigma_f} [(\phi \boldsymbol{n})^{-} + (\phi \boldsymbol{n})^{+}] \cdot [(\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u)^{+} + (\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u)^{-}] \, d\sigma \\
&\qquad + \frac{1}{2} \sum_f \int_{\sigma_f} [(\phi \boldsymbol{n})^{-} + (\phi \boldsymbol{n})^{+}] \cdot (\boldsymbol{\delta}_f^{+} + \boldsymbol{\delta}_f^{-}) \, d\sigma \\
&\qquad - \sum_b \int_{\sigma_b} [\boldsymbol{n} \cdot (\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} \phi)]^{-} \cdot (u^b - u^{-}) \, d\sigma \\
&\quad + \sum_b \int_{\sigma_b} (\phi \boldsymbol{n})^{-} \cdot (\boldsymbol{\mathcal{A}}_{\mathrm{v}} \boldsymbol{\nabla} u)^b \, d\sigma + \sum_b \int_{\sigma_b} (\phi \boldsymbol{n})^{-} \cdot \boldsymbol{\delta}_f^b \, d\sigma = 0 \ .
\end{aligned} \tag{13}$$

Notice that the second interface integral of the previous equation, i.e. that containing the sum $(\boldsymbol{\delta}_f^+ + \boldsymbol{\delta}_f^-)$, can be rewritten in the equivalent form

$$\frac{1}{2}\int_{\sigma_f} (\phi^+ - \phi^-)[(\boldsymbol{n}\cdot\boldsymbol{\delta}_f)^+ - (\boldsymbol{n}\cdot\boldsymbol{\delta}_f)^-]\, d\sigma,$$

thus showing that what is really needed in the computations is only the normal component $(\boldsymbol{n}\cdot\boldsymbol{\delta}_f)^\pm$ of $\boldsymbol{\delta}_f$. An equation giving the normal component $\boldsymbol{n}\cdot\boldsymbol{\delta}_f$ can be obtained from Eq. (11) for an interface — or from Eq. (12) for a boundary side — by considering test functions $\boldsymbol{\psi} = \phi\boldsymbol{n}$. For an internal interface, $\boldsymbol{n}\cdot\boldsymbol{\delta}_f$ on the face f is for example given by

$$\int_{e^\pm} \phi(\boldsymbol{n}\cdot\boldsymbol{\delta}_f)^\pm\, d\Omega = -\frac{1}{2}\int_{\sigma_f} \phi\left[\boldsymbol{n}\cdot(\boldsymbol{\mathcal{A}}_v^T\boldsymbol{n})\right]^\pm (u^\mp - u^\pm)\, d\sigma\ . \qquad (14)$$

Eq. (13) with the values of $\boldsymbol{\delta}_f$ expressed in terms of $u^\pm$ according to (14) provides a symmetric DG discretization of the viscous terms of the NS equations entirely in terms of the original variable u, which does not suffer the convergence problems related to the auxiliary variables $\boldsymbol{\delta}$ originally introduced to define the weakly continuized viscous flux function. In practice, even if starting from a formulation of the NS equations as a system of two first order equations, we have ended with a formulation in which the auxiliary variables can be computed on a local basis for the internal interfaces and for the boundary sides, with no requirement for additional storage at global level.

The DG discretization of the viscous terms here described bears some resemblance with that introduced by Baumann and Oden (see e.g. [5,6]), which can be obtained from Eq. 13 by removing the third interface and boundary integral summations (i.e. the terms involving $\boldsymbol{\delta}_f$) and by changing the sign of the first interface and boundary integral summations. Because of the different sign, the DG scheme here described leads to a symmetric discretization (for a symmetric matrix $\boldsymbol{\mathcal{A}}_v$) while that of Baumann and Oden does not. Baumann and Oden's scheme is however simpler since it does not require the terms containing $\boldsymbol{\delta}_f$.

The resulting scheme is characterized by a very compact support. In fact, the unknowns associated with element e are only coupled with the unknowns associated with the elements which share a face with e. This results in a discretized spatial operator which can be solved very efficiently and is therefore very well suited to be used with an implicit time integration scheme, as we are going to describe in the next section.

3 Implicit Time Integration

The space discretized equations are advanced in time with the implicit backward Euler time integration scheme. If the equations are linearized in time,

at each time step we have to solve a linear system $\boldsymbol{A}^n \boldsymbol{x}^n = \boldsymbol{b}^n$ of algebraic equations, where $\boldsymbol{x}^n$ is the solution increment $(\boldsymbol{U}^{n+1} - \boldsymbol{U}^n)$ and $\boldsymbol{b}^n = -\boldsymbol{R}^n$. The coefficient matrix $\boldsymbol{A}^n$ can be regarded as a sparse block matrix of $m \times m$ blocks, m being the number of unknown fields (ρ, ρE, ρu, ρv) times the number of degrees of freedom used to represent each field within an element. The number of nonzero blocks of a generic block row i is equal to the number of elements sharing a face with element i plus one. The linearization of the NS equations is simply accomplished by evaluating all the Jacobians, both inviscid and viscous, at time level n, thus reducing both the inviscid and the viscous parts of the NS equations to linear operators in u. The construction of the viscous linearized operator starting from Eq. (13) and (14) is a little bit involved but straightforward.

The linear systems are solved by means of the left preconditioned GMRES iterative solution algorithm. It is well known that the efficiency of the PGMRES algorithm is crucially dependent on the use of a preconditioner matrix $\boldsymbol{P}$. A reasonable compromise between efficiency and storage requirements has been found by using as preconditioner the $m \times m$ block diagonal part of $\boldsymbol{A}$ (see also [6], in which the linear system is solved by Jacobi and Gauss-Seidel iterations).

4 Numerical Results

In order to check the accuracy and the order of convergence of the DG discretization of the viscous term, we have considered the simple test problem

$$\boldsymbol{\nabla} \cdot (\boldsymbol{\mathcal{A}}_v \boldsymbol{\nabla} u) = s, \tag{15}$$

on the unit square with homogeneous Dirichlet conditions, where u is a vector unknown, and $\boldsymbol{\mathcal{A}}_v$ is a full constant matrix. The tests have been performed on 5 successively finer grids starting from an extremely coarse grid containing only 8 triangular elements. In all the cases the method converges with the expected order of accuracy provided that the grid is fine enough to capture the relevant feature of the solution (results not reported for brevity).

We next present the computational results for two laminar calculations around the NACA0012 airfoil at two flow regimes, namely $\mathrm{M}_\infty = 0.8$, $\alpha_\infty = 10°$, $\mathrm{Re} = 73$, and $\mathrm{M}_\infty = 0.5$, $\alpha_\infty = 0°$, $\mathrm{Re} = 5000$. The convergence history of the computations is monitored by the error indicator r_n, defined as

$$r_n = \log_{10} \left[||R(\boldsymbol{U}_P^n)||_{L_2} / ||R(\boldsymbol{U}_Q^0)||_{L_2} \right], \tag{16}$$

where $R(\boldsymbol{U}_P^n)$ denotes the residual at the n-th time step using polynomial approximation of order P (that is the polynomial order corresponding to the solution that we are monitoring), and $R(\boldsymbol{U}_Q^0)$ is the residual function computed for the initial data using polynomial approximation of order Q. This definition of the error indicator allows a comparison among residual histories of computations having different order of accuracy.

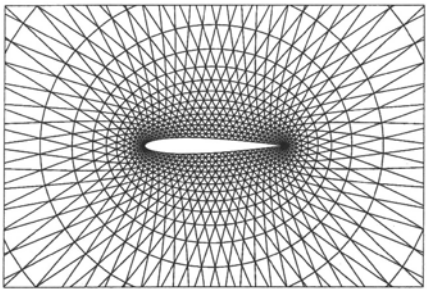

Fig. 1. Grid for the Naca0012 airfioil computations.

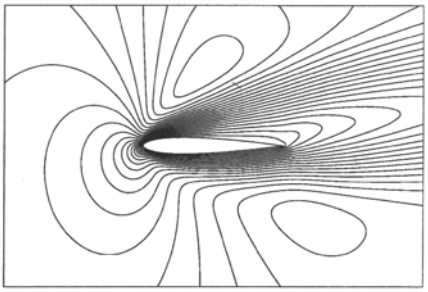

Fig. 2. Re = 73 Mach isolines computed with P3 elements.

In all the computations the CFL number is increased as the residual of the solution decreases according to equation $\mathrm{CFL}^{n+1} = \mathrm{CFL}_0 / r_n^{\beta}$ and subject to the additional user specified constraint $\mathrm{CFL}_0 \leq \mathrm{CFL} \leq \mathrm{CFL}_N$. All the test cases have been computed using $\beta = 1$ and $\mathrm{CFL}_N = 100$. The linear systems have been solved with the restarted GMRES(20) method implemented in the Sparskit2 library developed by Y. Saad and coworkers. All the computations have been performed on a grid containing 2048 triangular élements, 64 distributed along the airfoil and 16 in the direction normal to the airfoil (see Figure 1).

We have performed several test calculations using the ILU preconditioner with various levels of fill-in (up to 20) and the very simple block diagonal preconditioner described in the previous section. Quite surprisingly, for the NACA0012 computations the block diagonal preconditioner gives much better results than the ILU preconditioner both in terms of cpu time and in terms of the number of iterations needed to solve the linear system within a given level of accuracy. In fact, the ILU preconditioner displayed the same rate of convergence of the much simpler block diagonal preconditioner only allowing relatively large levels of fill-in (but with a consequently much greater cost both in terms of cpu time and memory required). This unexpected behavior could be due to a poor ordering of the unknowns, which does not affect the performance of the block diagonal preconditioner but could impair that of the ILU preconditioners. We have however not yet adopted any reordering strategy. We plan to address this issue in future work.

The solution computed with cubic elements for the Re = 73 test case is displayed in Figure 2, which show the accuracy which can obtained even on very coarse grids by using an high-order accurate method. The convergence history for the Re = 73 test case in terms of iterations and of cpu time is displayed in Figures 3 and 4. Notice that we have used as initial data for the high order computations the lower order solution previously computed. The solutions computed with linear, quadratic and cubic elements for the Re = 5000 test case are displayed in Figures 5, 6 and 7. Figure 8 shows a

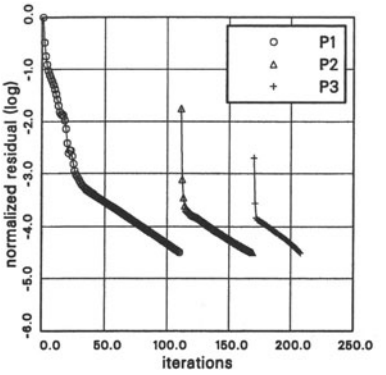

Fig. 3. Re = 73 residual history as a function of time steps.

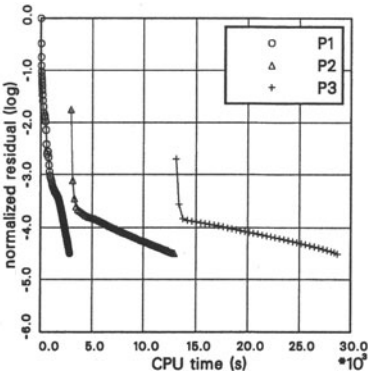

Fig. 4. Re = 73 residual history as a function of cpu time.

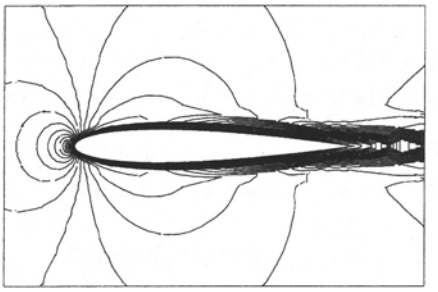

Fig. 5. Re = 5000 Mach isolines computed with P1 elements.

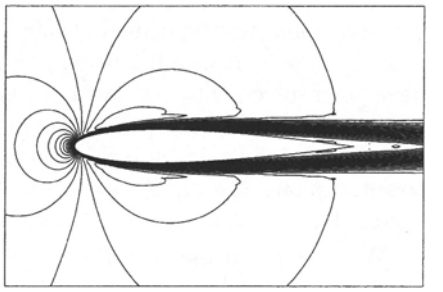

Fig. 6. Re = 5000 Mach isolines computed with P2 elements.

detail of the Mach isolines near the trailing edge and in the wake region behind the airfoil. This test case shows the improvement of the solution accuracy obtained by using high-order element. In fact P3 elements are needed to obtain an accurate solution on this grid (see e.g. [1]). The convergence history of the Re = 5000 test case, again in terms of both iteration number and cpu time, is given in Figures 9 and 10. Notice that the Re = 5000 test case requires less computational effort than the Re = 73 case, suggesting that the system of equations to be solved at each time step gets more ill-conditioned for lower Reynolds numbers.

5 Conclusions

We have here presented a method for the numerical solution of the compressible Navier-Stokes equations based on a DG spatial discretization and on the

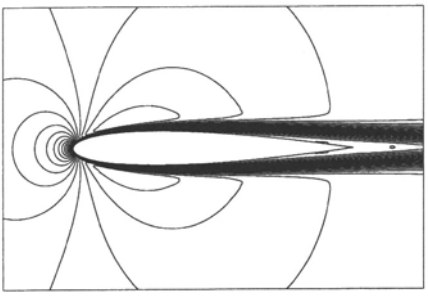

Fig. 7. Re = 5000 Mach isolines computed with P3 elements.

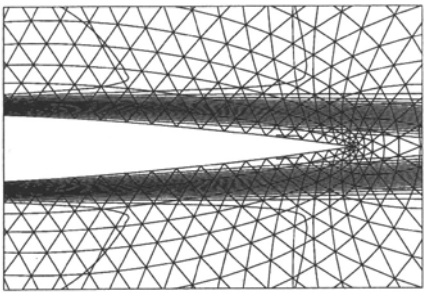

Fig. 8. Re = 5000 Mach isolines detail with P3 elements.

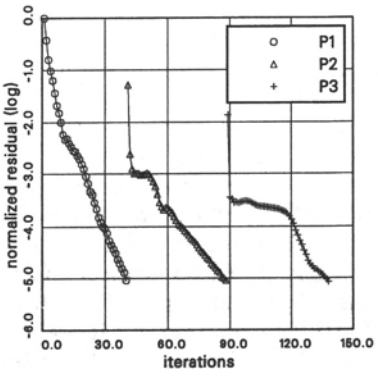

Fig. 9. Re = 5000 residual history as a function of time steps.

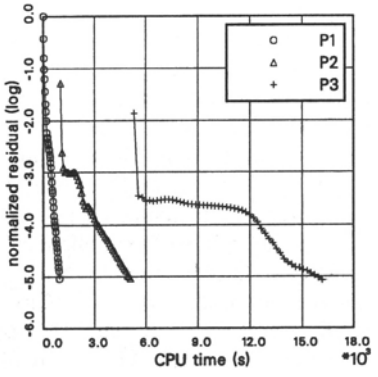

Fig. 10. Re = 5000 residual history as a function of cpu time.

fully implicit backward Euler time integration scheme. The linear systems arising from the implicit time discretization are solved with the preconditioned GMRES method. Various preconditioners have been considered, but the most efficient for the test cases attempted so far turned out to be a relatively simple block diagonal preconditioner. It is not clear at this stage of our investigation if the worst performance of ILU type preconditioners is due to a poor unknown ordering or if it is the block diagonal preconditioner which performs surprisingly well for DG discretizations (not excluding some other conjecture which we can not envisage at this moment). The results obtained so far are quite encouraging but we have considered only relatively simple test cases and we have not yet adopted any kind of reordering technique which could improve the performance of ILU preconditioners. We plan to work in this direction in the near future.

References

1. F. Bassi and S. Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations. *J. Comput. Phys.*, 131:267–279, 1997.
2. F. Bassi and S. Rebay. High-order accurate discontinuous finite element solution of the 2D Euler equations. *J. Comput. Phys.*, 138:251–285, 1997.
3. F. Bassi and S. Rebay. An implicit high-order discontinuous galerkin method for the steady state compressible Navier–Stokes equations. In K. D. Papailiou, D. Tsahalis, D. Périaux, C. Hirsh, and M. Pandolfi, editors, *Computational Fluid Dynamics 98, Proceedings of the Fourth European Computational Fluid Dynamics Conference*, volume 2, pages 1227–1233, Athens, Greece, September 5–7 1998. John Wiley and Sons.
4. F. Bassi, S. Rebay, G. Mariotti, S. Pedinotti, and M. Savini. A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows. In R. Decuypere and G. Dibelius, editors, *2nd European Conference on Turbomachinery Fluid Dynamics and Thermodynamics*, pages 99–108, Antwerpen, Belgium, March 5–7 1997. Technologisch Instituut.
5. C. E. Baumann and J. T. Oden. A discontinuous hp finite element method for convection-diffusion problems. to appear on CMAME, 1998.
6. C. E. Baumann and J. T. Oden. A discontinuous hp finite element method for the Euler and Navier-Stokes equations. to appear on IJNME, 1998.
7. F. Brezzi, G. Manzini, D. Marini, P. Pietra, and A. Russo. Discontinuous finite elements for diffusion problems. Technical Report 1112, IAN-CNR, via Ferrata, 1, 1998. submitted for publication to Numer. Meth. PDEs.
8. F. Brezzi, G. Manzini, D. Marini, P. Pietra, and A. Russo. Discontinuous galerkin approximations for elliptic problems. Technical Report 1110, IAN-CNR, via Ferrata, 1, 1998. to appear in "Atti del Convegno in Memoria di F. Brioschi, Istituto Lombardo di Scienze e Lettere, Milano, 22/23 ottobre 1997".
9. F. Brezzi, G. Manzini, D. Marini, P. Pietra, and P. Russo. Analisi delle proprietà di elementi finiti di tipo discontinuo. Technical Report 1107, IAN-CNR, via Ferrata, 1, 1998. relazione finale del progetto ENEL-MIGALE.
10. B. Cockburn, S. Hou, and C.-W. Shu. The Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case. *Math. Comp.*, 54:454–581, 1990.
11. B. Cockburn and C.-W. Shu. The local discontinuous Galerkin method for time-dependent convection-diffusion systems. SIAM J. Numer. Anal. 35:2440-2463, 1998.
12. D. Hanel, R. Schwane, and G. Seider. On the accuracy of upwind schemes for the solution of the Navier–Stokes equations. AIAA Paper 87-1105 CP, AIAA, July 1987. Proceedings of the AIAA 8th Computational Fluid Dynamics Conference.